Crystal Reports™ 9
User's Guide

Crystal Decisions, Inc.
895 Emerson St.
Palo Alto
California, USA 94301

Issue 1.

Trademark Acknowledgements

Contents

Chapter 6: Crystal Repository

Chapter 7: Designing Optimized Web Reports

Chapter 10: Running Totals

Chapter 11: Multiple Section Reports

Chapter 12: Formatting

Chapter 13: Charting

Chapter 14: Mapping

Chapter 15: OLE

Chapter 16: Cross-Tab Objects

Chapter 17: Creating and Updating OLAP Reports

Chapter 18: Distributing and Viewing Reports

Chapter 19: Report Alerts

Chapter 20: Creating Reports from Excel and Access

Chapter 21: Using Formulas

Chapter 24: Understanding Databases

Appendix A: Report Processing Model

Welcome to Crystal Reports 1

Welcome! This chapter introduces you to Crystal Reports, the world standard for desktop and web reporting, and provides you with an overview of the contents of this User's Guide.

About Crystal Reports

Crystal Reports is designed to work with your database to help you analyze and interpret important information. Crystal Reports makes it easy to create simple reports, and, it also has the comprehensive tools you need to produce complex or specialized reports.

Create any report you can imagine

Crystal Reports is designed to produce the report you want from virtually any data source. Built-in report experts guide you step by step through building reports and completing common reporting tasks. Formulas, cross-tabs, subreports, and conditional formatting help make sense of data and uncover important relationships that might otherwise be hidden. Geographic maps and graphs communicate information visually when words and numbers are simply not enough.

Extend reporting to the Web

The flexibility of Crystal Reports doesn't end with creating reports — your reports can be published in a variety of formats including Microsoft® Word and Excel, E-mail and even over the Web. Advanced Web reporting lets other members of your workgroup view and update shared reports inside their web browser.

Incorporate reports into applications

Application and web developers can save time and meet their users needs by integrating the report processing power of Crystal Reports into their database applications. Support for most popular development languages makes it easy to add reporting to any application.

Whether it’s the web master in IT, the promotion manager in marketing, the database administrator in finance or the CEO, Crystal Reports is a powerful tool designed to help everyone analyze and interpret the information that's important to them.

About this guide

This guide includes procedures for typical reporting tasks such as placing fields, formatting reports, and sorting records. It also contains information on more specific areas of interest such as advanced formula creation and accessing different types of data. Use this guide as a reference for your basic reporting needs as well as an introduction to new concepts in report creation.

Chapter Contents

The following is a short description of each chapter in this guide.

Chapter 1: Welcome to Crystal Reports

Welcome! This chapter introduces you to Crystal Reports, the world standard for desktop and web reporting, and provides you with an overview of the contents of this User's Guide.

Chapter 2: Installing Crystal Reports

This chapter shows you how to install Crystal Reports locally from the product CD. It also describes how to install Crystal Reports to a network server, and how to install from a network server to a workstation machine. Additional topics include "Customizing your installation" and "Crystal Enterprise."

Chapter 3: Quick Start

This chapter begins with an overview of the sample reports and data commonly used with Crystal Reports. It then presents an overview of the Report Creation Wizards—both new and advanced users can benefit from looking at the Standard Report Creation Wizard as it has many steps in common with the other Report Creation Wizards. Finally, the chapter offers two tutorials: one for new users and another for more advanced users.

Chapter 4: Report Design Concepts

This chapter describes the basic concepts of report design and helps you decide what information you want to include in your report.

Chapter 5: Introduction to Reporting

This chapter shows you the basics of creating a report in Crystal Reports. First you'll learn about the report design environment, then about how to select database tables, place objects on a report, and how to sort, group, and total your report data.

Chapter 6: Crystal Repository

This chapter introduces you to the Crystal Repository. It shows you how to add items to the repository, how to update existing items, and how to use repository items in your Crystal reports.

Chapter 7: Designing Optimized Web Reports

This chapter suggests ways to enhance reports so they take advantage of performance enhancements made to Crystal Reports. While the suggestions made here are *especially* important for optimizing the performance of reports distributed over the thin-wire environment of the Web, the majority of the guidelines and procedures are applicable to all of your reports.

Chapter 8: Record Selection

This chapter shows you how to filter the records you want included in a report. For example, using the record selection tools, you can limit the records in your report to include only records for a specific group of customers, a specific range of account numbers, or a particular date range.

Chapter 9: Sorting, Grouping, and Totaling

Sorting, grouping, and totaling are the steps that turn disorganized data into useful information on a report. This chapter describes the types of sorting, grouping, and totaling you can do within a report.

Chapter 10: Running Totals

Running totals are a flexible and powerful way to create specialized summaries and continually incrementing totals.

This chapter shows you how to add a basic running total and a running total within a group to your report. You will also learn how to create conditional running totals and running totals using formulas.

Chapter 11: Multiple Section Reports

This chapter introduces the various types of sophisticated reports you can create using the multiple section reporting capabilities in Crystal Reports. These capabilities enable you to create reports that treat individual values differently based on sets of criteria you establish. These concepts are then applied to the creation of Form Letters.

Chapter 12: Formatting

Formatting refers to changes in the layout and design of a report, as well as the appearance of text, objects, or entire report sections. This chapter details methods you can use to draw attention to data, change the presentation of dates, numbers, and other values, hide unwanted sections, and perform a variety of other formatting tasks to give a report a professional appearance.

Chapter 13: Charting

Crystal Reports enables you to present summarized data in colorful, easy-to-read charts. This chapter demonstrates how to create charts and how to use them in reports to make report data more meaningful and easier to understand. You can choose from a number of chart layouts and types, as well as drill down to see the details behind the graphical summaries and format chart objects.

Chapter 14: Mapping

Crystal Reports enables you to include maps with reports made up of geographic data. This chapter explains how to use maps in reports to make report data more meaningful and easier to interpret. You can customize and rearrange the appearance of a map and activate the drill-down mode to view the details behind the graphical summaries.

Chapter 15: OLE

This chapter explains how Object Linking and Embedding (OLE) can be used to edit graphics or other objects from within your report instead of opening an additional application.

Chapter 16: Cross-Tab Objects

A Cross-Tab object is a grid that displays values that are grouped and summarized in two directions. This chapter provides you with information about how you can use Cross-Tab objects in your report.

Chapter 17: Creating and Updating OLAP Reports

This chapter describes how to create OLAP reports, how to update them when the location of your data changes, and how to work with the data displayed in an OLAP grid.

Chapter 18: Distributing and Viewing Reports

This chapter provides you with information about how to distribute finished reports using a variety of methods (printing, faxing, exporting).It also includes information about Report Parts and how to set up hyperlinks so you can see Report Parts in various Report Viewers. Finally, it ends with information about using smart tags with report objects in Office XP.

Chapter 19: Report Alerts

This chapter provides information about creating and using alerts in your Crystal reports.

Chapter 20: Creating Reports from Excel and Access

This chapter describes the Add-Ins for Microsoft Excel and Microsoft Access. It explains how you create reports from an Access table or query and an Excel spreadsheet with the Crystal Report Wizard.

Chapter 21: Using Formulas

This chapter explains the basics of formulas and introduces you to the Formula Workshop in order for you to begin to create formulas.

Chapter 22: Parameter Fields

This chapter explains what parameter fields are and how they can be applied to create a single report that can be used to access different types of data depending on the user's needs.

Chapter 23: Subreports

A subreport is a report within a report. With subreports, unrelated reports can be combined into a single report. You can coordinate data that otherwise cannot be linked and present different views of the same data in a single report. This chapter shows you how to create and use subreports.

Chapter 24: Understanding Databases

This chapter describes the essentials you'll need to know to understand database design and use. After describing relational databases (including those from ODBC data sources), indexing, and table linking, the chapter introduces the concepts of server-side processing (pushing report processing to the server level) and field mapping (re-establishing report and database field mappings after a database changes).

Appendix A: Report Processing Model

This appendix provides you with in-depth information about the Report Processing model. This model determines the order in which data is accessed and manipulated during report generation.

Glossary

This guide comes with a comprehensive glossary explaining basic database and reporting concepts as well as terms specific to the program.

Online help

Crystal Reports online help includes all topics within the user's guide. It also provides additional reference material ranging from specific information on the function of a button to general information on how to create a report formula based on a sample business scenario.

Locate information quickly

Access online help from the Crystal Reports help menu.

Use the Contents tab to view all major sections in the online help and drill down to specific headings within each section, the Index tab to view topics in alphabetical order, and the Search tab to enter a keyword to view all the sections that relate to the keyword.

Sample Reports

Many topics in the Crystal Reports online help include a list of related sample reports. Use these reports to illustrate concepts the topic describes. The sample reports can be adapted to your own needs.

Sample reports are located in the Crystal Reports directory under `\Samples\En\Reports`.

Product registration

There are several ways you can register your product:

- Fill out the Product Registration form on the Crystal Decisions, Inc. web site at: http://www.crystaldecisions.com/register/
- Print the Product Registration form and fax it to the registration fax number closest to you. Crystal Decisions will then fax you a registration number that can be entered into the product the next time you use it.

  Registration fax numbers

  USA / Canada +1 (604) 681-5147
  United Kingdom +44 (0) 20 8231 0601
  Australia +6 2 9955 7682
  Germany +49 (0) 69 9509 6182
  Hong Kong +852 2893 2727
  Singapore +65 777 8786

Registration is required to access online or telephone technical support. In addition, registering the product ensures that you are kept up-to-date with product advancements.

Crystal Care technical support

To find out about the technical support programs available for Crystal Reports:

- Consult the enclosed Crystal Care information card.
- Go to our support web site at: http://support.crystaldecisions.com/crystalcare/
- Contact your regional office. For details, go to: http://www.crystaldecisions.com/contact/offices.asp

Crystal Training

Whether you're a developer, information technology professional, or business user, we offer a wide range of Crystal Reports training courses designed to build or enhance your existing skills. Courses are available online, at certified training centers, or at your own site:

- For a complete list of training courses and special offers, visit: http://www.crystaldecisions.com/training/
- Or contact your regional office. For details, go to: http://www.crystaldecisions.com/offices/

Crystal Consulting

Our global team of certified consultants and consulting partners can guide you through a corporate-wide solution—including strategy, design, integration and deployment—for the fastest results, maximum performance, and increased productivity.

- To learn more, visit:
  http://www.crystaldecisions.com/consulting/
- Or contact your regional office. For details, go to:
  http://www.crystaldecisions.com/offices/

Document conventions

This guide uses the following conventions:

- Commands and buttons
  For easy recognition within procedures, User Interface (UI) features appear in bold type. For example: On the **File** menu, click **New**.
- Keyboard shortcuts
  Delete means the Delete key, or the Del key on your numeric keypad. Enter means the Enter, Return, or CR key, depending on which of these keys appears on your keyboard.
- Key combinations
  CTRL+KEY, SHIFT+KEY, and ALT+KEY are examples of key combinations. Hold down the first key in the combination and, at the same time, press the second key in the combination (designated above as KEY). For example: CTRL+C means hold the Control key down and press the letter C on your keyboard (CTRL+C is the Windows Copy command).
- Monospaced font indicates data that you enter using your keyboard. For example: In the Formula Editor, type `If Sales > 1000 Then crRed`

Installing Crystal Reports 2

This chapter shows you how to install Crystal Reports locally from the product CD. It also describes how to install Crystal Reports to a network server, and how to install from a network server to a workstation machine. Additional topics include "Customizing your installation" on page 14 and "Crystal Enterprise" on page 15.

Installing Crystal Reports

The Crystal Reports Installation Wizard works with Microsoft Windows Installer to guide you through the installation process. The Installation Wizard automatically recognizes your computer's operating system and updates files as required.

This chapter provides step-by-step instructions for installing Crystal Reports and shows how to customize your installation. The main topics are:

- Installation requirements
- Installing on a local machine from CD
- Installing to and from a network server
- Customizing your installation
- Crystal Enterprise

As one of the final steps in the installation process, you'll be asked if you want to register the product. Follow the on-screen instructions to complete this process.

Installation requirements

Local installation (from CD) requirements

- Microsoft Windows 98 (Second Edition), NT 4.0, 2000, Me, XP.
- Minimum RAM: 32 MB (64 MB for Windows NT).
- Recommended RAM: 64 MB.
- Minimum hard drive space required (all editions): 60 MB.
- Maximum hard drive space required: 350 MB.

We also recommend having an additional 100 MB of free disk space on your C: drive for use by Windows during the installation. If your system does not meet these requirements, the program may not run correctly.

Note: Check the Release Notes on the Crystal Reports CD for the most recent installation requirements.

Network installation requirements

- Microsoft Windows 98 (Second Edition), NT 4.0, 2000, Me, XP.
- Minimum RAM: 32 MB (64 MB for Windows NT).
- Recommended RAM: 64 MB.
- Typical hard drive space required on a network server: 217 MB.
- Typical hard drive space required on a workstation: 105 MB.

We also recommend having an additional 100 MB of free disk space on your C: drive for use by Windows during the installation. If your system does not meet these requirements, the program may not run correctly.

Note: Check the Release Notes on the Crystal Reports CD for the most recent installation requirements.

Installing on a local machine from CD

If you are installing Crystal Reports on a computer running Windows NT, Windows 2000, or Windows XP, you must have Administrator privileges. The installation process creates registry entries and may update some system files that require Administrator rights.

Close all currently running programs and stop as many services as possible when installing Crystal Reports.

If you want to limit the features you install, see "Customizing your installation" on page 14.

To install on a local machine

1. Insert the Crystal Reports CD and, if the CD does not start automatically, browse to your CD-ROM drive and double-click **Setup.exe**.

   Note: Depending on the configuration of your current system, you may receive a dialog box informing you to update existing files. If this happens, click Yes and restart your machine. The Installation Wizard updates the required files.

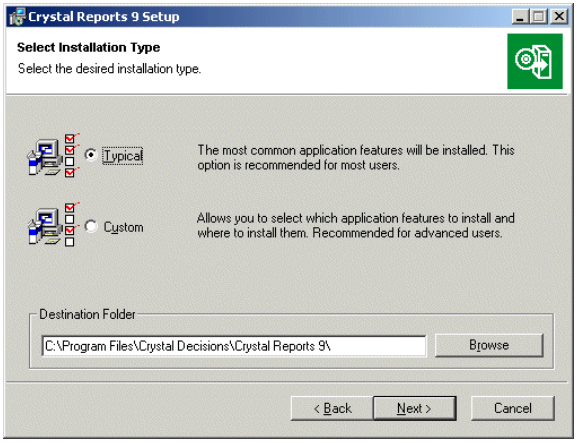
2. Read and accept the License Agreement to proceed with the installation.

3. In the User Information dialog box, type your name, organization, and the Product Key Code.

   Tip: The Product Key Code is printed on the sticker on the back of the CD envelope.

4. Click **Next**.

   The Select Installation Type dialog box appears.



5. Choose the type of installation that you want to perform:
   - **Typical** installs the most common application features.

- **Custom** enables you to choose the features that you want installed, to specify where they will be installed, and to check the disk space required by each feature. For details, see "Customizing your installation" on page 14.

6 Click **Browse** if you want to install Crystal Reports to a directory different from the default location.

The default is `C:\Program Files\Crystal Decisions\Crystal Reports 9\`

7 Click **Next**.

The Start Installation dialog box appears.

8 Click **Next** to begin copying files to your local drive.

Installing to and from a network server

A network installation of Crystal Reports involves two steps:

1 Run the Administrator's installation to copy files to a server machine on the network. See "Installing Crystal Reports to a network" on page 12.

2 Access the server machine from a workstation, and run Setup.exe to install Crystal Reports on the workstation. See "Installing Crystal Reports from a network" on page 13.

Installing Crystal Reports to a network

This procedure must be performed by a network administrator who has write access and network privileges. When this procedure is complete, end users will be able to access Setup.exe from the network to install Crystal Reports onto their local machines.

If you are installing Crystal Reports on a computer running Windows NT, Windows 2000, or Windows XP, you must have Administrator privileges. The installation process creates registry entries and may update some system files that require Administrator rights.

Close all currently running programs and stop as many services as possible when installing Crystal Reports.

Note: If users do not have the Microsoft Windows Installer configured on their machines, the setup process detects the workstation's operating system and installs the appropriate Microsoft Windows Installer package.

To install Crystal Reports to a network

1 Initialize the Administrator's installation by running **Setup.exe** with the additional command line switch **/a**.

For example, click the Start button, click Run, and type:

```
<path> Setup.exe /a
```

where `<path>` is the location of the Crystal Reports setup program.

2 In the Admin Installation dialog box, click **Browse** to choose the network location where you want to install the Crystal Reports files.

3 Click **Next**.

4 In the Admin Installation verification dialog box, click **Next** to begin copying the files to the network.

When the files have finished copying to the network, users can double-click Setup.exe to begin the installation.

Installing Crystal Reports from a network

If your network administrator has installed Crystal Reports to the network, make sure you have read privileges to that network before beginning this process.

If you are installing Crystal Reports on a computer running Windows NT, Windows 2000, or Windows XP, you must have Administrator privileges. The installation process creates registry entries and may update some system files that require Administrator rights.

Close all currently running programs and stop as many services as possible when installing Crystal Reports.

To install Crystal Reports from a network

1 Access the network server that contains the Crystal Reports installation files.

2 Double-click **Setup.exe**.

Note: Depending on the configuration of your current system, you may receive a dialog box informing you to update existing files. Click Yes and restart your machine. The Installation Wizard updates the required files.

3 Read and accept the License Agreement to proceed with the installation.

4 In the User Information dialog box, type your name, organization, and the Product Key Code.

Tip: You may need to contact your Administrator for the Product Key Code.

5 Click **Next**.

The Select Installation Type dialog box appears.

6 Choose the type of installation that you want to perform:

- **Typical** installs the most common application features.
- **Custom** enables you to choose the features that you want installed, to specify where they will be installed, and to check the disk space required by each feature. For details, see "Customizing your installation" on page 14.

Note: If you want to install certain features so they run from the network, choose the Custom installation.

7 Click **Browse** if you want to install Crystal Reports to a directory different from the default location.

The default is `C:\Program Files\Crystal Decisions\Crystal Reports 9\`

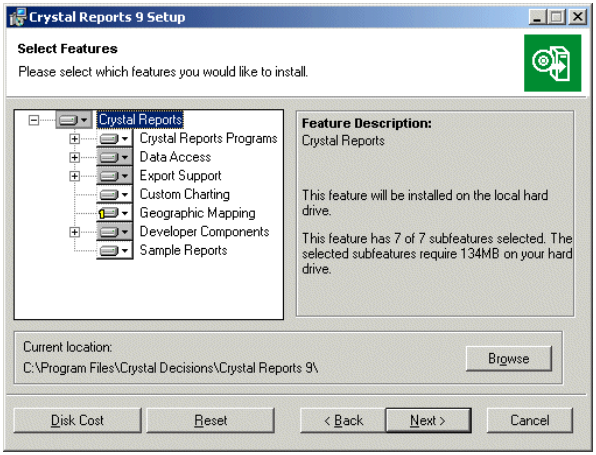
8 Click **Next**.

The Start Install dialog box appears.

9 Click **Next** to begin copying files to your local drive.

Customizing your installation

Selecting the Custom installation option invokes the Select Features dialog box, which allows you to install specific features, to change the default location of various features, and to check the amount of disk space required by each feature.



The icons in the feature tree indicate whether the feature and its subfeatures will be installed or not:

- A white icon means that the feature and all its subfeatures will be installed.
- A shaded icon means that the feature and some of its subfeatures will be installed.
- A yellow 1 means that the feature will be installed when required (installed on demand).
- A red X means that the feature or subfeature is either unavailable or will not be installed.

Crystal Reports uses an “install on-demand” technology for some of its features. As a result, the very first time a particular feature is used after being installed, there may be an extra wait for the “install on-demand” to complete. This behavior will affect new installations only once and will not occur when features are restarted.

To select the configuration and location of a feature or subfeature, click its icon.

Note: Each feature or subfeature can have its own configuration and location.

Use the following table to determine your installation options for each feature or subfeature:

Type of feature installation	Description of what is installed
Will be installed on local hard drive	• Installs the feature on the local hard drive. • Uses the Typical install settings to install some of the feature's subfeatures to the local hard drive.
Entire feature will be installed on local hard drive	• Installs the feature and all of its subfeatures on the local hard drive.
Will be installed to run from CD / network	• Runs the feature and its Typical subfeatures off the CD / network. **Note:** Some subfeatures are not set up to run from the CD / network. These subfeatures will be installed on your local hard drive.
Entire feature will be installed to run from CD / network	• Runs the feature and all of its subfeatures off the CD / network. **Note:** Some subfeatures are not set up to run from the CD / network. These subfeatures will be installed on your local hard drive.
Feature will be installed when required	• Installs the feature or subfeature from the CD / network when first used.
Entire feature will be unavailable	• Neither the feature nor its subfeatures are installed.

Note: Subfeatures are listed below each feature. A subfeature can have a different type of installation than its parent feature.

Crystal Enterprise

Crystal Enterprise provides Crystal Reports with web capabilities and introduces features such as report scheduling, security, and scalability.

The version of Crystal Enterprise included with some editions of Crystal Reports 9 installs all the components necessary for running Crystal Enterprise on one machine.

All services are enabled after the installation is complete. An Administrator account is created as well as a guest user account. Sample reports are also installed.

For instructions on installing Crystal Enterprise, see the Documentation folder on the CD.

Quick Start 3

This chapter begins with an overview of the sample reports and data commonly used with Crystal Reports. It then presents an overview of the Report Creation Wizards—both new and advanced users can benefit from looking at the Standard Report Creation Wizard as it has many steps in common with the other Report Creation Wizards. Finally, the chapter offers two tutorials: one for new users and another for more advanced users.

Learning how to use Crystal Reports

You can teach yourself how to use Crystal Reports by choosing from the methods available in this chapter:

- You can study the sample reports and sample database included with Crystal Reports.
- You can use the detailed descriptions and instructions in the "Quick start for new users" on page 20.
- You can use the summaries and topic cross-references in the "Quick start for advanced users" on page 39—especially useful if you're already familiar with reporting concepts.

Each method is a helpful way to learn and understand Crystal Reports and, although any one might be enough to get you up and running, you can always come back to this chapter and consult the other methods as you need them.

Sample data - Xtreme.mdb

Crystal Reports comes with Xtreme.mdb, a sample database you can use when learning the program. Xtreme.mdb is a Microsoft Access database and all of the necessary drivers are included. You should be able to open the database directly and begin designing reports. Virtually all of the examples in this manual are based on Xtreme.mdb data.

Xtreme.mdb is a database that contains data for Xtreme Mountain Bikes, a fictitious manufacturer of mountain bikes and accessories.

Note: The sample data has been designed to illustrate various reporting concepts in a training environment, not to teach database design. While there are alternative ways of designing a database, this design was selected to keep the tutorials and examples focused on reporting, not on data manipulation.

Report Creation Wizards

The tutorials in this chapter show you how to build a report from scratch. As a complement or an alternative, however, you may want to use the Report Creation Wizards available through the Crystal Reports Gallery. There are four Report Creation Wizards:

- Standard
- Cross-Tab
- Mail Label
- OLAP

Each wizard guides you through the creation of a report by providing a series of screens. Many of the wizards have screens unique to a specific type of report. For example, the Mailing Labels Report Creation Wizard has a screen that allows you to specify the type of mailing label you want to use.

Standard

The Standard Report Creation Wizard is the most generic of the wizards. It guides you through choosing a data source and linking database tables. It also helps you add fields and specify the grouping, summarization (totals), and sorting criteria you want to use. Finally, the Standard Report Creation Wizard leads you through chart creation and record selection.

The Templates screen contains predefined layouts for you to apply to your report to give it more impact.

Cross-Tab

The Cross-Tab Report Creation Wizard guides you through the creation of a report in which your data is displayed as a cross-tab object. Two special screens (Cross-Tab and Grid Style) help you create and format the cross-tab itself.

Mail Label

The Mailing Labels Report Creation Wizard lets you create a report that is formatted to print on any size mailing label. You can use the Label screen to select a commercial label type, or you can define your own layout of rows and columns for any multi-column style report.

OLAP

The OLAP Report Creation Wizard lets you create a report in which your OLAP data is displayed as a grid object. Although similar to the Cross-Tab Report Creation Wizard in several ways, the OLAP Report Creation Wizard appears to be different due to the requirements of working with OLAP data sources. You first specify the location of your OLAP data, and then you choose the dimensions you want to include in the grid. Next you filter the report data and choose the style of the grid object, which you can also customize. Finally, you can define labels for your grid and insert a chart, if you wish.

This is only a brief overview of the four Report Creation Wizards available in Crystal Reports.

Quick start for new users

The following tutorial has been designed to give you confidence when creating your first report.

In this tutorial, you will get an introduction to the program as you create a Customer List report. The Customer List is one of the most basic business reports and typically has information such as Customer Name, City, Region, and Contact Name.

You begin by learning the basic concepts: selecting a database, placing some fields on the report, and then selecting specific records to be included. You will then learn how to:

- Insert and move database fields.
- Add and format a title.
- Display a report in the Preview tab so you can fine-tune your work.
- Use the Select Expert to ensure the report includes only the data you need.
- Move objects.
- Group and sort data.
- Insert pictures.
- Print a report.

Before you begin

This tutorial assumes you are familiar with Microsoft Windows and uses conventional terms and procedures common to the Windows environment. If you are not familiar with Windows, please refer to the documentation that came with Microsoft Windows for further explanation.

The default font for all report sections in the program is set to Arial, 10 point. If you have changed the default font, or if your printer does not support this font, the field size, field spacing, and screen shots will look different than those included in this tutorial.

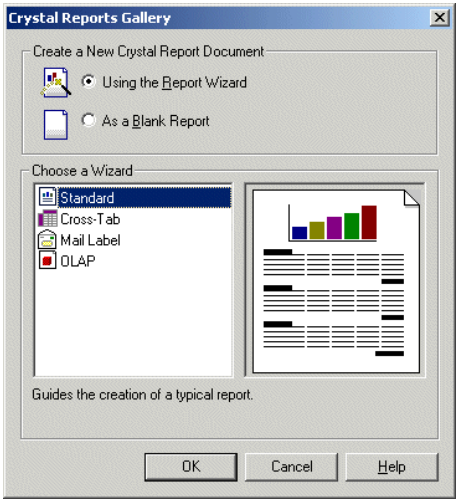
This tutorial has been designed using Microsoft Windows 2000. Screen shots may vary slightly if you are using a different platform.

If you are not familiar with the Crystal Reports environment, review "Formatting" on page 173, which describes working with the grid, free-form placement, using guidelines, and formatting activities.

Creating the report

1 Click **New** on the Standard toolbar.

The Crystal Reports Gallery appears.



The gallery contains a number of wizards to guide you through the creation of specific types of reports. Since you will be learning reporting concepts here, you can skip the wizards and build your report from scratch. After you have completed this tutorial, you may want to build some reports using the wizards to decide which method of report construction you are most comfortable with.

2 Select **As a Blank Report** on the Crystal Reports Gallery dialog box and click **OK**.

The Database Expert dialog box appears.

Note: You can create reports based on database files, SQL / ODBC data sources, dictionary files, query files, and a variety of other data sources.

Selecting a database to use

The next step in creating a report is to select a database. Select the Xtreme.mdb sample database for this tutorial.

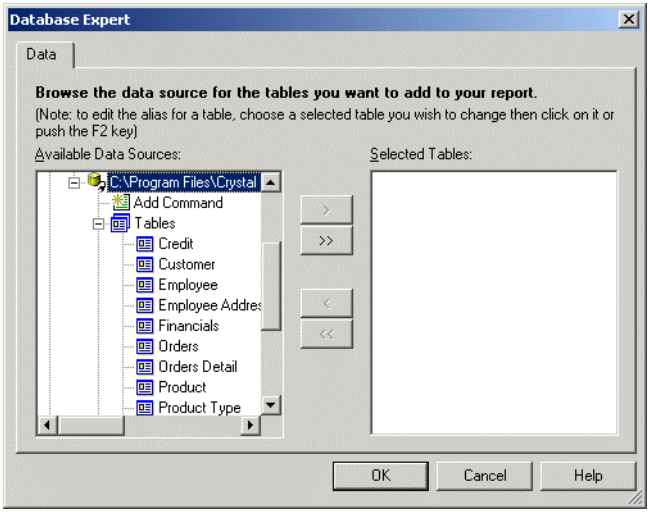
To select a database

1 In the Database Expert dialog box, go to the **Create New Connection** folder and expand **Database Files**; then search for the **Xtreme.mdb** sample database.

By default, this file was installed in the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory.

Note: You have to use the Open dialog box to find the Xtreme database. If you wish to see database and server properties, right-click the database in the Database Expert and select Properties from the shortcut menu.

2 Expand the **Tables** node of the **Xtreme.mdb** connection to see a list of tables.



Because you are dealing only with customers in this tutorial, you will select the Customer table.

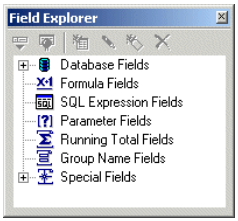
3 Select Customer and click the > arrow to add it to the Selected Tables list, and then click **OK**.

The Design tab of the Report Designer appears.

4 Click **Field Explorer** on the **Standard** toolbar.

The Field Explorer dialog box appears.

Note: Depending on how it appeared when you last used Crystal Reports, the Field Explorer dialog box might be docked or in floating mode. You can manually dock the Field Explorer on the left side, the right side, or the bottom of the designer. In floating mode, the Field Explorer dialog box can be dragged to any location in the designer. You can also resize the dialog box by dragging any of its edges with the Resizing cursor.



Report sections

The Design tab is divided into five sections: Report Header (RH), Page Header (PH), Details (D), Report Footer (RF), and Page Footer (PF). If at any time you are unsure of the report section in which you are working, simply look at the shaded area to the left of the report which always displays either the section names or the initials that designate the names. See "Design tab" on page 59.

If the Short Section Names check box is selected in the Design View area of the Options dialog box, then the Report Header, Page Header, Details, Report Footer and Page Footer section names will appear as RH, PH, D, RF and PF respectively. If this check box is not selected, follow these steps:

1. On the **File** menu, click **Options**. The Options dialog box appears with the Layout tab active.
2. In the Design View area, select the **Short Section Names** check box.
3. Click **OK** to return to the report.

Inserting a field

You'll use the Field Explorer dialog box to insert database fields when you create a new report.

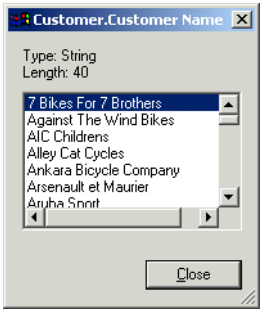
This dialog box is set to remain on-screen until you close it. All the tables available for use are listed in this box.

You will now start placing objects on the report by inserting the Customer Name field.

To insert a field

1. Expand the **Database Fields** node in the Field Explorer dialog box and expand a table.
2. Highlight a field name by clicking the name once.

   When you highlight a field name, you can review a subset of the values for that field as well as the field type and size by right-clicking and selecting Browse Data from the shortcut menu.



3 Click the **Customer Name** field and drag it into the **Details** section of the report.

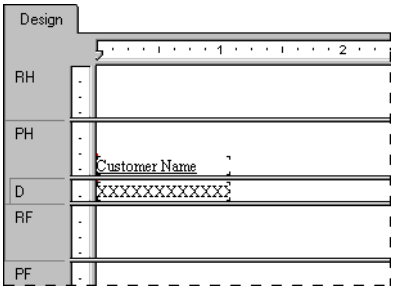
An object frame appears with the Arrow cursor as you drag the field onto the report:

- The object frame represents the object you have just selected for placement.
- The size of the object frame approximates the size of the data in the field selected.

4 Move the object frame as far to the left as you can in the Details section.

If you move the field too far to the left, the Arrow cursor turns into a Stop cursor, indicating that you cannot drag the field that far. Keep in mind that you cannot place any objects outside the page margin.

The Design tab should look similar to this:



Understanding fields

Before going any further, take a look at the field you just placed in the Details section:

- First of all, the object frame indicates that when the report is printed, a field value will appear where the box is positioned.
- The X's in the object frame indicate that the database field contains a text string. Other data types have different character representations. For example, a currency data type is represented by $55,555.56.

  Note: The X's appear only if you haven't selected the Show Field Names check box on the Layout tab of the Options dialog box.
- The number of X's in the object frame is the data width, the maximum number of characters in the field as defined by the database. The width of the object frame is the field width (the amount of space allocated to the field for printing). Initially it is set to the width needed to display the maximum number of characters in the field (using the font selected for the field). You can change this width by resizing the field.
- The size of the X's indicates the point size selected for the characters in the field.

- The font and style (Bold, Underline, and so on) used in displaying the X's indicate the font and style selected for the characters in the field. Later in this tutorial you will learn how to make changes to these properties.
- The line spacing is adjusted to the point size selected for the characters in the field.

Adding additional fields

Next, you will insert two additional fields in the report. This time, however, you will use the Ctrl-click combination to add them at the same time.

To add additional fields

1. Highlight the **City** field in the Field Explorer dialog box, press the Ctrl key on your keyboard, and then highlight the **Country** field. Release the Ctrl key.

   If you scroll through the field list, you will notice that both fields remain selected.

   Note: Using the Ctrl-click combination allows you to select a non-continuous range of fields. The Shift-click combination can be used to select several fields from the list that are contiguous.

2. Drag the fields to place them.

   As the cursor is moved over the report, an object frame appears along with the Arrow cursor.

3. Place the fields to the right of the **Customer Name** field.

   Both fields appear in the Details section of the report in the same order in which they are listed in the Field Explorer dialog box.

Selecting fields

When a field is selected, the object frame appears with a handle (box) on its right, left, top, and bottom edge. These handles indicate that the field is selected, and therefore active. To do anything with a field (change the font, move it, and so on), you first have to select it:

- Position the cursor inside the object frame and click once. The handles appear, indicating the object is selected.
- Move the cursor away from the object frame and click in an empty part of the window. The handles disappear.

That's all it takes to select and deselect objects.

Resizing fields

To resize the field, follow these steps:

1. Click the **Customer Name** field in the Details section to select it.
2. Press the Ctrl key and click the field heading to select both objects.

3 Move the cursor over the resizing handle on the right edge of the fields until the cursor turns into a Resizing cursor.

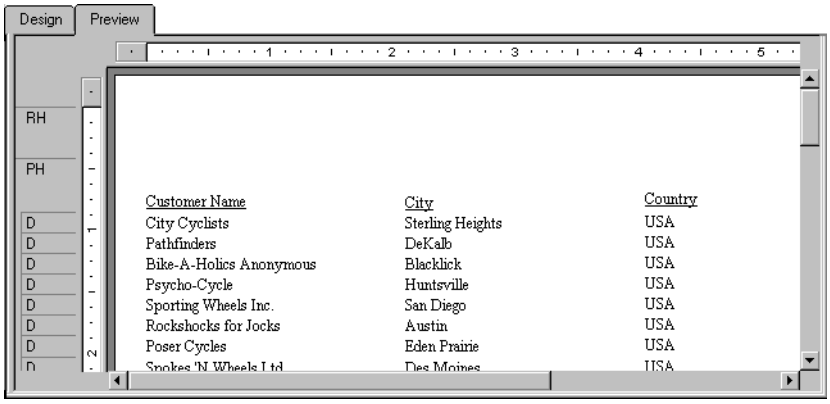
4 Resize the fields to the left until they are approximately two inches in length.

Reviewing your work

Now let's see how the report looks with three fields placed and positioned.

1 Click **Print Preview** on the Standard toolbar to activate the **Preview** tab.

The screen should look similar to this:



Note: The first time you preview a report, you must click Preview on the Standard toolbar to activate the Preview tab. The Preview tab appears to the right of the Design tab. You can then switch between designing and previewing the report by clicking the corresponding tab.

You have the beginnings of a customer list report, but you still have several fields to add.

2 When you are finished reviewing the report, return to the **Design** tab by clicking it.

Combining database fields in a text object

Instead of adding the Contact First Name and Contact Last Name fields as separate objects, you can insert both fields in a text object. This allows you to control the formatting of both fields by making changes to only one object. When you insert fields in a text object, the fields are automatically trimmed (they do not have any extra white space on either side). This is important because a field is a fixed size but the data in the fields can vary in size, leaving various amounts of unwanted white space.

To combine fields in a text object

1. Click **Insert Text Object** on the Insert Tools toolbar. As you move the cursor over the report, an object frame appears next to it.
2. Insert the field to the right of the fields in the **Details** section.

   As you drag the field, the Design tab automatically scrolls to the right, if necessary. When you click to place the object, a text object appears and the horizontal ruler of the Design tab changes to a ruler / tab selector that is used for editing the text object. If you click an empty area of the report or a field object, the standard Design tab ruler appears.
3. Click once on the border of the text object to select it for resizing. Handles appear on all sides of the object.
4. Move the cursor over the right sizing handle of the text object and increase the width by about 1 inch. You may need to scroll to the right and continue resizing.
5. Double-click inside the text object to select it for editing. Notice the insertion point is now flashing within the text object.
6. Select the **Contact Last Name** field in the Field Explorer dialog box. Remember, you can move the Field Explorer dialog box by dragging and dropping it by its title bar.
7. Drag the field to the text object.
8. Move the cursor over the text object until the cursor becomes a Drag and Drop cursor.
9. Release the mouse button to place the field in the text object. The cursor now appears after the **Contact Last Name** field, within the text object.
10. Type a comma and a space after **Contact Last Name**.
11. In the Field Explorer dialog box, highlight the **Contact First Name** field.
12. Drag the field to the text object.
13. Move the cursor over the text object until the cursor becomes a Drag and Drop cursor. Move the cursor to the right of the comma and space you just typed, and release the mouse button. The field will be inserted to the right of the comma and space.
14. Click the **Preview** tab to look at the fields you just placed.

The report should now look similar to this:

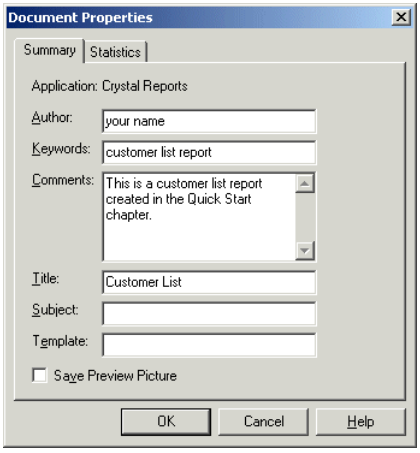
Adding summary information

The next step is to add summary information to your report. Adding summary information allows you to specify the author, title, and subject of the report, as well as any keywords or comments related to the report. When you add summary information, users can find information related to the report quickly.

To add summary information

1. On the **File** menu, click **Summary Info**.

   The Document Properties dialog box appears with the Summary tab active.



2. Enter information about your report in the text boxes provided. Be sure to enter the title “Customer List” in the **Title** text box. This information will be used in the next section of the tutorial.

3. Click **OK** when finished.

Adding a title

As you can see, the report looks incomplete without a title. Although you can add a title using a text object, you can also tell the program to take the title information directly from the Title text box in the Document Properties dialog box.

To add a title

1. Click the **Design** tab.
2. In the Field Explorer, scroll down to **Special Fields** and expand it.
3. Choose **Report Title**.
4. Drag the cursor over the report. An object frame appears.
5. Position the object frame in the upper left-hand corner of the Page Header (PH) section of the report and release the mouse button to place the object.
6. Click the **Preview** tab to review your changes.

The report title object now displays the title that you entered in the Title text box of the Document Properties dialog box.

Formatting objects

Now you can format the report title. This time, however, you will remain in the Preview tab to do the work. This will make it easier to see your work while you are formatting the title.

To format an object

1. To center the title, you will first need to expand the **Title** field so that it's about the same width as the data in your report. To do this, select the object by clicking it.
2. Position the cursor on the right edge of the object until the cursor turns into a Resizing cursor. Drag the right edge of the field box until it is even with the right edge of the data in the **Contact Name** field object.

   You have created a large field that extends from the left edge to the right edge of the report.

3. With the report title object still selected, click **Align Center** on the Formatting toolbar. The title is centered within the object.
4. Right-click the object and choose **Format Field** from the shortcut menu.
5. When the Format Editor appears, click the **Font** tab.
6. Set the report title to a larger, bolder version of the active font by selecting **Bold** from the **Style** list and **16** (or a point size suitable to the font you are using) from the **Size** list.

7 Change the color of the text by selecting **Maroon** from the **Color** palette. Notice that the Sample box shows an example of how the text will look.

8 Click **OK** when finished.

9 Resize the report title object vertically to accommodate the increased size of the title.

The title is now formatted to stand out on the report.

Adding a field heading

As you can see, the Contact Name field is the only field without a heading. In this section you will create a heading.

To add a field heading

1 Select the **Contact Name** field on either the Design or Preview tab.

2 On the **Insert** menu, click **Field Heading**.

A field heading is added to the Page Header just above the Contact Name field.

The Contact Name field now has a heading that looks just like the other field titles.

Saving the report

1 Click **Save** on the Standard toolbar to save your work.

Since this is the first time you are saving the report, the Save As dialog box appears displaying the default directory where the file will be saved. Notice that a default file name, based on the report title you defined earlier, is also displayed.

2 Type Custlist.rpt in the **File name** box and click **Save**.

Your report is saved to the default directory or another directory you chose.

3 Click the **Preview** tab to view the report.

The report should now look similar to the following:

Congratulations! You have just created a basic listing report. You will continue to refine this report throughout the rest of this tutorial.

Record Selection

Crystal Reports allows you to limit or restrict the records that are to be included in a report. In this section you will learn how to:

- Select the records you want included in the report.
- Save a report, including the selection criteria.

For example, it may be useful to have a customer list that only lists customers from the USA. The sample data contains records from the United States and International customers. It is easy to restrict lists like this using the Select Expert. See "Record Selection" on page 107.

Entering the selection criteria

When you scroll through the report, you will see that it contains information for customers from many different countries. In this step, you will limit the number of countries displayed to the USA.

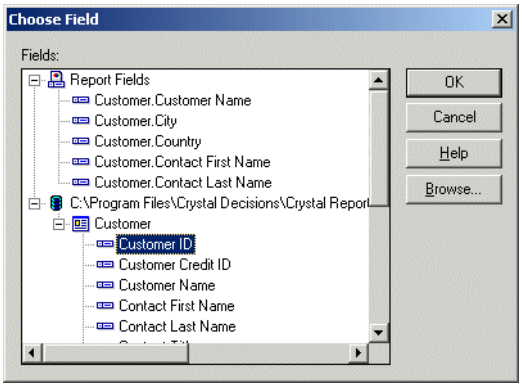
To enter selection criteria

1 To begin, click the **Design** tab to return to design mode.

2 Click an empty area of the report to make sure all fields are deselected.

3 Click **Select Expert** on the Expert Tools toolbar.

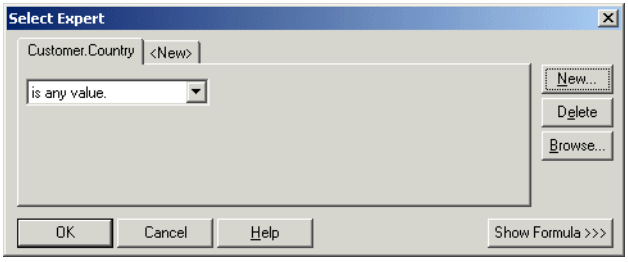
The Choose Field dialog box appears.



This dialog box lists all the fields currently in the report in the Report Fields section and then lists all fields that are available from each table in the database fields section.

4 Since you are going to base record selection on the country field, select Country in the **Fields** list and click **OK**.

The Select Expert appears.



Imagine that you are completing the following sentence:

`Select all records where a customer's country is`

You complete the sentence with the condition you want the program to use when selecting records for your report. Right now the condition is *any value*, which implies there is no restriction on the record selection.

5 Click the arrow on the operators box to see what other options you have. Since you want only those records where the Country is USA, select the **is equal to** condition.

A new box appears on the right. The dialog box sentence now reads:

`Select all records where a customer's country is equal to`

All that you need to complete the sentence is the value USA.

6 Click the arrow on the empty list. A list of all the country values appears. Select **USA** from the list.

The sentence now reads:

```
Select all records where a customer's country is equal to USA
```

7 Click **OK** to return to the **Design** tab.

8 Click the **Preview** tab to review the results of your work.

The Change In Record Selection Formula Detected dialog box appears.

9 Click **Refresh Data**.

10 Save this version of the report without overwriting the original report by choosing **Save As** from the File menu and giving the new report the name USA.rpt.

Congratulations! You have started formatting your report and have added selection criteria to it. More than that, you have learned how to manipulate your data. By now, you have a good idea of the powerful kinds of reports you can prepare.

Deleting a field

Now that the report contains only records from the USA, displaying the Country field in the body of the report is not necessary. You can delete this field before continuing.

To delete a field

1 Select the **Country** field and the **Country** column heading using the Ctrl-click combination.

2 Press Delete on your keyboard.

That is all it takes to delete fields from the report.

The report should now look similar to this:

Balancing field spacing

Now that the Country field has been deleted, there is a large amount of white space between the City and Contact Name fields. You might be satisfied with the spacing as it stands, but it might be more readable if the columns were better balanced across the page.

To balance field spacing

1. Return to the **Design** tab. Select the **Contact Name** field and its field heading by using the Ctrl-click combination.
2. Place the cursor over one of the two highlighted text objects and drag them to the left, closer to the **City** field.
3. Click the **Preview** tab and review your work again.

   The report should look similar to this:

   The spacing between the fields is much better, but it looks as if the report title is off-center.

4. Click the report title object to select it.
5. Position the cursor on the right handle of the object until the cursor turns into a Resizing cursor. Drag the right edge of the object frame until it is even with the right edge of the data in the **Contact Name** field object.

   The report title automatically centers itself based on the size of the object.

Grouping and sorting

Reports can be grouped and sorted in a variety of ways. Sorting and grouping tools provide a great deal of flexibility for customizing reports.

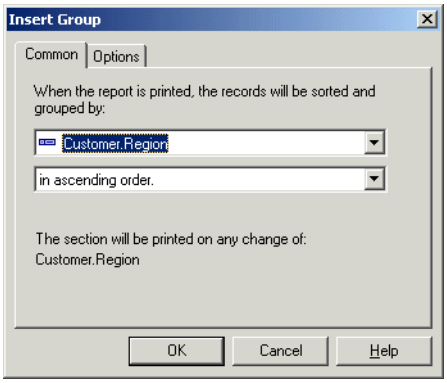
Grouping the report

In many reports you need to break the data into groups in order to make it easier to read and to understand. Crystal Reports lets you do this easily. For this customer list, you will group the customers by region and then sort the customers alphabetically within each group.

To group a report

1 While on the **Design** tab, click **Insert Group** on the Insert Tools toolbar.

The Insert Group dialog box appears.



2 Select Region from the Customer table in the first drop-down list.

The program takes all records with the same value in the region field and places them together in a group on the report.

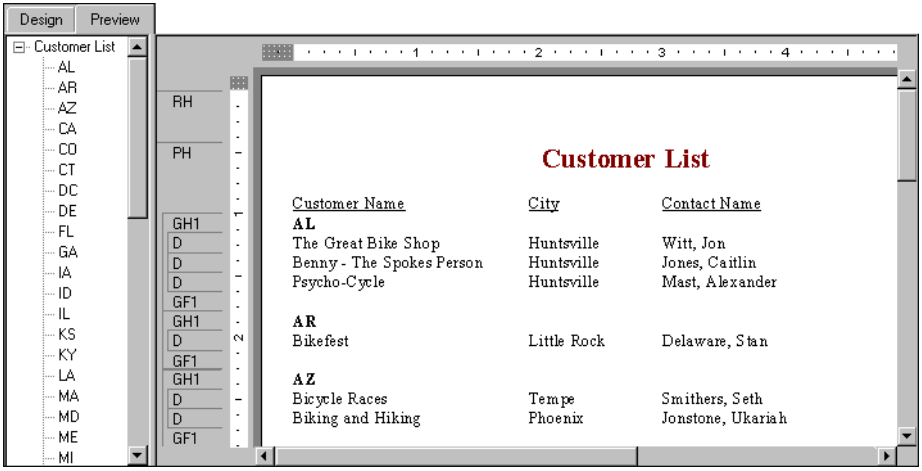
3 Select **in ascending order** from the second drop-down list.

The region grouping will be displayed on the report in alphabetic ascending order.

4 Click **OK**.

Notice that two new sections now appear in the Design tab: GH1 (Group Header #1) and GF1 (Group Footer #1). This is how the program shows that the report has been grouped.

5 Click the **Preview** tab to see what the report looks like.



6 If the group tree is not visible, select **Toggle Group Tree** on the Standard toolbar to see the groups included in the report.

You can view the group of interest by clicking on the group name in the Group Tree. For example, to see the Texas customer group, click TX in the Group Tree. The program jumps to the Texas group, displaying that group in the Preview tab. The Group Tree allows you to quickly jump to a specific group of interest instead of scrolling through the report looking for the group. For more information on the group tree, see "Group Tree view" on page 63.

Note: For many reports, you will want to insert summaries, subtotals and grand totals. For example, when creating a sales report rather than a customer list, you would want to calculate the total sales amount for each region. See "Sorting, Grouping, and Totaling" on page 119.

Understanding "live" group headers

When a group is inserted, a group name field is automatically inserted in the Group Header section of the report. The group name field displays the current group's name. For example, if you group by region, and preview the report, the group header for the CA (California) group shows "CA."

The group field name is automatically formatted to stand out from the records in the group.

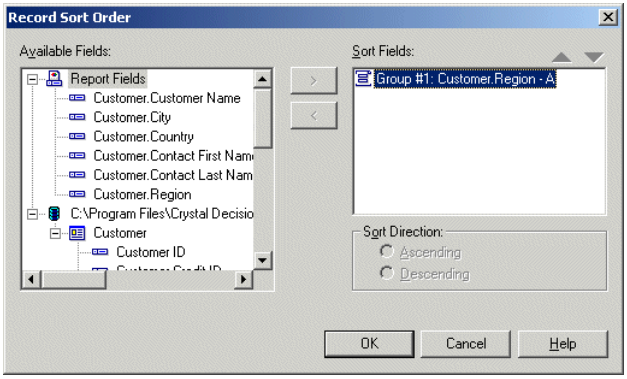
Sorting records

In a typical customer list report, customer names are listed alphabetically. In this example, you will sort the customer names alphabetically within each region.

To sort records

1 While on the **Preview** tab, click **Record Sort Expert** on the Expert Tools toolbar.

The Record Sort Order dialog box appears.

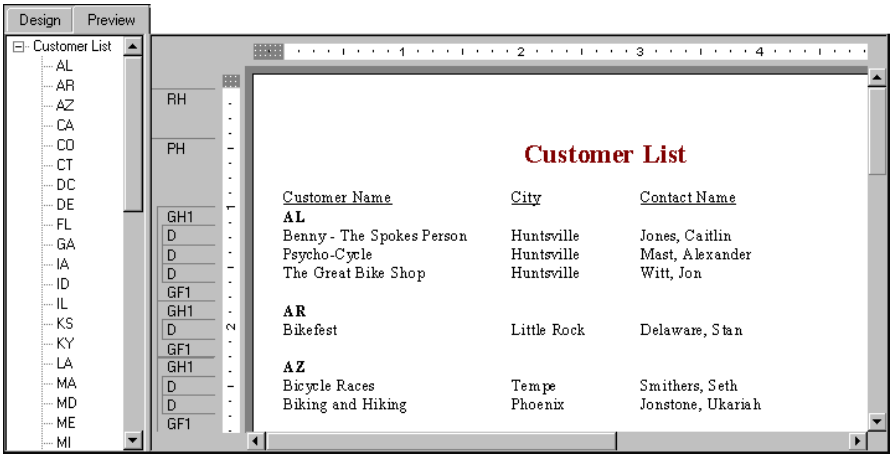


The Available Fields list box displays all fields currently on your report and all the fields in the data source. You can choose to sort based on any of these fields.

The Sort Fields list box displays the fields that are already sorted in the report. Since the region field has already been sorted, the sorting you are about to do will be within each region, and not for the entire report.

2 Highlight the **Customer Name** field and click the > arrow button to add it to the Sort Fields list.

3 Select **Ascending** for the Sort Direction and click **OK**. The report should now look similar to the following:



Notice that the records within each group are in alphabetic order.

Completing the report

You have just one step left to complete the report. A company logo needs to be added to the first page of the report.

Inserting a company logo

In this section, you will place a company logo at the top of the first page of the report.

To insert a company logo

1. While on the **Design** tab, click **Insert Picture** on the Insert Tools toolbar.

   The Open dialog box appears.

2. Choose the **Xtreme.bmp** and click **Open**.

   An object frame appears as you move the cursor over the report. The object frame represents the logo you will place.

   Tip: The Xtreme.bmp file can be found in the Databases subfolder of the `Samples\En` folder.

3. Position the object frame in the upper left-hand corner of the Report Header (RH) section of the report and click once to place it.

   Placing the logo in the Report Header (RH) section ensures that the logo is printed only on the first page of the report.

   Note: Although there does not appear to be enough room in the Report Header (RH) section when you place the graphic, the section will automatically expand to accommodate the picture.

4. Click the **Preview** tab to view the report.

   The final report should look like this:

5 Save the report by clicking **Save** on the Standard toolbar.

You have just completed your first report.

Related topics

For information about distributing reports (printing, exporting, and so on) see the printing topics in "Formatting" on page 173 and "Distributing and Viewing Reports" on page 297.

Quick start for advanced users

If you are an experienced Windows user who wants to get right into the program, follow these steps to set up a report for the first time.

If you're not an experienced user, check the "Quick start for new users" on page 20.

To choose a report type and data source

1 In Windows, click **Start**, point to **Programs**, then select Crystal Reports 9.

2 Click **As a Blank Report** and **OK** to open the Database Expert dialog box.

As an alternative to creating a blank report, you can choose one of four Report Creation Wizards. View style samples for each by selecting from the list of wizards.

After choosing a wizard, you can build your report on a variety of data sources.

3 Browse through the contents of the Database Expert to find the data source you want to use for your report.

4 Select a table you want to use in your report and click the > arrow to add it to the Selected Tables list.

5 After you have added all the tables you want to work with, click **OK** on the Database Expert.

Note: If you select more than one table in the Database Expert dialog box, the Links tab appears. For more information on linking, see "Linking multiple tables" on page 66.

The Design tab appears with Report Header, Page Header, Details, Page Footer, and Report Footer areas. A report is created by inserting and formatting items in each of these areas.

Note: To use additional database tables for a report and match them up on a record-by-record basis, click Database Expert on the Expert Tools toolbar, select the table(s), and then set up the links on the Links tab when it appears.

To work with report elements on the Design tab

1 Each of the default report areas contains a single section. To add additional sections, click **Section Expert** on the Expert Tools toolbar and use the Section Expert to add the desired sections.

Once you have added sections to an area, you can move, merge, and delete them in the Section Expert. See "Designing with guidelines" on page 187.

2 To turn the grid on or off, choose the **Options** command from the **File** menu and select or clear the Grid check box in the Design View area of the Layout tab when the Options dialog box appears. See "Using the grid" on page 186.

3 If you are working with the grid off and you want to use snap-to guidelines for positioning objects, click the top or left ruler wherever you want guidelines to appear.

Some further ways you can use guidelines:

- Drag a field to a guideline until it snaps to the guideline.
- Drag the guideline arrow to move the guideline (and any objects that are snapped to it).
- Drag the guideline arrow away from the ruler to remove the guideline. See "Designing with guidelines" on page 187.

4 If the Field Explorer dialog box is not visible, click **Field Explorer** on the Standard toolbar.

Expand the Database Fields node to display a list of fields. To speed the entry of multiple fields, this box will remain on-screen until you close it. This dialog box can be moved to a new location or resized, if you wish. See "Placing data on the report" on page 68.

5 Select the field(s) you want to appear on the report.

You can select and place them one at a time, or use the Shift-click combination to select a number of contiguous fields, or the Ctrl-click combination to select fields from the list at random. Drag and drop is also active. Place the fields in the Details section where you want them to appear.

When you place multiple fields, they appear in the same order that they appear in the Field Explorer dialog box. The program marks the position of each field with a rectangular frame. The characters in the frame indicate whether the field is text (XXX...), number (555...), currency ($555...), date (12/31/99), time (00:00:00), DateTime (12/31/99 00:00:00), or Boolean (T/F).

Note:

- The field names and field types can be viewed by selecting the Show Field Names check box on in the Options dialog box (Layout tab).
- The program automatically places field titles in the Page Header section unless the Insert Detail Field Headings check box is not selected in the Options dialog box (Layout tab).

- If additional Details sections are added to the report, field titles will only be placed in the Page Header section for fields in the Details A (the original) section of your report.

6 Once the objects are in place, you may want to adjust the report sections. To do this, right-click the shaded area to the left of the section ruler and use the shortcut menu that appears:
   - To expand a section to accommodate an additional line, choose the Insert Line command.
   - To have the program automatically align the objects in the section horizontally, choose the Arrange Lines command.
   - To reduce the size of a section to eliminate unnecessary white space above and below objects, choose the Fit Section command. See "Using white space between rows" on page 206.

7 To generate a report title, you must first enter the title in the Document Properties dialog box. On the **File** menu, click **Summary Info**. Enter a title in the **Title** text box of the Document Properties dialog box. Click **OK**.

8 Choose **Report Title** from the Special Fields list in the Field Explorer. A rectangular placement frame appears when you move the cursor over your report. Click once in the Report Header (RH) section to place the report title. The report title field contains the text that you typed in the **Title** text box of the Document Properties dialog box. See "Adding a title page to the report" on page 77.

9 To see how the results will print, click **Print Preview** on the Standard toolbar.

   To speed processing time while building a report, you can preview the report using only a small subset of the available data. To do this, go to the File menu, point to Print and then click Preview Sample. See "Preview tab" on page 62.

   In either case, the program takes you to the Preview tab. You can fine-tune a report in the Preview tab while viewing the results as actual report data. You can also close the Preview tab and continue working on the report in the Design tab.

To use other reporting features

1 If you want to format a field, change the placement or width of a field, or insert a summary, click the field to select it. Handles appear on the top, bottom, and sides of each selected field:
   - To change the placement of the field(s), use the mouse to drag the field placement frame to its new position.
   - To change the width of the field, use the mouse to drag the right or left handle.
   - Right-click the field to format or to summarize it. A shortcut menu appears listing commands for formatting and summarizing the field.

   Tip: Many font and formatting options are available on the formatting toolbar.

   Note: To apply formatting only under certain conditions, click Conditional Formula next to the formatting property in the Format Editor, and create a formula that defines those conditions. See "Working with conditional formatting" on page 208.

2 To create a formula that makes data calculations or comparisons, select **Formula Fields** in the Field Explorer.

Note: You can also click Formula Workshop on the Expert Tools toolbar to work with formulas, custom functions, SQL expressions, and so on.

3 Click **New**. The Formula Name dialog box appears. Enter a name for the formula and click **Use Editor**. The Formula Workshop appears with the Formula Editor active.

4 Enter the formula in the Formula Editor. Enter fields, operators, and functions by selecting them from their respective lists or by typing them in. You can check the formula syntax by clicking **Check**.

5 When you are finished editing, click **Close** to return to the Field Explorer dialog box.

6 Click **Insert to Report** to place the formula just like you would a database field. See "Using Formulas" on page 329.

7 To insert a subreport (a report within a report), click **Insert Subreport** on the Insert Tools toolbar and choose an existing report to import as a subreport or use the Report Creation Wizard to create a new subreport. See "Inserting subreports" on page 369.

If you want the records in a subreport to match up with the records in a primary report, click the Link tab of the Insert Subreport dialog box and specify the link when the tab appears. See "To link a subreport to the data in the primary report" on page 372.

8 To insert a cross-tab object in a report, click **Insert Cross-Tab** on the Insert Tools toolbar and set up the cross-tab in the Cross-Tab Expert when it appears. See "Cross-Tab Objects" on page 261.

9 To create a parameter field (a field that prompts you for a value whenever you retrieve data for a report), click **Field Explorer** on the Standard toolbar, then select Parameter Fields in the Field Explorer dialog box when it appears. Click **New** to set up a parameter field. Once created, you can insert the parameter field in a report like a database field or select it from the **Fields** list in the Formula Workshop.

Parameter fields can be used in reports (as title or label prompts), in selection formulas (as selection criteria prompts), and in formulas (for a variety of purposes including specifying sort fields). See "Parameter Fields" on page 349.

10 To add a chart, click **Insert Chart** on the Insert Tools toolbar. See "Charting" on page 217.

11 To add a map, click **Insert Map** on the Insert Tools toolbar. See "Mapping" on page 233.

12 To insert a spreadsheet, picture, or other OLE object that you can edit from within the Report Designer using the tools from the object's native application, choose **OLE Object** from the Insert menu. See "OLE" on page 251, and "Working with static OLE objects" on page 256.

13 To change the record sort order, click **Record Sort Expert** on the Expert Tools toolbar. The Record Sort Order dialog box appears. Highlight the field(s) you want to use for sorting the report data and the sort direction. See "Sorting single and multiple fields" on page 121.

14 To limit the report to specific records (for example, the records of California customers who have year-to-date sales greater than $10,000), click the first field on which you want your selection to be based and then click **Select Expert** on the Expert Tools toolbar. When the Select Expert appears, set up the record selection criteria.

15 To print the report, click **Print** on the Standard toolbar.

That's it! It is that easy to build a report.

Report Design Concepts

4

This chapter describes the basic concepts of report design and helps you decide what information you want to include in your report.

Basic report design

The purpose of this chapter is to suggest a structured approach to preparing a Crystal report. This approach includes the following elements:

- Deciding on the content of the report.
- Developing a prototype on paper.

This section has been designed to provide a conceptual understanding of the reporting process.

Deciding on the content of the report

Before you do anything else, you should outline the information you want the report to provide. The following sections provide a guide to making that outline.

Stating the purpose

What is the overall purpose of the report?

Reports are management tools. Their purpose is to help you quickly grasp the essential elements and relationships found in raw data, to help you make effective decisions. For a report to be effective, it has to present the correct data in a logical way. If it presents the wrong data, or if it presents the right data in a haphazard manner, the report may slow the decision-making process or may even encourage incorrect decisions.

A good starting place in the development of a report is to write out the purpose of the report in a sentence or two. The purpose statement helps you focus on your primary needs, and it gives the report both a starting point and a goal.

Here are some examples of purpose statements.

- The purpose of this report is to show monthly and year-to-date sales by sales representatives, compare this year's numbers to last year’s, and flag representatives whose sales figures do not meet company standards.
- The purpose of this report is to show sales activity for each item in inventory, and to suggest reorder quantities based on that activity.
- The purpose of this report is to calculate bowling averages and handicaps for each member of the bowling league.

Defining the purpose of the report before you start is a critical step in the overall process.

Who is going to read the report?

A single report is often used by many individuals. A detailed, company-wide sales report, for example, may be used by sales representatives, the regional sales manager, the national sales manager, and the Chief Operating Officer (COO).

These individuals will be interested in different aspects of the report:

- A sales representative will use the report to evaluate individual sales performance and compare this performance to that of other representatives in the region.
- The regional sales manager will use the report to evaluate regional representatives and compare the region's performance to that of other regions.
- The national sales manager will use the report to evaluate the performance of regional managers and compare overall sales to the current sales forecasts.
- The COO will use the report to evaluate the performance of the Vice-President of Marketing and the sales department as a whole, and to project such things as manufacturing needs and warehouse locations.

Since each user of the report has different interests, it is important to plan the report so it includes the information each user is looking for.

Determining the layout of the report

What is the report title going to be?

Write out a working title for the report. You may decide to change it later, but at least you will have a title to use when creating the prototype report.

What identifying information is needed in the header and footer?

You may wish to include the print date, information on who prepared the report, a block of text to describe the purpose of the report, the range of data covered, or something similar. If you are going to include such information, write it down so you can use it in preparing your prototype.

The information can come from a variety of sources, depending on the kind of information you plan to use.

- Information on who prepared the report might be drawn from individual data fields in the database table(s) used. If it is to be drawn from a database table, what table? Or, what combination of tables?
- A block of text can be created as a text object and placed anywhere on the report.
- Crystal Reports can generate information such as the print date or page numbers.

Finding the data

What data do you want to use in the report?

Do you know the type of database you are reporting from? Will you be reporting off a data file, SQL/ODBC, or a Dictionary?

If you do not know, ask the database administrator in your organization for help in setting up the database type and location of the data. For more information, see Accessing Data Sources in the online help.

Are you familiar enough with the data to find the necessary information? When looking for a Customer Contact name, can the field be found in a database table?

If not, your MIS professional, database administrator, or co-workers will have to help you become familiar with the data.

What specific data should appear in the body of the report?

The body should contain all the data needed to fulfill the statement of purpose you wrote for the report. It should also contain all of the data needed by the various users that you have identified.

This step requires you to look at the available database table(s). Crystal Reports allows you to combine data from different databases when you create reports, so you have a great deal of flexibility in your work.

- Much of the data in a typical report is taken directly from data fields. Which data fields will be used, and where are they located?
- Other data will be calculated based on data fields. Which data fields will be used in the calculations?
- Still other data will be placed directly into the report using text objects (headings, notes, labels, and so on).

Does the data exist or does it need to be calculated?

Some report information can be drawn directly from data fields (sales information, for example); other information will have to be calculated based on data field values (for example, sales commission, based on the relationship of sales to quota). In your planning, it can be helpful to segregate or flag data that needs to be calculated from that which can be used directly. See “Specifying formulas” on page 335.

What types of fields contain data?

You should take the time to get to know the data type for data fields that will be used in your calculations. Since formula functions and operators work with specific kinds of data, it is important to recognize the data type you are working with, before you start any calculations. For example, some functions require numeric data, while others work with only string fields.

Manipulating the data

Do you want the data organized into groups?

How? By customer? By date? By hierarchy? Or by other criteria? Crystal Reports provides several options for grouping data in a report. See "Grouping data" on page 122.

Do you want the data sorted based on record or group values?

Crystal Reports gives you both alternatives. See "Understanding sort options" on page 120.

Do you want the report to contain only specific records or groups?

Crystal Reports gives you the opportunity to base a report on all records in a given database, or on a limited set of records from the database. Crystal Reports can be used to select records based on simple date ranges or comparisons, or to create complex formulas to identify the records to be included. Take a few minutes to determine the records needed for the report and list the criteria to be used for selecting those records. See "Selecting records" on page 108.

Do you want to summarize the data?

Do you want to total, average, count, or determine the maximum or minimum value included in all the values in any column on the report?

Crystal Reports allows you to do this, and it also allows the grand total (or the grand total average, grand total count, and so on) to be placed at the bottom of the selected column. See "Calculating a percentage" on page 141 and "Selecting top or bottom N groups or percentages" on page 137.

What information should be flagged on the report?

You may want to call attention to some data by flagging it on the report. For example, non-moving inventory items are often flagged on inventory reports so they can be given special attention. You might want to flag each item that has shown no activity during the last month, during the last three months, or during some other defined period. To flag information, identify it and any conditions that will trigger the flagging.

How do you want information flagged?

You may want to flag items with an asterisk or some other symbol, or you may want a word to appear as a flag. In any case, you should write out flagging instructions so they are handy.

Crystal Reports gives you the opportunity to underline report elements, and change the font type, size, or color used for specific report items. It allows you to put borders around items and to draw lines and boxes (to break the report into sections), set off headings, and so on. All of these formatting tools can be used to highlight key data on a report. See "Formatting" on page 173.

Determining printing area characteristics

Each report area has its own printing characteristics. It is important to understand these characteristics because they affect when and how often different report objects get printed.

In what order will the areas print on the report?

Areas print in the order they appear on the Design tab (top to bottom). If there is more than one section in an area, the sections print in the order they appear. For example, if you have three Report Header sections, all three of those sections will print, in order, before the section(s) in the Page Header area begin to print.

How often do report objects print?

The way objects print will determine how you design your report. This will help you decide where to place charts, Cross-Tabs, and formulas to get specific results.

Report Header

Objects placed in the Report Header area print once, at the beginning of the report.

- Charts and Cross-Tabs placed in this area contain data for the entire report.
  Note: Both charts and Cross-Tabs can filter report data by using a Group Sort; in such cases, the data shown in the chart or Cross-Tab is a subset of the data for the entire report.
- Formulas placed in this area are evaluated once, at the beginning of the report.

Page Header

Objects placed in the Page Header area print at the beginning of each new page.

- Charts or Cross-Tabs cannot be placed in this section.
- Formulas placed in this area are evaluated once per page, at the beginning of each new page.

Group Header

Objects placed in the Group Header area print at the beginning of each new group.

- Charts and Cross-Tabs placed in this area contain data just for the group.
- Formulas placed in this area are evaluated once for each group, at the beginning of the group.

Details area

Objects placed in the Details area print with each new record.

- Charts or Cross-Tabs cannot be placed in this area.
- Formulas placed in this area are evaluated once for each record.

Group Footer

Objects placed in the Group Footer area print at the end of each group.

- Charts and Cross-Tabs placed in this area contain data just for the group.

- Formulas placed in this area are evaluated once for each group, at the end of the group.

Report Footer

Objects placed in the Report Footer area print once at the end of the report.

- Charts and Cross-Tabs placed in this area contain data for the entire report.
  Note: Both charts and Cross-Tabs can filter report data by using a Group Sort; in such cases, the data shown in the chart or Cross-Tab is a subset of the data for the entire report.
- Formulas placed in this area are evaluated once, at the end of the report.

Page Footer

Objects placed in the Page Footer area print at the bottom of each page.

- Charts and Cross-Tabs cannot be placed in this area.
- Formulas placed in this area are evaluated once per page, at the end of each new page.

Developing a prototype on paper

While a paper prototype is useful regardless of your level of expertise with Crystal Reports, it is particularly valuable when you are first learning the program. With the paper prototype in hand, you can put your full effort into learning and using the commands, rather than into trying to design and learn at the same time.

To design a paper prototype

1. Get the same size paper you will be using for the finished report.
2. Position the title and other descriptive header information, using boxes or lines to represent report elements.
3. Position the footer information.
4. Review the page layout for balance.
5. Look at the information you intend to include in the body of the report:
   - Count the number of fields being used and estimate the appropriate spacing between fields.
   - Use rectangles to pencil in the fields within the estimated spacing.
   - Change the spacing if you need to.
   - Decide on a logical sequence for presenting the data in the body of the report.
   - Label the fields to indicate that sequence.
6. Use small boxes to indicate group values and totals.
7. Place random flags in the column where you want flags to appear.

8. Darken any elements you want highlighted to make them stand out from the rest of the prototype.
9. Review the finished product for layout and balance, and make changes as needed.

Introduction to Reporting

5

This chapter shows you the basics of creating a report in Crystal Reports. First you'll learn about the report design environment, then about how to select database tables, place objects on a report, and how to sort, group, and total your report data.

Report creation options

Each time you create a new report, you have three options:

- Use a Report Creation Wizard.
- Use another report as a model.
- Create a report from scratch.

You will probably use each option at some time.

Report Creation Wizards

The Report Creation Wizards help create reports as quickly as possible and many new users and developers alike prefer to create the majority of their reports using them. All you have to do is choose the wizard that most closely matches your report type. The wizard walks you through the process of creating reports step-by-step.

Another report

To build a new report based on one that already exists, another report can be used as a model. Open the report you want to use in this way by selecting the Open an Existing Report option and save it to a new file using Save As (found on the File menu). This method is useful to:

- Create a new report with a different grouping or different record selection than that of an existing report.
- Reconstruct a report based on an earlier time period using the same report structure used today.
- Create an entirely new report based on a set of databases that are linked in another report. You can create a report and delete the fields without disturbing the underlying links. Then, without relinking, you can build all your new reports based on this report.

Crystal Reports also lets you format a report by applying a template. See "Using a template" on page 174 for more information.

New report

The As a Blank Report option is used to create a report from scratch. This is useful when you want the full flexibility and control of building a report from the ground up, or when a report type is different from the many report types available in the wizards.

The As a Blank Report option was chosen for the "Quick start for new users" on page 20, because the process of creating a report from scratch most fully illustrates the basics of reporting.

Choosing data sources and database fields

Crystal Reports makes it simple to select data sources and database fields by providing easy-to-use functionality in the Database Expert dialog box and the Field Explorer dialog box. Each of these dialog boxes uses the familiar Windows tree structure to allow you to navigate through the possible choices.

Note: Crystal Reports supports Unicode by converting data from non-Unicode databases as it accesses it (this data conversion happens within Crystal Reports; the data in your database is not affected). Because Crystal Reports offers Unicode support, you can display different languages in the same report.

The Database Expert

The Database Expert provides an integrated tree view of all data sources you can use with Crystal Reports. In the Database Expert, you can select from the following as a data source for your report:

- A currently connected data source.
- An SQL command that has been saved to the Crystal Repository.
- A data source that has been added to your Favorites folder.
- A recently accessed data source (the Database Expert automatically maintains a list of such data sources for you).
- An existing data source (for example, a data file residing locally, or an ODBC data source that has already been set up).

In the Database Expert, you can also specify links between database tables when you have selected more than one table for your report.

Data tab

Tree View

The Data tab of the Database Expert shows a tree view of possible data sources you can select when creating a report. The tree—in the Available Data Sources list—is made up of folders for:

- Current Connections
- Repository
- Favorites
- History
- Create New Connection

The Create New Connections folder contains subfolders for many popular data sources. Among these, you'll find:

- Access / Excel (DAO)
- Crystal Queries
- Database Files

- Dictionary / Infoview
- ODBC (RDO)
- OLAP
- OLE DB (ADO)

Note: The data source options available in the Create New Connections folder depend on the data access components selected during installation.

For a brief description of each of these folders and subfolders, see "Selecting the data source" on page 65.

Shortcut Menu

You can right-click any item in the Available Data Sources list of the Database Expert to see a shortcut menu with the following options:

- **Add to Report**
  Use this option to add a table or stored procedure to your new report. This option is also available by clicking the > arrow on the Database Expert.
- **Add to Favorites**
  Use this option to add a selected data source to the Favorites folder.
- **Remove from Report**
  Use this option to remove a table or stored procedure from your report. This option is also available by clicking the < arrow on the Database Expert.
- **Properties**
  Use this option to obtain detailed information on the selected item.
- **Rename Favorite**
  Use this option to rename a data source in the Favorites folder.
- **Delete Favorite**
  Use this option to remove a data source from the Favorites folder.
- **Remove from repository**
  Use this option to delete an existing SQL command from the Crystal Repository.
- **Rename repository object**
  Use this option to rename an existing SQL command in the Crystal Repository.
- **Refresh**
  Use this option to refresh the list of available data sources in the Database Expert.

Links tab

Database tables are linked so records from one database match related records from another. For example, if you activate a Suppliers table and a Product table, the databases are linked so that each product (from the Product table) can be matched up with the supplier (from the Supplier table) that made the product.

The majority of reports will probably require data from two or more tables, so linking will be necessary. The process of linking is made easy by using the Links tab of the Database Expert. See "Linking tables" on page 385.

The Field Explorer

Use the Field Explorer dialog box to insert, modify or delete fields on the Design and Preview tabs of Crystal Reports. To see the Field Explorer, select the Field Explorer command from the View menu.

Tree View

The Field Explorer shows a tree view of database fields and special fields that you can add to your report. It also shows formula fields, SQL expression fields, parameter fields, running total fields, and group name fields that you have defined for use in your report.

Fields that have already been added to the report, or fields that have been used by other fields (such as formula fields, groups, running total fields, summaries, and so on) have a green check mark next to them.

Toolbar and Shortcut Menu

The Field Explorer's toolbar provides buttons with tool tips and hot-key combinations. You can right-click any item in the tree view to bring up a shortcut menu.

The toolbar offers these functions:

- **Insert to Report**
  Use this option to add a field to the report. You can insert more than one field at a time by selecting multiple fields, right-clicking, and choosing Insert to Report.
  Alternatively, to insert a field, you can drag and drop it in the Design or Preview tabs.
- **Browse**
  Use this option to browse data for a database field, formula field or SQL expression field.
  Note: On the shortcut menu, this option is called Browse Data.
- **New**
  Use this option to create a formula field, SQL expression field, parameter field or running total field.
- **Edit**
  Use this option to modify an existing formula field, SQL expression field, parameter field or running total field.
- **Rename**
  Use this option to modify the name of an existing formula field, SQL expression field, parameter field or running total field.
- **Delete**
  Use this option to remove a formula field, SQL expression field, parameter field or running total field. You can also select multiple fields, right-click, and choose Delete to remove them all at once.

In addition to the functions available on the toolbar, the shortcut menu offers these functions as well:

- **Move Parameter Up (or Down)**
  Use these options to change the order of parameter fields.
- **Show Field Type**
  Use this option to see the field type (string, number, and so on) when you're looking at a list of database fields. The length of string fields is included in brackets at the end of their names.
- **Refresh**
  Use this option to refresh the list of available fields in the Field Explorer.

Group Name Fields

You can insert an existing Group Name field shown in the Field Explorer by right-clicking it and selecting Insert to Report. Unlike a formula field, parameter field or running total field, however, you cannot create a Group Name field through the Field Explorer. (A Group Name field is created when you insert a group.)

The Report Explorer

The content of the Report Explorer represents the content of the report in a tree view. The root node is the report itself, while the first-level nodes represent the report's sections. Within each section, the report's fields and objects are listed. Any item you select in the tree view will be selected in the report (in either Design or Preview modes).

You can modify report fields and objects by selecting them in the Report Explorer. When you right-click the selected item, you see a menu that contains the actions you can carry out on the item. For example, if you right-click a text object, the menu will contain the options to edit the text, to format its font or color, and so on. Likewise, if you right-click a section node, the menu will contain the options to hide, suppress, format and so on.

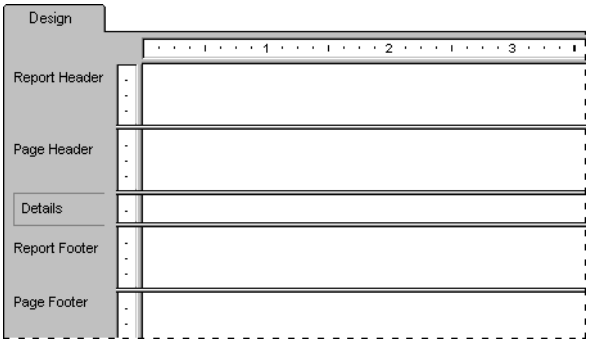
Note: You cannot add additional fields or report objects when using the Report Explorer, but you can delete them.

You can select multiple fields for formatting by using Shift-click or Ctrl-click.

About the report design environment

Design tab

When working with Crystal Reports, you will probably use the Design tab more than any other part of the program.



The Design tab is the place you do most of the initial work when creating a report. It designates and labels the various sections of the report. You can do the initial formatting, place objects in the sections where you want them to appear, specify sorting, grouping, and totaling needs, and so forth.

The Design tab provides a very efficient environment for designing a report because you work in the tab with data representations, not with data itself. When a field is placed on the report, the program uses a frame to identify the field on the tab; it does not retrieve the data. Thus, you can add and delete fields and other objects, move them around, set up complex formulas, and more, without tying up the computer or network resources needed to gather the data.

The report created in the Design tab is a kind of virtual report; it has the structure and instructions for creating the final report, but it is not the report itself. To turn the Design tab report into a final report or into a report that you can fine-tune, you "just add data." You do this whenever you preview the report, print it, or output it in any other way. The actual data will now appear in the report.

Design tab areas

When you first begin creating a report, Crystal Reports automatically creates five areas in the Design tab.

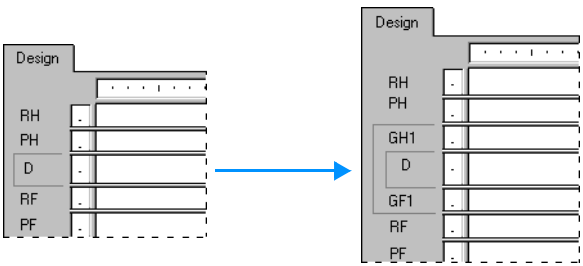
- **Report Header**
  This section is generally used for the report title and other information you want to appear at the beginning of the report. It can also be used for charts and cross-tabs that include data for the entire report.

- **Page Header**
  This section is generally used for information that you want to appear at the top of each page. This can include such things as chapter names, the name of the document, and other similar information. This section can also be used to display field titles above the fields on a report.
- **Details**
  This section is used for the body of the report, and is printed once per record. The bulk of the report data generally appears in this section.
- **Report Footer**
  This section is used for information you want to appear only once at the end of the report (such as grand totals) and for charts and cross-tabs that include data for the entire report.
- **Page Footer**
  This section usually contains the page number and any other information you want to appear on the bottom of each page.

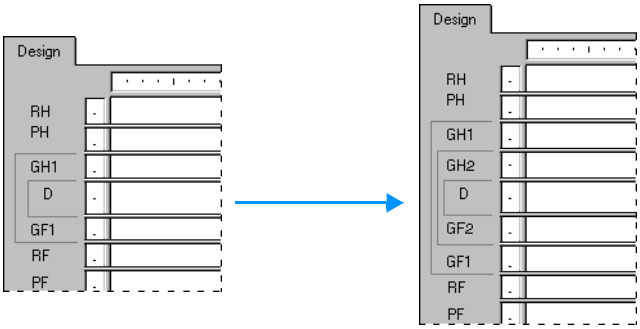
If a group, summary, or subtotal is added to the report, the program creates two additional sections:

- **Group Header**
  This section typically holds the group name field, and can be used to display charts or cross-tabs that include data specific to the group. It is printed once at the beginning of a group.
- **Group Footer**
  This section generally holds the summary value, if any, and can be used to display charts or cross-tabs. It is printed once at the end of a group.

When a group, summary, or subtotal is added, the Group Header area appears directly above the Details area and the Group Footer area appears directly below the Details area.



If you set up additional groups, the program creates new group areas between the Details area and the existing Group Header and Group Footer area(s).



Like the original areas, each of these newly added areas can contain one or more sections. By default, they each contain a single section.

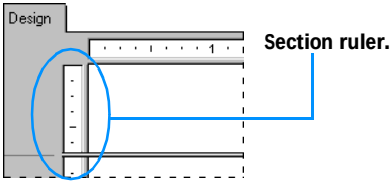
Identifying and working with areas and sections

By default, each area contains only a single section. The name for that section appears directly to the left of the section. If you have multiple sections in an area, the sections are designated as a, b, c, and so forth.

Note:

- Initials, such as RH, PH, D, PF, RF, and so on, are used to identify each section if you have selected the Short Section Names check box in the Design View area of the Options dialog box.
- If you right-click the shaded area containing a section name, a shortcut menu appears with section-specific options.

If you have selected the Show Rulers options on the Layout tab of the Options dialog box, the program displays a section ruler immediately to the left of each section. The section ruler is used to add, remove, and move guidelines, and to provide a visual reference when you are placing objects. See "Designing with guidelines" on page 187.



Whenever a new section is added, the program creates a ruler for that section. See "Using multiple sections in reports" on page 163.

Other Design tab capabilities

There are several other capabilities built into the Design tab. With the Design tab, you can:

- Resize a section by dragging its boundary. See "Resizing a section" on page 162.
- Split a section (create two sections from one) by clicking its left boundary. See "Splitting a section" on page 161.
- Add horizontal and vertical guidelines by clicking the rulers. See "Designing with guidelines" on page 187.
- Zoom in and out on a report at any magnification from 25% to 400% of the original size. See "Using the zoom feature" on page 76.

Preview tab

To preview a report before printing it, click Print Preview on the Standard toolbar.

The program gathers the data, makes the necessary calculations, and displays the report in the Preview tab. With the data in place, you can review the spacing and formatting of your report and see the actual results of all your summaries, formula calculations, and record and group selections.

In true WYSIWYG (What You See Is What You Get) fashion, you can work directly on this live data, fine-tuning it until the report has the exact look you want.

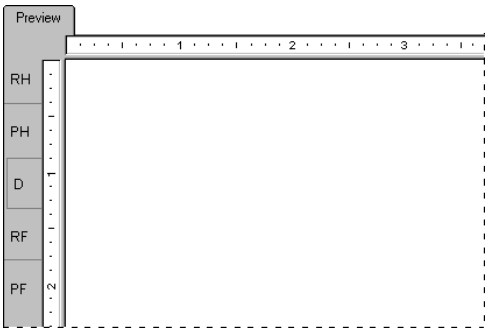
The program works with data in the following manner:

- The first time the Preview tab is used, it retrieves data from your underlying data source(s) and saves it with the report (unless you have set up the program not to save data).
- From that point on, the program uses the saved data whenever you preview the report unless you specifically refresh it or add a field that requires the program to retrieve new data.

Crystal Reports provides two views for previewing a report:

- Standard view
- Group Tree view

Standard view



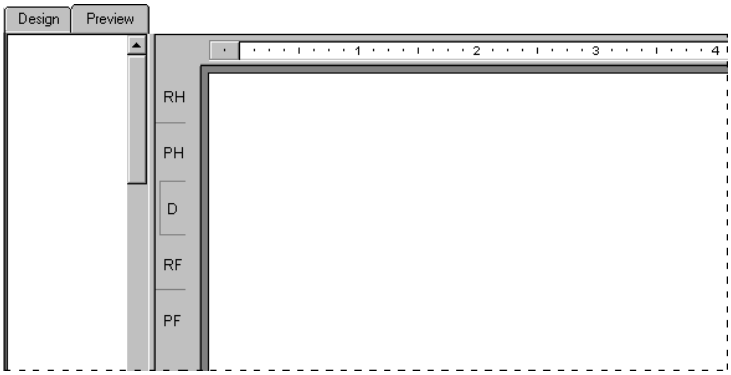
In standard view, the report is displayed a page at a time. Using the navigation buttons in the Preview tab, you can move to the beginning or end of the report, or you can move backward and forward through the report one page at a time. For shorter reports or reports in which you're primarily interested in seeing the “bottom line” totals, the standard view provides all of the functionality you need.

The Data Age indicator

The Data Age indicator indicates the date the data was last refreshed or initially retrieved, whichever is the most recent. If the data was initially retrieved or refreshed today, it indicates the time it happened.

1/16/2002 11:32

Group Tree view



The Group Tree view can be shown or hidden using Toggle Group Tree on the Standard toolbar.

The Group Tree view presents a split screen:

- The right pane displays the report.
- The left pane displays a high level outline of the report, showing the hierarchy of groups and subgroups in a familiar tree format.

The Group Tree normally displays the names of the groups and subgroups you created in your report. You can, however, customize these names using the Options tab of the Insert Group or Change Group Options dialog box. For more information on customizing Group Names in the Group Tree view, see “Grouping data” on page 122.

When you click the tree node for the group that interests you, the program jumps immediately to the part of the report that contains the information for that group. For longer reports or reports in which you wish to jump back and forth between different groups, the Smart Navigation features of the Group Tree view make your work extremely efficient.

Comparisons with the Design tab

You have the same formatting capabilities in the Preview tab as you do in the Design tab. Menus (both menu bar and shortcut menus) and toolbars remain active, providing essentially the same functionality you have when working with a report in the Design tab. However, when you are making numerous changes, it is quicker to make the changes in the Design tab. Some additional things to consider are:

- The Design tab and Preview tab are tied together internally. Any changes made in one are reflected in the other.
- The Preview tab has a single vertical ruler at the left of the tab rather than the individual section rulers seen in the Design tab. The functionality of the ruler is the same.
- The Preview tab identifies report sections in the shaded area to the left of the data. With a quick look you can tell which report section the data is printing from. While section names appear only once in the Design tab, they print each time a section prints in the Preview tab.
- The Record counter, the Data Age indicator (see "The Data Age indicator" on page 63), and the Page Forward / Page Back controls (see "Preview tab" on page 62), are all active in the Preview tab.
- The Preview tab highlights every value when you select a field; whereas, only the field frame is highlighted in the Design tab.

Working in the Preview tab has a different feel from working in the Design tab.

Each field in a database contains dozens, hundreds, or even thousands of values, depending on the number of records in the database. When you place a field in the Design tab, a single field frame represents all those values. When you highlight the field, sizing handles appear on the frame and the frame changes color.

In the Preview tab, however, you are working with the actual data. Instead of a field frame representing many field values, the values themselves appear. Some additional things to consider are:

- When you highlight a field or formula field value, you are actually selecting every value in the field:
  - The program places a sizing frame around the specific value you select.
  - It highlights every other value in the field.
- Likewise, when you select a summary value, you are actually selecting all the related summary values:
  - The program places a sizing frame around the specific value you select.
  - It highlights all the related summary values.

Aside from the differences in appearance, the process of building and modifying a report is the same in both the Design tab and the Preview tab. You should find it easy to work with your reports in both places.

Creating a new report

Selecting the data source

After deciding which option you want to use for creating your report (see "Report creation options" on page 54), the next step is to select a data source to use.

Most data sources can be chosen through the Database Expert dialog box. The Database Expert appears when you create a report from scratch using As a Blank Report, or when you choose Database Expert from the Database menu.

Note: You also select a data source in the Report Creation Wizards. The Data screen in all of the Report Creation Wizards, except the OLAP Report Creation Wizard, is much like the Database Expert dialog box.

To select a data source

1. Choose **Database Expert** from the **Database** menu.

   The Database Expert dialog box appears.

2. Use the tree view in the Available Data Sources list of the Data screen to select your data source:
   - **Current Connections**

     This folder shows a list of data sources you are currently connected to.
   - **Repository**

     This folder shows a list of data source connections that have been stored in the Crystal Repository.
   - **Favorites**

     This folder shows a list of data sources you commonly use and have maintained in your Favorites list.
   - **History**

     This folder shows a list of data sources you have used recently. The last five data sources used are displayed.
   - **Create New Connection**

     This folder shows subfolders for various data sources you can connect to.

   Some popular choices in the Create New Connection folder are described here:
   - **Access/Excel (DAO)**

     This option lets you connect to a supported database type (Access, dBASE, Excel, Lotus, and so on). You can create a new connection using the Make New Connection option.
   - **Crystal Queries**

     This option lets you browse for a query (.qry) file.
   - **Database Files**

     This option shows a list of standard PC databases that reside locally. You can use Find Database File to browse for a PC database using the Open dialog box.

- **Dictionary/Infoview**
  This option lets you browse for a dictionary (.dct or .dc5) or infoview (.civ) file.
- **ODBC (RDO)**
  This option shows a list of ODBC data sources you have already configured for use.
- **OLAP**
  This option opens the Crystal OLAP Connection Browser so you can choose an OLAP cube as a data source.
- **OLE DB (ADO)**
  This option shows a list of OLE DB providers you have already configured for use. You can also specify a Microsoft Data Link file to use.
- **More Data Sources**
  This option shows a list of other data sources that can be accessed through native drivers.

Note: The data source options available in the Create New Connections folder depend on the data access components selected during installation.

Adding tables

After selecting the data source, you can add one or more tables to base your report on.

To add a table

1. Choose **Database Expert** from the **Database** menu.
   The Database Expert dialog box appears.
2. On the **Data** tab, search for the database you want to use in your report. See "Selecting the data source" on page 65.
3. In the Available Data Sources list, select the table you want to add to your report and click the > arrow to add it to the Selected Tables list.
   You can insert more than one field at a time by selecting multiple fields, right-clicking, and choosing Insert to Report. You can also drag and drop fields to the Selected Tables list.

Linking multiple tables

If the report contains data from two or more database tables, they need to be linked at this point when creating reports.

Note: It isn't necessary to link tables in reports created from a query or command because any links required by the data have already been processed.

To add and link multiple tables

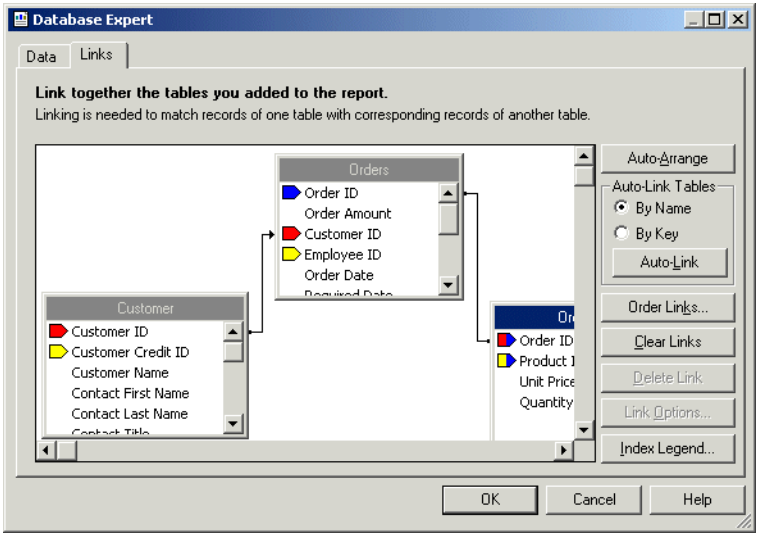
1. Choose **Database Expert** from the **Database** menu.
   The Database Expert dialog box appears.

2 On the Data tab, select the tables you want to add to your report. See "Adding tables" on page 66.

The Links tab appears in the Database Expert.

3 Click the **Links** tab to display the databases currently available for linking.

Crystal Reports automatically links tables by name or key when possible.



4 To create links manually, drag a field from one table to a field in another table. If successful, a link line is created. If unsuccessful, a message is issued.

Note: You can link tables by table name or by foreign key information.

5 If you've deleted links and you want to recreate them automatically, click **Auto-Link**.

6 Click **OK** when finished.

Note: When manually creating links, the field you are linking "to" must be of the same data type as the field you are linking from. When a native connection is used, the field you are linking "to" does not have to be indexed. For more information, see "Indexed tables" on page 383.

The Database Expert closes, and you are returned to your report. The linked databases are now available for use in your report. If you are not satisfied with the link, you can modify it using the Links tab of the Database Expert.

Related topics

"Understanding Databases" on page 379

"Linking options" on page 394

Placing data on the report

Placing data on a report is a very important task. You need to know what type of data should be placed on the report and where on the report it should be placed.

Database fields

Much of the data placed on a report are database fields, displaying data as it is stored in the database. For example, in the "Quick start for new users" on page 20, the Customer Name, City and Country fields are placed on the report. Normally, database fields will be placed in the Detail section, but under certain circumstances, they will be placed in other sections of the report.

To insert a database field

1. On the Standard toolbar, click **Field Explorer**.

   The Field Explorer dialog box appears. To speed the report building process, this dialog box remains on-screen until you close it. You can move this dialog box wherever you wish.
2. Expand the **Database Fields** folder to see all the tables chosen from the database(s).
3. Expand the individual tables to see all the fields they contain.
4. Select the field you want to appear in the report.
5. Click **Browse** to review the values in the selected field.
6. Click **Insert to Report** to place it in the report.

Formula fields

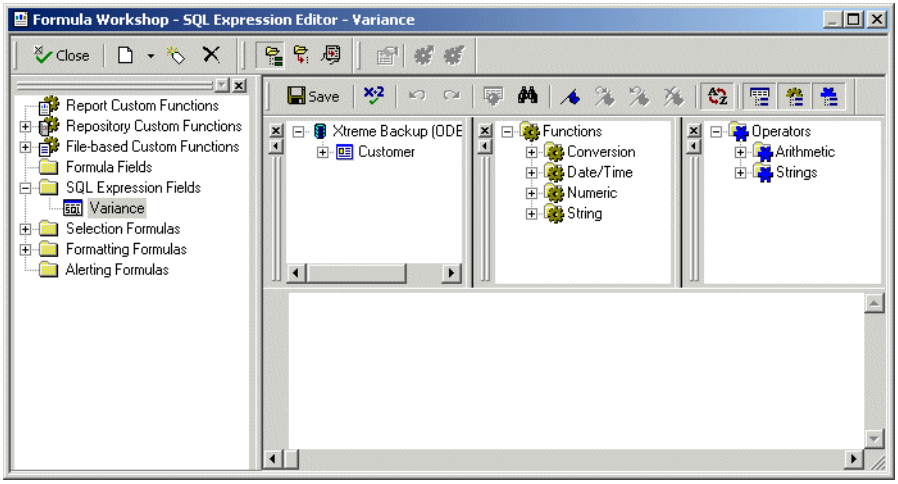
To display data that is a calculated value, you need to create a formula field and place that formula field on the report. For example, if the database only stores the order dates and ship dates for orders but you need to display the number of days it takes to ship the order, you must create a formula field that will calculate the number of days between ordering and shipping. This is just one example of the use of formula fields. See "Using Formulas" on page 329, for an introduction to formulas.

SQL Expression fields

SQL expressions are like formulas, but they are written in Structured Query Language (SQL), not in the Crystal Reports formula language. An SQL expression can be used to query the database for specific sets of data. You can sort, group, and select based on SQL expression fields.

To create an SQL Expression field

1. On the Standard toolbar, click **Field Explorer**.
   The Field Explorer dialog box appears.
2. Scroll down to **SQL Expression Fields** and highlight it. Click **New**.
   The SQL Expression Name dialog box appears.
3. Enter a name in the **Name** box, and then click **OK**.
   The Formula Workshop appears with the SQL Expression Editor active.



4. Type the expression in the SQL Expression Editor.
   Note: For an overview of the formula language, see "Formula components and syntax" on page 333.
5. Click **Save**.

Parameter fields

To prompt the user of a report to enter information, create a parameter field. Think of a parameter as a question that the user needs to answer before the report is generated. The information users enter, or the way they respond, determines what appears in the report. For example, in a report used by salespeople, there might be a parameter that asks the user to choose a region. The report would return the results for the specific region, instead of returning the results for all of the regions. See "Parameter Fields" on page 349 for an introduction to parameter fields.

Running total fields

To display a total that evaluates each record and provides a running sum of all the values in a field (or all the values in a certain set of values), a running total field needs to be created and placed in the report. If the first three values in a field were 2, 4, and 6, a running total would print 2, and then 6 (the sum of 2 + 4), and then 12 (the sum of 2 + 4 + 6). See "Running Totals" on page 147, for an introduction to running totals.

Special fields

To display information such as Page Numbers, Print Date, and Report Comments use the commands in the Special Fields tree view of the Field Explorer dialog box.

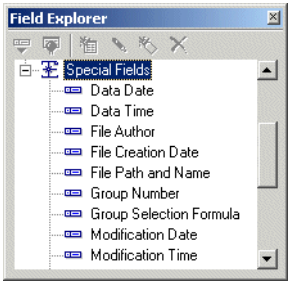
To insert a special field

Crystal Reports allows you to easily insert Page Number, Record Number, Group Number, Print Date, and Total Page Count fields, among others, into your report.

1. On the Standard toolbar, click **Field Explorer.**

   The Field Explorer dialog box appears.

2. Scroll down to **Special Fields** and expand it by clicking.



3. Choose a command from the Special Fields list to insert in the report.

   Each special field is inserted into the report as an object. An object frame appears. You can now place it on the report.

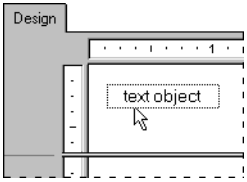
Note: To change the formatting of an inserted object, click the object to select it and click Format on the Expert Tools toolbar. The Format Editor appears where you can make the desired changes. See "Formatting" on page 173.

Text objects

Text objects are used in reports for a multitude of purposes. They are a powerful way of inserting titles; labeling summaries and other data on your report; and of easily combining database fields. For example, in the "Quick start for new users" on page 20, text objects are used to easily display the two contact name database fields as one object, to insert a column heading for the concatenated contact name, and to insert a title in your report.

To insert a text object

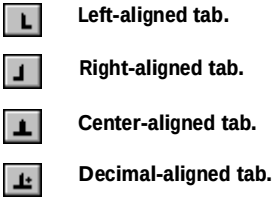
1 On the Insert Tools toolbar, click **Insert Text Object**. An empty object frame appears.

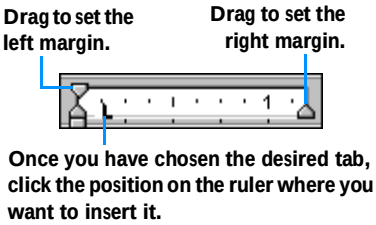


2 Position the text object where you want it to appear in the report.

Click once on the border of the text object to select it for resizing and moving.

Double-click inside the text object to select it for editing. The Design tab ruler changes to a text object ruler sized to the length of the selected object. To the left, a tab indicator appears. By clicking the tab indicator, you can cycle through the four tab options available.





The ruler allows you to add indents and align text within the text object.

Note: When you first insert the text object into the report, the object is automatically selected for editing.

Picture fields

When designing reports, there will be times when you want to include a picture. For example, you may wish to put a company logo in the report header.

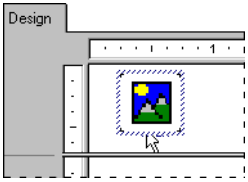
To insert a picture

1 On the Insert Tools toolbar, click **Insert Picture**.

The Open dialog box appears.

2 Select the desired picture file from the file list and click **Open** to return to the report.

An object frame appears with the picture inside, ready to be positioned.



3 Position the picture object where you want it to appear in the report and left-click once.

BLOB (Binary Large Object) fields

A BLOB field is a database field whose data consists of Binary Large Objects—such as bitmap graphics, images, OLE objects, metafiles, and so on. Inserting a BLOB field into your report allows you to access these binary objects as you would other data types.

In general, Crystal Reports allows you to access BLOB fields containing:

- Device-independent bitmaps (DIB).
- JPEG, TIFF, or PNG images.

In addition, if your data resides in a Microsoft Access database, then Crystal Reports enables you to report on BLOB fields containing OLE 1 and 2 objects and metafiles.

You insert BLOB fields as you would any other database field. For details, see "To insert a database field" on page 68.

A BLOB field object differs from other database field objects in that it offers options to control cropping, scaling, and sizing—just like an inserted picture or OLE object. To access these options, right-click the BLOB field object, select Format Graphic from the shortcut menu, and click the Picture tab.

Note: To retain the ability to activate and edit an OLE object with its server application, you should insert the object into your report independently, either by linking or by embedding, rather than inserting it in a BLOB field. For more information about inserting linked and embedded OLE objects, see "Inserting OLE objects into reports" on page 253.

Hyperlink fields

You can select a report object on the Design or Preview tab and create a hyperlink to another location.

The hyperlink is saved with your report and is available to other users as a way of viewing additional information.

Note: Crystal Reports also lets you create hyperlinks with Relative URLs, so your web reports retain independence from their location on any particular server.

To create a hyperlink field

1. Select a report object on the **Design** or **Preview** tab.
2. On the **Format** menu, click **Hyperlink**.

   The Format Editor dialog box appears with the Hyperlink tab active.
3. Click the type of hyperlink you want to create.

   The available types in the upper area are:

   - **No Hyperlink**

     This is the default option. There is no hyperlink associated with the selected report object.
   - **A Website on the Internet**

     Select this option if you want your report object to be linked to a static web address. Use the formula button to create a URL based on a field value. For example, you might enter the following formula if your Customer Name field contained information that would create a series of meaningful URLs:

     ```
     "http://www." + {Customer.Customer Name} + ".com"
     ```
   - **Current Website Field Value**

     Select this option if you want the program to create a hyperlink out of the field you selected. The field must be stored as a proper hyperlink in your data source.
   - **An E-mail Address**

     Select this option if you want to create a “mailto” address from the field you selected. Use the formula button to create an address based on a field value.
   - **A File**

     Select this option to create a hyperlink to a file on a specific computer or networked computer. Use the formula button to create a file path based on a field value.
   - **Current E-mail Field Value**

     Select this option if you want the program to create an email hyperlink out of the field you selected. The field must be stored as a proper email address in your data source.

   The available types in the DHTML Viewer Only area are:

   - **Report Part Drilldown**

     For information about how to use this option, see “The Report Part Drilldown option” on page 308.
   - **Another Report Object**

     For information about how to use this option, see “The Another Report Object option” on page 310.

   Note: Not all hyperlink types are available at all times. The object you select and its location on the report determine which types are available.

4 After you have chosen a hyperlink type, enter the appropriate hyperlink information (the URL of a web site, for example).

5 Click **OK** when you are finished.

The hyperlink is inserted as appropriate. Click it on the report to go to the web site, to send an e-mail, and so on. For information about how to use the hyperlink types in the DHTML Viewer Only area, see "Setting up navigation" on page 307.

Formatting data

At this point in creating a report, you may want to do some basic formatting. Perhaps you would like to change the font size and style of a text object used as a title. Or, if you have a number field, such as a sales figure, you might want to place a dollar sign before the number or change the number of decimal places displayed.

For example, in the "Quick start for new users" on page 20, you format the title, add a text object to identify the Contact Name information, and insert the company logo. See "Formatting" on page 173.

Record selection

Record selection, the task of paring down the data to include only the data required for your report, is a crucial step in report creation. You will rarely want a listing of all the information in a database. Most often you will be interested in only the sales in a given time period or for a certain product, an so on. For example, a sales report may be designed to only include sales from one product line for the last calendar month.

The sample data used for the "Quick start for new users" on page 20, has information from both United States and international customers. Record selection is used to create a report listing only customers in the United States. See "Record Selection" on page 107 and "Grouping data" on page 122.

Grouping, sorting, and summarizing data

Once a basic report is created, you will want to organize the data by grouping related information, sorting individual records, summarizing, subtotaling, and grand totaling.

Grouping records

To organize the data, you may want to group related data together. For example, in the "Quick start for new users" on page 20, after grouping the Customer List by region, you will divide the list into regional groups. That way, a sales manager for the California region could quickly locate the California group and focus exclusively on the customers within that region. See "Grouping data" on page 122.

Sorting records

Crystal Reports allows you to specify the order in which you want the records on your report displayed. For example, after grouping in the "Quick start for new users" on page 20, you sort the records within each region in alphabetic order by Customer Name. Many of your reports will use some type of sorting. Depending on the report, you will sort the records in a list or sort in conjunction with grouping. See "Sorting single and multiple fields" on page 121, and "Sorting records within groups" on page 125.

Summaries, subtotals, and grand totals

Many reports use some sort of totaling. For example, in a North American sales report grouped by state, you might want to calculate the total dollar amount sold in each state. You do this by creating a subtotal on the sales field. Summaries are also used at the group level, allowing you to calculate averages, counts, and other group (aggregate) values. For example, in a sales report you may want to calculate an average of sales per state (average summary on the sales field) and calculate the number of products sold in the state (distinct count of the product name field).

Using the drill-down option on summarized data

You can drill down on your data, to show the data behind individual groups, using the Drill-down cursor. See "Sorting, Grouping, and Totaling" on page 119.

Drill-down cursor

Crystal Reports allows you to drill down on group or summary information in the Preview tab in both the Standard and the Group Tree view (See "Standard view" on page 62 and "Group Tree view" on page 63). When you position the cursor over any summary value that you can drill down on, the program displays a Drill-down cursor.

Tip: Group headers appear on the Drill-down tab just as they do in the main report itself.

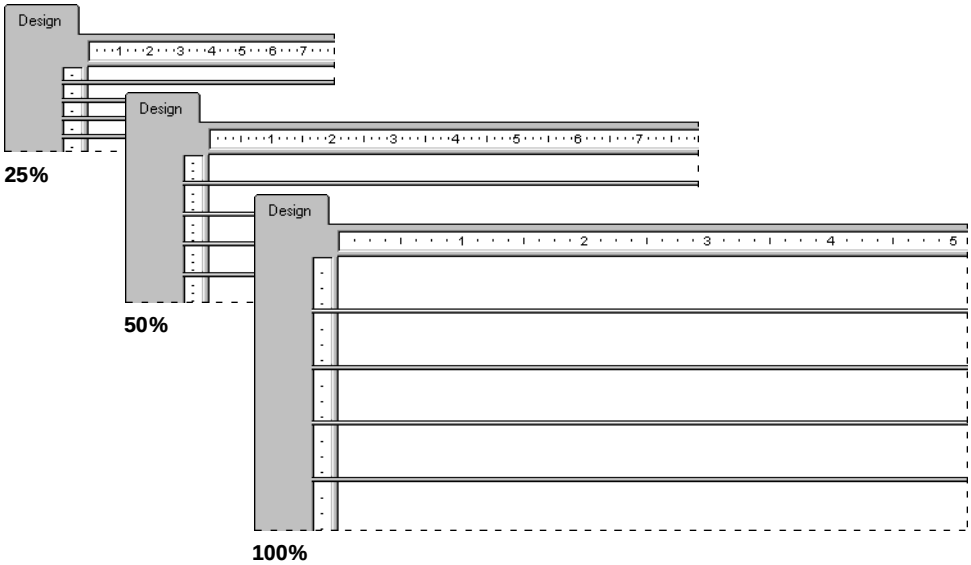
If you then double-click, the program reveals the details behind that specific summary value. For example, if the Drill-down cursor becomes active over the city summary, you can double-click to see the details behind that summary.

- If you have only a single summary, you can look at the summary or at the data from the individual records that are summarized.
- If you have multiple summaries, you can look at the summaries behind summaries (the city summaries that make up the region summaries, for example), or at the data from the individual records that are summarized.

Using the zoom feature

You can easily zoom in on a report. You can choose any magnification from 25% to 400%. This feature is active in both the Design and Preview tabs.

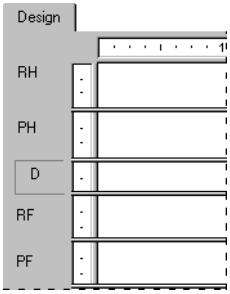
To zoom the report in or out, set the zoom level in the Zoom Control box found on the Standard toolbar.



It is helpful to view reports at low magnifications in order to get an overall picture of the layout of your report. Views at higher magnifications focus attention on the details of the report.

Inserting page headers and footers

You can use page headers and footers by placing the information in the Page Header or Page Footer sections of the Design tab.



- Information to appear only on the first page of the report goes in the Report Header (RH).
- Information to appear only on the last page of the report goes in the Report Footer (RF).
- Information to appear at the top of every page goes in the Page Header (PH).
- Information to appear at the bottom of every page goes in the Page Footer (PF).

Text, fields, or formulas can be used in these sections just as in the Details section.

Related topics

"Creating footers after the first page" on page 211

Adding a title page to the report

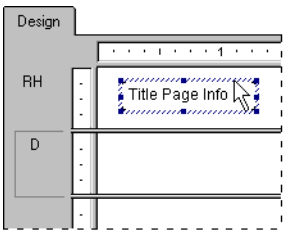
Crystal Reports provides a quick, easy way to add a title page to a report by selecting Report Title from the Special Fields in the Field Explorer dialog box. In order to use this field, you must have a title entered in the Summary tab of the Document Properties dialog box. See "Adding summary information to the report" on page 78.

To add a report title

1. On the Standard toolbar, click **Field Explorer.**

   The Field Explorer dialog box appears.

2. Scroll down to **Special Fields** and expand it by clicking.
3. Select **Report Title**, and click **Insert to Report**. An object frame appears when the cursor is moved over the report.



4. Move the object frame to the Report Header section and click once to place the frame.

5. With the report title selected, click **Section Expert** on the Expert Tools toolbar.

   The Section Expert appears.

6. With the Report Header section highlighted, select the **New Page After** check box.

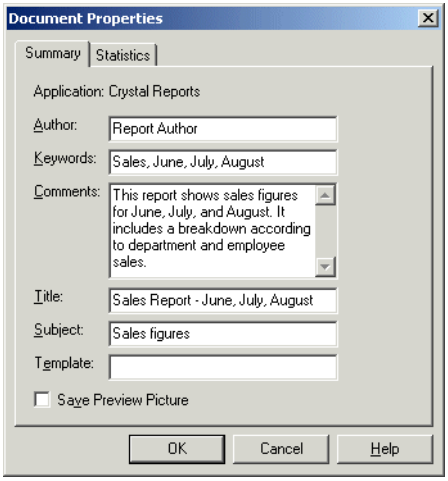
Now the title will appear on the first page and the report will begin on the second page.

Adding summary information to the report

There may be times when you want to include non-printing comments with a report (a personal note to the report recipient, a note to explain more thoroughly the data on which the report is based, a report title, a comment about some particular data on the report, and so on).

Summary Info on the File menu provides a facility for including anything from a short note to hundreds of lines of text with your report. The comments do not print with the report; they remain in the Summary tab of the Document Properties dialog box where they can be reviewed on demand.

When you choose Summary Info, the Document Properties dialog box appears with the Summary tab active.



Enter the desired information and click OK when finished to return to your report.

Beyond basic reports

Once you are comfortable with the basics of reporting, you will be ready to investigate the more powerful reporting features of Crystal Reports, including:

- Charts (see “Charting” on page 217)
- OLE objects (see “OLE” on page 251)
- Form letters (see “Form letters” on page 165)
- Subreports (see “Subreports” on page 365)
- Cross-tabs (see “Cross-Tab Objects” on page 261)
- Multi-section reports (see “Multiple Section Reports” on page 157)

Crystal Repository 6

This chapter introduces you to the Crystal Repository. It shows you how to add items to the repository, how to update existing items, and how to use repository items in your Crystal reports.

What is the Crystal Repository?

The Crystal Repository is the central location for you to store and manage your report objects. Data definitions such as custom functions and custom SQL commands can also be stored and maintained in the Crystal Repository. These objects are then accessible to users and report developers for use in new reports that can be distributed throughout your company.

Physically, the Crystal Repository is a database that stores supported object types. These types include:

- Text objects
- Bitmaps
- Custom functions
- Commands (queries)

The Crystal Repository shipped with Crystal Reports is an Access database. By default, it is located at `\Program Files\Common Files\Crystal Decisions\2.0\bin\Repository.mdb`. However, most database types are available for the Crystal Repository as long as you have a valid ODBC data source (DSN) set up for the type you choose. You can create an empty database, configure the DSN, and then set the database path in the `orMap.ini` file. If you create a different database for your Crystal Repository, the shipped repository is replaced.

Tip: For security, use an ODBC database as your Crystal Repository and set security at the database level.

By maintaining a shared repository of report objects, you can modify a particular object and update all reports containing that object as they are opened for use. A central location for report objects also helps with the task of managing your data—an important benefit in maximizing productivity and minimizing costs at your company.

Work flow

Although there are many ways to use the Crystal Repository, this sample work flow will acquaint you with some of the things you might want to do:

- **Create a new repository database and configure your connection (optional)**
  Crystal Reports comes with a sample repository database (Crystal Repository). If you want to use this database, you can skip this step.
- **Set up folders for your repository**
  The Repository Explorer displays the contents of your Crystal Repository as a tree with folders and subfolders. You can add and rename folders as you like.
- **Add report objects to your repository**
  There are different ways to do this depending on the type of object you want to add:
  - Select a text object or bitmap image and drag it to a folder or choose “Add to Repository” from the shortcut menu.

  - Select a Report Custom Function in the Formula Workshop and click the "Add to Repository" button.
  - Create or modify a command in the Database Expert and choose the "Add to Repository" option.
- **Add repository objects to a report**

  There are different ways to do this, depending on the type of object:
  - Text objects and bitmap images can be dragged to the report from the Repository Explorer.
  - Custom functions and commands can be added through their respective user interfaces.
- **Update report objects in your repository**

  Again, there are different ways to do this, depending on the type of object:
  - Text objects and bitmap images can be dragged back to the original repository object.
  - Custom functions and commands can be updated through their respective user interfaces.

Configuring a new repository

If you want to use a new database as your repository, you must configure an ODBC DSN for it and then modify the `orMap.ini` file.

Note: Configuring a new repository database assumes that you are replacing the sample Access database shipped with Crystal Reports. Only one repository is allowed.

To configure a new repository

1. Open the **ODBC Data Source Administrator**, which is usually found in the Windows Control Panel.
2. Click the **System DSN** tab, and then click **Add** to add a new ODBC data source.
3. In the Create New Data Source dialog box, choose the ODBC driver appropriate for your data source type from this list and click **Finish**.
4. Configure your ODBC connection as appropriate to the driver you chose.

   For more information about setting up your ODBC connection, see "Setting up an ODBC data source" in the online help.
5. Click **OK** when finished.
6. Edit the **orMap.ini** file, installed by default at `\Program Files\Common Files\Crystal Decisions\2.0\jars`, and change the existing data source mapping as follows:

   ```
   <the name you want to call your repository>=<the name of the ODBC DSN you created>
   ```

   Note: You cannot use these characters in your repository's name: - # " { } ; /

7 Save the changes you made to the .ini file.

Your new repository has been configured. If your new database is secure, you'll be prompted to log on the next time you open the Repository Explorer in Crystal Reports.

Accessing the sample Crystal Repository

The sample repository supplied by Crystal Reports has an ODBC System DSN, called Crystal Repository, which is set up when you install. You don't have to do anything additional before you use the sample repository.

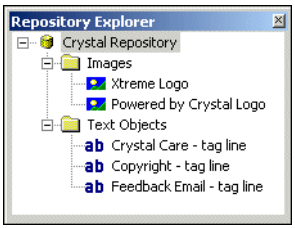
To open the sample Crystal Repository

1 Start **Crystal Reports** and open a report.

2 Click the **Repository Explorer** button on the Standard toolbar.

Tip: Another way to do this is to click the Repository Explorer option under the View menu.

The Repository Explorer appears.



3 Expand the node called **Crystal Repository** to see the sample repository.

The Repository Explorer might appear in a docked position in the Report Designer depending on where it was when you last used Crystal Reports. You can drag it where you like, or dock it elsewhere.

Adding folders to the repository

You can organize the contents of the repository by creating folders and subfolders in its tree view.

1 In the Repository Explorer, right-click the **Crystal Repository** node and click **New Folder** on the shortcut menu.

A new folder is added to the bottom of the repository tree.

2 Name your new folder and hit the Enter key.

3 To add a subfolder, right-click your new folder and click **New Folder** on the shortcut menu.

4 Name your new subfolder and hit the Enter key.

Adding items to the repository

You can add text objects and bitmap images to the repository by dragging them from your report to the appropriate repository folder. A dialog box appears so you can add identifying information about the object.

Custom functions are added to the repository through the Formula Workshop, and commands are added through the Database Expert.

Once report items are in the repository, they can be shared between many reports. Each person who uses Crystal Reports can connect to the repository and choose items to add to a report.

When you add an object to the repository, that object becomes “connected” to the repository. As long as an object remains connected, you can update any report that uses that object with the latest version in the repository.

Adding a text object or bitmap image

1. Select a text object or bitmap image in your report and drag it to the appropriate folder in the Repository Explorer.
2. In the **Name** field of the Object Information dialog box, create a name for your object or image.
3. Enter an author and description if you want, and then click **OK**.
4. In the Repository Explorer, right-click the repository node and click **Refresh** on the shortcut menu.

   You’ll see that your text object or bitmap image has been added to the appropriate folder.

You could have added the text object or bitmap image without dragging and dropping it.

1. Select a text object or bitmap image in your report and click **Add to Repository** from the shortcut menu.

   The Add Item dialog box appears.

   Notice that, because you did not drag the object to a specific repository folder, this dialog box contains an area to select a location from.
2. In the **Name** field, create a name for your object or image.

   Note: You cannot use these characters in your object’s name: # “ { } ; /
3. In the Location area, double-click your repository, select a folder, and then click **OK**.
4. In the Repository Explorer, right-click the repository node and click **Refresh** on the shortcut menu.

   You’ll see that your text object or bitmap image has been added to the appropriate folder.

If you try to edit any of the objects you added to the repository, you'll see that you can't make changes; the objects are in read-only mode. This is true of any report object that is stored in the repository: as long as it is connected to the repository it can't be changed in the report. If you right-click the object in the report and choose "Disconnect from Repository" from the shortcut menu, the item is disconnected from the repository and becomes editable. If you want other reports to be updated with the edited report object, you must add it back to the repository.

Adding a custom function

1. In the Formula Workshop, expand the **Report Custom Functions** node and select the custom function you want to add to the repository.

2. Click the **Add to Repository** button.

   The Add Custom Functions to Repository dialog box appears.

3. Select the repository to add the custom function to.

4. Click the **OK** button.

   You'll see that your custom function has been added to the Repository Custom Functions node.

Tip: You can also add a custom function to the repository by dragging its Report Custom Functions node—in the Workshop Tree—and dropping it on a Repository Custom Functions node.

Adding a command

1. In the Selected Tables area of the Database Expert, select the command you want to add to the repository.

2. Right-click the command and select **Edit Command** from the shortcut menu.

3. In the Modify Command dialog box, select **Add to Repository** and click **OK**.

4. In the Add Item dialog box, specify a name and repository location for the command.

   You'll find the command in the Repository node of the Database Expert, the Set Datasource Location dialog box, and the Data screen of the Report Creation Wizards.

Using repository objects in reports

Once you have an object or a collection of objects added to the repository, you can start using them in your Crystal reports. Each type of repository object is added through its own user interface. In the case of text objects and bitmap images, you simply drag them from the Repository Explorer to a report. Custom functions are selected when you work in the Formula Workshop, and commands are selected in

the Database Expert, Set Datasource Location dialog box, or Report Creation Wizard (Data screen).

When you add a repository object to a report, it remains connected to the repository and is in read-only mode. To edit the object, you must disconnect it from the repository and unlock its formatting.

Note: To reconnect a report object that has been disconnected from its repository, re-add the object or update the repository copy. Objects that remain disconnected from the repository cannot be automatically updated when the report is next opened.

This section shows you how to add a bitmap image and a custom function to a new report.

Adding a text object or a bitmap image to a report

1. Click the **Repository Explorer** button on the Standard toolbar.

   Tip: Another way to do this is to click the Repository Explorer option under the View menu.

2. Expand the **Text Objects** or **Images** folder of **Crystal Repository** and drag a text object or bitmap image to your report.

Note: Folders in your Crystal Repository do not have to be named to represent the objects they hold; you can use a name of your choice. See "Adding folders to the repository" on page 82 for more information.

Adding a custom function to a report

1. Click the **Formula Workshop** button on the Expert Tools toolbar.

   Tip: Another way to do this is to click the Formula Workshop option under the Report menu.

2. In the Formula Workshop, expand the **Repository Custom Functions** node until you find the custom function you want to add.

3. Right-click the custom function and click **Add to Report** from its shortcut menu.

   If the custom function you are adding to your report requires other custom functions from the repository, they can be added at the same time.

You'll see your custom function has been added to the Report Custom Functions node of the Formula Workshop.

Note: You can also add custom functions to a report while creating formulas in the Formula Workshop. For more information about this method, see "Creating a formula in the Formula Expert" on page 339.

Adding a command to a report

1. Click the **Database Expert** button on the Expert Tools toolbar.

   Tip: Another way to do this is to click the Database Expert option under the Database menu.

2. In the Database Expert, expand the **Repository** folder until you find the command you want to add.

3. Right-click the command and click **Add to Report** from its shortcut menu

You'll see your command has been added to the Selected Tables area of the Database Expert.

Modifying objects in the repository

You can modify repository objects by changing them in a report and adding them back to the repository. As well, you can rename objects and move them to different folders directly in the repository.

Be aware that when you modify an object and add it back to the repository, your changes affect all those who use that same repository: a renamed object is renamed for all users, and reports that contain the modified object may be updated upon opening them in the Report Designer.

To modify a repository object

Note: This procedure shows you how to modify and update a text object in the repository. Custom functions and commands are modified and updated in their respective user interfaces.

1. From the **Text Objects** folder of **Crystal Repository**, drag the **Crystal Care - tag line** object into a report.

2. Right-click **Crystal Care - tag line** in the repository and click **Properties** from its shortcut menu.

   Note the current version number.

3. Click the **Design** tab of your report, right-click the **Crystal Care - tag line** object, and select **Disconnect from Repository** from the shortcut menu.

4. Double-click the text object, and modify its text.

5. Drag the modified text object from the report back to the **Crystal Care - tag line** object in the Repository Explorer.

   When you drop the text object, the Add or Update Object dialog box appears.

6. Leave **Update the report object in the repository** selected and click **OK**.

7. Right-click **Crystal Care - tag line** in the repository once again and click **Properties** from its shortcut menu.

   Notice that the version number has been increased by one.

Updating connected repository objects in reports

Repository objects that are used in a report and are connected to that report's repository can be updated automatically when opening the report in Crystal Reports. This behavior is controlled by an Options setting and is global for all reports.

Note: Objects that are used in a report but are disconnected from the repository are not automatically updated when the report is opened.

To set the update option

1. On the **File** menu, click **Options**.
   The Options dialog box appears.
2. Click the **Reporting** tab.
3. Select the **Update Connected Repository Objects When Loading Reports** check box.
4. Click **OK**.

If you don't want to use this global option, you can also update a report's custom functions by selecting the Update Repository Objects option on the Open dialog box for an individual report

Deleting items from the repository

Any object you store in the repository can be deleted from that repository without removing it from the reports that use it. Once you remove an object from the repository, it is removed for all users.

Note:

- When you delete an object from the repository, it appears to remain connected in the reports that use it. When you try to update repository objects for those reports, a warning message is shown so you can quickly identify the object.
- The ability to delete repository objects is controlled by the permissions of the database. Once you accept the warning message in Crystal Reports, you will get an error message if you do not have adequate permission to delete the object from the database. The deletion is a logical deletion (that is, the object isn't removed from the database, but is flagged for deletion) and, therefore, it is an update to the database. You must have database update privileges before you can delete objects from the repository.

To delete an object from the repository

1. Click the **Repository Explorer** button on the Standard toolbar.

   Tip: Another way to do this is to click the Repository Explorer option under the View menu.

2. In the appropriate folder, select the object you want to remove and press the Delete key.

   You are asked to confirm the deletion. The note in this message box reaffirms that deleting the object from the repository will not remove it from any reports it's used in.

3. Click the **Yes** button.

   The object is removed from the repository.

You can also delete a repository object by right-clicking it and selecting Delete from the shortcut menu. In the case of a command, the shortcut option is called "Remove from repository."

Using Undo in the repository

You cannot undo any action that updates the repository. You *can* undo anything that affects only the report and not the repository. For example, you can undo the disconnect activity.

Designing Optimized Web Reports 7

This chapter suggests ways to enhance reports so they take advantage of performance enhancements made to Crystal Reports. While the suggestions made here are *especially* important for optimizing the performance of reports distributed over the thin-wire environment of the Web, the majority of the guidelines and procedures are applicable to all of your reports.

Overview

Whether you distribute your enterprise reports over a local network, a corporate Intranet, or the Web, you can use Crystal Reports' powerful, built-in performance features to deliver web reporting speed.

Crystal Reports automatically provides these major performance advantages—even if you don't apply any of the strategies set out in this chapter:

- **Page-on-demand technology**
  Page-on-demand report access lets users download only the specific report pages they need to see, thus improving response times and reducing web traffic. Further, placeholders and partial page technology allow you to view report pages and data over the Web immediately, without having to wait for the processing of large objects, such as graphics and subreports.
- **Optimized, multithreaded Report Engine**
  The Crystal Report Engine's multithreading capabilities and thread-safe database drivers allow you to continue working on your important tasks, while many other operations are processed simultaneously in the background. The Report Engine also minimizes the number of passes made through the data, speeds up processing with improved memory management, and handles subreports and parameters as efficiently as possible.

In addition to these built-in features, the "Key strategies for optimizing web reports" discussed in the following sections bring additional performance benefits, which are often substantial. When you design new reports (or improve your old ones) in accordance with these strategies, reports run faster and tie up fewer processing resources. Consequently, report users can easily access the data they need—faster than ever.

Tip: If you're new to web reporting, or to reporting in general, reading this chapter will prepare you for future reporting tasks—you'll gain important insight into designing faster, better reports.

Note: Understanding databases and how they work is often important when considering performance. See "Databases overview" on page 380 for background information.

Key strategies for optimizing web reports

- As your company grows, Crystal Reports and Crystal Enterprise can grow along with you. For details, see "Scale with Crystal Enterprise" on page 91.
- For tips and considerations on how to create fast, interactive reports, see "Making the right design choices" on page 91.
- For information about getting the most out of your existing database, see "Streamlining your reporting environment" on page 95.
- To minimize data transfer and enhance report performance, see "Using enhanced record selection formulas" on page 98.

- To decrease the processing and data transfer times of grouped, sorted, or totalled reports, see "Improving grouping, sorting, and totaling" on page 104.

In general, report performance issues are similar, no matter how you distribute your reports. By following these strategies, you'll notice significant improvements not only in multi-user web environments, but also in single-user situations.

Scale with Crystal Enterprise

Crystal Enterprise extends the value and scalability of Crystal Reports. It provides eBusiness with a scalable, web-based solution for managing the access and delivery of hundreds or thousands of Crystal reports to every decision-maker—across the enterprise and beyond.

Crystal Enterprise provides a flexible, managed reporting solution. It gives you the ability to deliver the report-design power of Crystal Reports, atop a scalable, web-based infrastructure for managing a central report repository, user security, and report scheduling and processing.

Making the right design choices

This section offers design options and considerations that will help improve the performance of your reports. The topics covered range from basic suggestions, such as updating older reports to the latest file format, to more involved decisions, such as whether to use live or saved data, and how to use subreports efficiently.

When designing your reports, and especially when designing reports for the Web, you should allow report users to drive the data they see. In other words, display summarized information, so each user can navigate the report quickly and then drill down to access additional data. In this way, web traffic and response times are minimized, because only the data requested by the user is transferred from the database server.

These are only a few of the benefits of designing user-driven reports:

- Report users gain interactive control over the type and quantity of information they view over the Web.
- Data transfer and network traffic decrease, because only the information requested by users is returned from the database server.
- When users need real-time reporting of live data over the Web, user-oriented reports respond quickly and communicate efficiently with the database server.
- Reports become more useful, because each user customizes the report's contents, thereby creating a reporting solution specific to his or her particular decision-making problem.

Using faster report formats

The quickest way to help improve the performance of your old reports is to save them in the latest Crystal Reports format. Crystal Reports has been enhanced to process old reports faster than ever before: update your older reports to take advantage of these enhancements.

To update the format of an older report, just open it in Crystal Reports and select Save from the File menu. The older version of the report will be replaced with a version 9 report.

Note: If for some reason you need to keep an old report in its original file format, use the Save As command (instead of Save) and enter a new name for the version 9 report.

Choosing between live and saved data

When reporting over the Web, the choice to use live or saved data is one of the most important decisions you'll make. Whichever choice you make, however, Crystal Reports displays the first page as quickly as possible, so you can see your report while the rest of the data is being processed.

Live data

Live reporting gives users real-time access to live data, straight from the database server. Use live data to keep users up-to-date on constantly changing data, so they can access information that's accurate to the second. For instance, if the managers of a large distribution center need to keep track of inventory shipped on a continual basis, then live reporting is the way to give them the information they need.

Crystal Reports supports live reporting. However, you should first consider whether or not you want all of your users hitting the database server on a continual basis. If the data isn't rapidly or constantly changing, then all those requests to the database do little more than increase network traffic and consume server resources. In such cases, you may prefer to use reports with saved data.

To ensure the efficiency of real-time reporting, read all of the suggestions in this chapter. These topics, however, are of particular importance:

- "Taking advantage of on-demand subreports" on page 94
- "Performing grouping on server" on page 104
- "Incorporating parameter fields into record selection formulas" on page 101

Saved data

Reports with saved data are useful for dealing with data that isn't continually updated. When users navigate through reports with saved data, and drill down for details on columns or charts, they don't access the database server directly; instead, they access the saved data. Consequently, reports with saved data not only minimize data transfer over the network, but also lighten the database server's workload.

You can schedule these reports within Crystal Enterprise, so they automatically refresh from the database on a predetermined basis. For example, if your sales database is only updated once a day, or once a week, then you can run the report on a similar schedule and save it with data. Sales representatives then always have access to current sales data, but they aren't hitting the database every time they open a report. Alternatively, you can refresh reports with saved data on an as-needed basis.

If you do use reports with saved data, incorporate the other suggestions in this chapter to make sure your reports are designed for optimum performance.

To save a report with data, first make sure the Save Data with Report option is selected on the File menu; then Save your report.

Designing summary reports

Designing and distributing summary reports is a relatively easy way to ensure that users quickly find the data they need over the Web. A summary report can include as much data as any other report. However, by hiding a summary report's Details section, you avoid overwhelming users with data they may not immediately need.

When the Details section is hidden, users navigate with the Group tree first, to locate the desired data. Then, by drilling down on the report, they can request specific data, which is returned quickly without unnecessary records. This is especially important to improving navigation of long summary reports, which might consist of hundreds, thousands, or even tens of thousands of pages.

To facilitate navigation in this way, you first need to group the data and insert the summary fields you want to include in your report. For detailed information and instructions, see "Grouping data" on page 122 and "Summarizing grouped data" on page 135.

Once you've grouped and summarized your report data, hide the Details section (and any other large report sections) so that users can easily navigate to the data that's important to them.

Hiding details in a summary report

1. Open your report in Crystal Reports.

   If you haven't yet created a grouped and summarized report, open Summary Group.rpt from the Feature Examples samples folder.

2. On the **Report** menu, click **Section Expert** to open the Section Expert.
3. In the **Sections** list, click **Details**.
4. On the Section Expert's **Common** tab, select the **Hide (Drill-Down OK)** check box.
5. Click **OK**.

You'll notice the details are now hidden. To view the details, navigate through the report using the Group tree, and then drill down on the appropriate area of the report.

Note: For details on minimizing data transfer with summary reports, see "Performing grouping on server" on page 104.

Using subreports carefully

For general information on subreporting, see "Subreports" on page 365.

Taking advantage of on-demand subreports

If your report has a section that handles a large number of records, you can put that section into an on-demand subreport. An on-demand subreport appears as a hyperlink in the primary report. When you open the primary report, no data is retrieved for the on-demand subreport until you drill down on the hyperlink.

For example, when designing a report that shows each employee's quarterly sales for each product and each product type, you might also want to track each employee's progress by including weekly sales information. This additional data, however, may not be of interest to many users viewing the report. In such a case, extract the weekly sales portion of the report and attach it as an on-demand subreport. Detailed information about weekly sales is retrieved from the database only when a user drills down on the on-demand subreport.

Many report objects—such as large cross-tabs, OLAP grids, advanced charts, and maps—are ideal candidates to include in on-demand subreports, so that the object is not processed until you drill down.

To insert an on-demand subreport, see "Inserting subreports" on page 369 and "Creating an on-demand subreport" on page 376.

Tip: You could also place such report objects in a hidden Details section of a report that uses the Perform Grouping On Server option. When you do this, the database server performs the majority of the processing, and only a subset of the records is transferred from the server to the local machine (other records are retrieved when you drill down to a hidden section).

Using linked subreports

When a subreport is linked, Crystal Reports coordinates the data in the subreport with the matching records in the primary report. If you need to use regular linked subreports—that is, linked subreports that are not on-demand subreports—you should consider these guidelines:

- If the additional data provided by a regular linked subreport is useful to relatively few users, create a linked on-demand subreport instead. Users who need to see the extra data click a hyperlink to view the subreport; users who don't need to see the extra data won't have to download it from the database server.

- In some cases, placing regular linked subreports in the Details section of a main report may hinder performance—especially when your main report contains many records. (This is because you're creating a separate subreport for each record, and a separate query must be run for each database record in the main report.) As an alternative, consider using linked on-demand subreports in the Details section of your main report.

Linking tables instead of linking subreports

Whenever possible, coordinate your report data by linking database tables on the Links tab of the Database Expert, rather than by linking regular subreports—that is, subreports that aren't on-demand subreports—to the main report. Since each subreport is run as a separate report, linked tables often have a performance advantage.

Related topics

For further general guidelines, see "Database links vs. subreports in one-to-many situations" on page 368.

For complete details, see "Performance considerations in one-to-many links" on page 386.

Streamlining your reporting environment

Another step to ensuring that report users receive their information quickly is to assess your reporting environment. What kind of database do you use? How is data organized within the database? How are you connecting to the data you need to report off? How are you linking your database tables? By taking these important considerations into account, you can significantly minimize the amount of data that must physically travel across the network.

Selecting the fastest database and connection

To improve reporting performance, utilize your database to its full potential by having it do the majority of your report processing. Ideally, Structured Query Language (SQL) databases are the most efficient for carrying out this task.

For details on SQL databases, see "Using SQL and SQL databases" on page 403.

Using table indexes

You can also improve the way your data is organized within the database. For optimum processing speed, report off indexed fields on your SQL database. Using table indexes is an easy way to increase the speed of data access and to reduce the time it takes for Crystal Reports to evaluate data.

For a complete description of how table indexing works, see "Indexed tables" on page 383.

Improving table-linking choices

When you add multiple database tables to your report, you link the tables on a common field so that records from one table match related records from another. (Table linking works best if your database tables are indexed.) Linking your database tables in this way is usually much faster than incorporating linked subreports into your main report.

When you link two or more tables, you want your report to read as few records as possible, while at the same time finding all matching records. The best ways to adhere to this guideline are to assess your reporting needs and to plan your strategy before creating your report. When you know exactly what you need from your data source, Crystal Reports makes it easy to get that information.

There are many other specific issues to consider when linking tables. These additional considerations, however, are largely contingent upon your reporting environment. In other words, the steps to optimal table-linking performance depend upon your database type, the possibility of indexing tables, and the join type required between tables. You'll find complete descriptions for various reporting scenarios in the section entitled "Linking tables" on page 385.

In the majority of reporting scenarios, the following general procedures should ensure that your tables are linked for improved if not optimal performance.

To link tables effectively

1. Ensure that each database table is indexed on the field you are going to use.
2. Add the database tables to your report, and then link from the primary table to the lookup table on a common field. (Do this instead of inserting a linked subreport and binding it to the data in your primary report.)
3. Use a record selection formula that sets range limits on the indexed field in the primary table. This minimizes the number of records in the primary table for which Crystal Reports must find matching records in the lookup table.

Related topics

For complete details on table-linking scenarios, see "Linking tables" on page 385.

For general information on record selection formulas, see "Selecting records" on page 108.

For advanced record selection strategies, see "Using enhanced record selection formulas" on page 98.

Using thread-safe database drivers

If you share and refresh reports over the web, open them using thread-safe database drivers whenever possible. The Crystal Report Engine supports multiple threads. Thus, when you make multiple data requests through thread-safe database drivers, the Report Engine can process all of the requests simultaneously. As the result, you'll be able to view your reports sooner.

Currently, the ODBC driver crdb_odbc.dll is thread-safe for the following database types:

- Microsoft SQL Server
- Sybase

Currently, the following native database drivers are thread-safe:

- crdb_ado.dll (OLE DB)
- crdb_cdo.dll (Crystal Data Object)
- crdb_com.dll (COM data provider)
- crdb_dao.dll (Access)
- crdb_dataset.dll (DataSet provider)
- crdb_fielddef.dll (Field Definitions)
- crdb_filesystem.dll (File System data)
- crdb_olap.dll (OLAP)
- crdb_oracle.dll (Oracle)
- crdb_p2sdb2.dll (DB2)

Note: Check the Release Notes on the Crystal Reports CD for updates to these lists; more database drivers may be added if they are found to be thread-safe during testing.

Using stored procedures for faster processing

A stored procedure is a compiled SQL program, consisting of one or more SQL statements, which resides and runs on your SQL server. Although stored procedures can take time to set up properly, they can be incredibly powerful.

The advantages of using stored procedures are most noticeable when reporting off large sets of data, or when running reports that demand long, complex calculations. In such cases, you should ideally use a predefined stored procedure that performs the complex work for you on the database server.

Because stored procedures reside on your SQL server, you may need to ask your Administrator about accessing or setting up a stored procedure.

Related topics

For more information on stored procedures, see "Stored procedures" on page 406.

To select a stored procedure as a data source, see "SQL stored procedures" in the online help.

Using enhanced record selection formulas

The most important thing you can do to speed up report processing is to limit the amount of data that is returned from the database. Your primary tool for doing this is the record selection formula.

Crystal Reports analyzes your record selection formula and generates an SQL query from it. This SQL query is then processed by the database, which sends the resulting records back to Crystal Reports. Crystal Reports then evaluates the record selection formula locally for each of the records retrieved from the database, thereby calculating the set of records used to generate the report.

Unnecessary records are eliminated at two stages: at the database with the SQL query and within Crystal Reports by the record selection formula. For speed, you want as many records as possible to be eliminated in the first stage. By designing your record selection formula effectively, you can off-load much of the processing to the database server, thus eliminating unnecessary records before returning them to Crystal Reports. This is commonly called “pushing record selection down to the database server.”

This section offers several tips to ensuring that your record selection formulas can be pushed down to the database server.

Pushing down record selection—an example

This example demonstrates the benefits of writing record selection formulas that can be pushed down to the database server.

In the Orders table of the Xtreme sample database, there are 2192 records, of which 181 have order dates prior to 2001. Suppose you want to report on only those records. On the one hand, you could use this record selection formula:

```
Year ({Orders.Order Date}) < 2001
```

The SQL query generated will send all 2192 records to Crystal Reports, and then the record selection formula will reduce this to 181. To see this, click Show SQL Query on the Database menu and notice that the SQL query has no WHERE clause. This is because Crystal Reports is not able to push down the Year ( ) function in the WHERE clause.

On the other hand, this record selection formula generates the same report:

```
{Orders.Order Date} < #Jan 1, 1997#
```

This second formula, however, can be performed on the database server, so it is pushed down. The SQL query generated will send only 181 records to Crystal Reports. So, when the record selection formula is evaluated by Crystal Reports, no further records need to be eliminated. Click Show SQL Query on the Database menu and notice that the resulting SQL query has a WHERE clause.

As this example shows, your report's processing speed improves when you enhance your record selection formula. In this case, both formulas generate the same report, but the second takes advantages of the power and optimizations that the database server can use when handling its own data.

Tip: See the next section for additional information and limitations when setting up record selection requests.

Note: If you are new to record selection formulas, you may prefer to begin with the Select Expert or the sample record selection formula templates. For further details, along with an introduction to record selection, see "Selecting records" on page 108.

Record selection performance tips

Consider the following performance-related items when setting up record selection requests:

General

- To push down record selection, you must select "Use Indexes or Server for Speed" in the Report Options dialog box (available on the File menu).
- In record selection formulas, avoid data type conversions on fields that are not parameter fields.

  For example, avoid using ToText( ) to convert a numeric database field to a string database field.
- You are able to push down some record selection formulas that use constant expressions.

PC Databases

- You can only push down record selection on indexed fields.
- You can only push down AND clauses (not OR).

SQL Databases

- You can push down record selection on indexed or non-indexed fields.
- Your SQL server will respond faster if you use indexed fields.
- You can push down AND and OR clauses.
- Record selection formulas containing some types of embedded formulas can be pushed down.
- You should incorporate SQL Expression fields to push down formula calculations necessary for the record selection.
- On the Database menu, click Show SQL Query to view the SQL that will be sent to the database server.

Strategies for writing efficient record selection formulas

Note: This section assumes that you are familiar with the Select Expert and that you are reporting off of a SQL database.

Consider the following points when creating a record selection formula:

Consideration 1

Any record selection formula that you generate completely with the Select Expert, without writing pieces of the formula yourself, can be pushed down.

This case actually follows from the points below. However, you can write more types of record selection formulas using the tips below than is possible with the Select Expert. To do this, you need to edit the record selection formula directly with the Formula Workshop or from within the text area that appears when you click Show Formula in the Select Expert.

To open the Formula Workshop to modify record selection, click the Report menu, point to Selection Formulas, and then select Record from the submenu.

Consideration 2

Any selection formula that is of the form: *DataBaseField SupportedOperator ConstantOrParameterExpression* can be pushed down.

Of course *DataBaseField* is just a database field. *SupportedOperator* is any of =, <>, <, <=, >, >=, StartsWith, Like or In.

ConstantOrParameterExpression is any expression that involves constant values, operators, functions, and parameter fields. It cannot involve variables, control structures, or fields other than parameter fields. By their very definition, constant and parameter expressions can be evaluated without accessing the database.

Note: A constant or parameter expression can evaluate to a simple value, a range value, an array value, or an array of range values. Here are some examples of such expressions:

```
{?number parameter} - 3
Year ({?run date})
CurrentDate + 5
DateDiff ("q", CurrentDate, CDate("Jan 1, 1996"))
Month (Maximum ({?date range parameter}) + 15)
["Canada", "Mexico", "USA", {?enter a country}]
1000 To 5000
[5000 To 10000, 20000 To 30000, 50000 To 60000]
```

A complete example:

{Orders.Order Date} >= CurrentDate - 3

The program can also push down an expression that just contains a Boolean field (without the operator and constant parts).

```
{Orders.Shipped}
Not {Orders.Shipped}
```

Consideration 3

IsNull (*DataBaseField*) can be pushed down.

Consideration 4

SqlExpression SupportedOperator ConstantOrParameterExpression can be pushed down.

For example, the selection formula `{@ExtendedPrice} > 1000` cannot be pushed down if `{@ExtendedPrice} = (Quantity * Price)`. However, if the formula `@ExtendedPrice` is replaced with the equivalent SQL Expression, then your record selection formula will be pushed down.

Consideration 5

When using multiple expressions that follow the above considerations, separate them with AND and OR operators—you can also use NOT. You can have several of each, and you can use parentheses to give priority. For example:

```
{Orders.Order ID} < Minimum({?number range}) Or
{Orders.Order Amount} >= 1000
```

```
(IsNull({Customer.Region}) Or
{Customer.Region} = "BC") And
{Customer.Last Year's Sales} > 2000
```

Related topics

If your record selection formula does not respond as expected, see "Troubleshooting record selection formulas" on page 115.

Incorporating parameter fields into record selection formulas

Instead of displaying all of a report's data every time the report is opened, you can create parameter fields that prompt users to specify the data they want to see. To decrease the amount of data transferred from the database server, incorporate these parameter fields right into your record selection formula.

In general, parameter fields provide interactivity for users, who respond to the parameter prompts in order to specify the data they want to see. However, by incorporating your parameter fields right into your record selection formula, you not only provide interactivity, but you also decrease data transfer and increase performance.

You can add a parameter field to your record selection formula by using the Select Expert or the Record Selection Formula Workshop. When using the Record Selection Formula Workshop, you treat the parameter field as you would any other field.

In the following example, you will create a parameter field and then add it to the record selection formula with the Select Expert. For further details on designing and creating parameter fields, see "Parameter Fields" on page 349.

To add a parameter field to your record selection formula

1. Open the sample report **Group By Intervals.rpt** (located in the Feature Examples sample folder).

   Take a moment to navigate the Group tree and see how the data is organized; notice also, in the bottom right of the Crystal Reports window, that 269 records were returned for this report.

   Note: You'll see this number only if you have the Status Bar option selected on the View menu.

   Now click the Design tab.

2. On the **View** menu, click **Field Explorer**.

3. In the Field Explorer, right-click **Parameter Fields** and select **New** from the shortcut menu. The Create Parameter Field dialog box appears.

4. For the **Name** of the parameter, type `SalesQuota`.

5. For the **Prompting text**, type `What was last year's sales quota?`

6. Click the **Value type** list and select **Number**.

7. Make sure **Discrete value(s)** is selected, and then click **OK**.

   Note: You've now created the parameter field. The rest of these procedures describe how to add the parameter field to the record selection formula with the Select Expert.

8. On the **Report** menu, click **Select Expert**. The Choose Field dialog box appears.

9. Select the **Customer.Last Year's Sales** field, and then click **OK**. The Select Expert appears.

10. On the **Customer.Last Year's Sales** tab, click the drop-down list and select **is greater than**. A new drop-down list will appear.

11. Click this second list and select **{?SalesQuota}**.

    Tip: Click the Show Formula button to view the new record selection formula, which appears as: `{Customer.Last Year's Sales} > {?SalesQuota}`. Instead of using the Select Expert, you could have created this formula yourself in the Record Selection Formula Editor. (To see the editor, open the Formula Workshop and select Record Selection from the Selection Formulas folder.)

12. Click **OK** in the Select Expert.

You've now added your parameter field to the record selection formula. When you switch to Preview mode, or click the Refresh button, you will be prompted for new parameter values. You can then enter a numeric value representing last year's sales quota. The resulting report will display only those customers whose Last Year's Sales exceeds the numeric value you specify.

For instance, if you respond to the parameter by entering 40000, then the report will display only those customers whose sales exceeded $40,000 last year. Notice

also, in the bottom right of the Crystal Reports window, that only 58 records are returned for your report, instead of the 270 records that were returned before you put the parameter field into a record selection formula.

By enhancing this report, you've retrieved all the information you needed and, at the same time, you've ensured that the fewest number of records is transferred from the database server.

Related topics

To create enhanced parameters, see "Advanced parameter features" on page 356.

For general information about parameter fields, see "Parameter overview" on page 350.

For general details regarding record selection formulas, see "Record Selection" on page 107.

Using SQL expressions where appropriate

SQL expressions are like formulas, but they are written in Structured Query Language (SQL). They are useful in optimizing report performance because the tasks they execute are normally performed on the database server (as opposed to a regular formula, which is sometimes executed on the local machine).

Don't use SQL expressions exclusively, because Crystal Reports includes its own formula language, which is much more powerful than standard SQL. Both Crystal Syntax and Basic Syntax allow you to enhance and hone your formulas in ways that are either difficult or impossible with SQL. In certain circumstances, however, creating and using SQL Expression fields can speed up the processing of your reports.

Key uses of SQL Expression fields

To maintain optimum report processing speeds, avoid using formulas (whether Crystal or Basic syntax) within record selection formulas. Instead, replace the original formula with an equivalent SQL Expression field, and then incorporate the SQL Expression field into your record selection formula. Doing so will greatly improve the chances of your record selection being pushed down to the server.

In addition, avoid sorting, grouping, or totaling on a formula field (whether Crystal or Basic syntax). Instead, replace the original formula field with an equivalent SQL Expression field, and then sort, group, or total on the SQL Expression Field. Again, this will greatly improve the chances of the processing being done on the server.

Finally, if your database supports Case Logic, and your report needs to summarize an If-Then-Else formula calculation, replace the formula with an SQL Expression field. In such cases, SQL Expression fields enable Crystal Reports to perform the report's grouping on the server. For more information, see "Using SQL Expressions for Case Logic" on page 106.

Related topics

For an introduction to SQL, see "What is SQL?" on page 403 and "The SQL language" on page 407.

For instructions on creating an SQL Expression field, see "SQL Expression fields" on page 68.

For further tips on enhancing your record selection formula, see "Using enhanced record selection formulas" on page 98.

For general record selection procedures, see "Record Selection" on page 107.

Improving grouping, sorting, and totaling

Performing grouping on server

If you're reporting in real-time off live data over the Web, reduce the amount of data transferred from the database server by using the Perform Grouping on Server option. With this option, much data processing is off-loaded to the database server and only a subset of data is read initially. Detail data is returned from the database only when you drill down in a report.

Note: Server-side processing works only for sorted and grouped reports that are based on SQL data sources.

To enable server-side processing

1. On the **File** menu, click **Report Options**.
2. Select **Perform Grouping on Server** on the Report Options dialog box.

   This check box is inactive if Use Indexes or Server for Speed is not selected.

   Tip: When Use Indexes or Server for Speed is selected, you can quickly enable Perform Grouping On Server from the Database menu.
3. Click **OK**.

Benefits of grouping on server—an example

The following example describes a typical reporting scenario wherein grouping on server can significantly reduce the amount of unnecessary data transferred from the database server.

To decrease data transfer by grouping on server

1. Open the sample report **Group.rpt** (located in the Feature Examples folder).

   Look in the bottom right corner of the Crystal Reports window and notice that 269 records are returned for this report.

Note: You'll see this number only if you have the Status Bar option selected on the View menu.

2. On the **Report** menu, click **Section Expert** to access the Section Expert.
3. In the **Sections** list, select **Details**.
4. On the **Common** tab, select **Hide (Drill-Down OK)**. This hides the report's Details section, so that only group header displays in the report. (In this case, the report is grouped by Country.)
5. Click **OK** in the Section Expert. The Details records will disappear from the report.
6. Press F5 to refresh the report data (or click the **Refresh** button on the toolbar).

   Notice that 269 records are still returned for the report, even though the Detail records are hidden from view.
7. On the **Database** menu, click **Perform Grouping on Server**.

   Now notice that only 71 records are returned for the report. The grouping (by Country) has been performed on the database server, and fewer records have been transferred to the report.
8. Drill down on a country by double-clicking the report.

   Tip: When drill down is available, your cursor turns into a magnifying glass.

   Crystal Reports retrieves the appropriate Detail records as you need them.

   For instance, if you drill down on Australia, Crystal Reports quickly retrieves the seven records that make up the group.

By hiding the Details section of this report, you've created a summary report that's easy for users to navigate. Each user can first locate the Country with which he or she is concerned, and can then drill down to retrieve valuable details.

Moreover, by enabling the Perform Grouping on Server option, you've ensured that the initial processing is completed on the database server. Consequently, only the necessary records are transferred to the report.

For more information on server-side processing, see "Server-side processing" on page 409.

Using SQL Expressions for groups, sorts, and totals

For reports using Perform Grouping on Server, avoid sorting, grouping, or totaling on a formula field (whether Crystal or Basic syntax). Instead, replace the original formula field with an equivalent SQL Expression Field, and then sort, group, or total on the SQL Expression field. This will greatly improve the chances of the processing being done on the server.

For details on when else to use SQL Expressions, see "Using SQL expressions where appropriate" on page 103.

Using SQL Expressions for Case Logic

If your database supports Case Logic, and your report needs to summarize an If-Then-Else formula calculation, replace the formula with an SQL Expression field. In such cases, SQL Expression fields enable Crystal Reports to perform the report's grouping on the server.

For instance, suppose that you're reporting off of an MS SQL Server 7 database, which supports Case Logic. You need to include an If-Then-Else calculation in your report, and you need to summarize that calculation for each group in the report. By performing the calculation with an SQL Expression field of the following form, you use your database's ability to process Case Logic:

```
CASE DatabaseTable."DatabaseField"
WHEN 'SpecifiedValue' THEN Calculation1
ELSE Calculation2
END
```

If a record's *DatabaseField* value is equal to *SpecifiedValue*, then *Calculation1* is performed; for all other records, *Calculation2* is performed. By incorporating the SQL Expression field, you take advantage of your database server's ability to process Case Logic. Your report's grouping consequently takes place on the server, even when you summarize the SQL Expression field elsewhere in the report.

Note: The SQL syntax in this example is specific to MS SQL Server 7. You may need to refer to your database documentation or consult with your Administrator in order to determine the syntax appropriate to your database.

Inserting summary and running total fields where possible

Where possible, avoid creating formulas with global variables to calculate summaries or running totals.

Instead, create summaries by clicking the Insert menu and then selecting the appropriate command (Subtotal, Grand Total, or Summary). Create running total fields by opening the Field Explorer, right-clicking Running Total Fields, and then selecting New from the shortcut menu.

For general details on summarizing data, see "Sorting, Grouping, and Totaling" on page 119 and "Running Totals" on page 147.

Record Selection 8

This chapter shows you how to filter the records you want included in a report. For example, using the record selection tools, you can limit the records in your report to include only records for a specific group of customers, a specific range of account numbers, or a particular date range.

Selecting records

When a field is selected to appear on a report, field values from every record in the active table(s) are printed by default. In many cases, you may not want to include all the values, but only a subset of those values. For example, you may want to include:

- Records only for a specific group of customers.
- Records for a specific range of account numbers out of the total number of records in the database.
- Values from only those records that fall within a particular date range.

Options for selecting records

Crystal Reports includes a very sophisticated formula language that you can use to specify virtually any type of record selection. However, you may not always need the flexibility in record selection that the formula language provides. The Select Expert is designed for such instances.

You can select records in one of two ways:

- "Using the Select Expert" on page 109
- "Using formulas" on page 110

Once you are familiar with the Select Expert and formulas, you can use your record selection techniques to improve the performance of your reports. For additional performance tips and advanced record selection strategies, see "Using enhanced record selection formulas" on page 98.

Determining which field(s) to use

When you select records, you are basing your report only on those records that meet some conditions that you have set. You base those conditions on the kind of information you want in the finished report.

Assume, for example, that you want a report that only shows data from California. The challenge is to find the best way to identify those records that come from California.

- If the table used in a report has a State or Region field, you can specify in your request that the program use only those records in which the value in the state field is equal to California (Region is equal to CA).
- If the table does not have a State field and you still want to report only on California data, there may be another way to identify that data.
  - If the table has a Postal Code field, you could base your record selection on the range of ZIP codes that apply to California (Postal Code is between *n* and *N*).
  - If the table has an Area Code field, you could base your record selection on California Area Codes (Area Code is one of x, y,....z).

Note:

- If the Area Code is stored in the telephone number field, you will not be able to do this same record selection using the Select Expert based on the Area Code.

You will have to create a record selection formula using the formula language to extract the Area Code part of the phone number and then do record selection based on that.

- As a general rule, if you can base your record selection on a number of fields (as in this example), you should select an indexed field instead of a field that is not indexed for better performance.

Using the Select Expert

The Select Expert makes it easy to specify the records you want included in your report. When you work with the Select Expert, you select the field to which you want to apply selection conditions and then you specify those conditions.

The Select Expert can be used to set up simple record selection requests. For example:

- Customers from Arizona.
- Orders in the first quarter.
- Sales over $10,000.

The Select Expert can also be used to set up some sophisticated requests:

- Customers whose names start with “A”, “M”, or “S.”
- Customers from California or Florida who ordered in July.

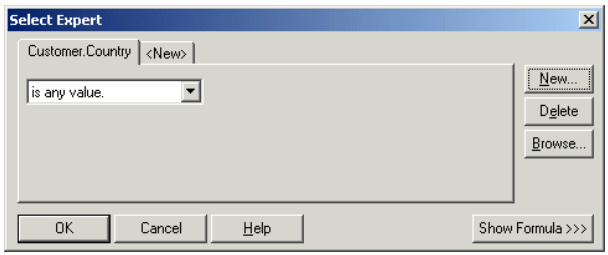
These are all range limit requests. One or more constants define the range. The program compares the field value in each record to the constant(s) and rejects records with values outside the range. The report is limited to values within the range. You can set up all of these types of record selection requests without any previous knowledge of the formula language.

Note: The Select Expert can be used to set up both record selection and group selection requests. When a group name or summary field is selected, the program knows that the selection criteria set up is intended for group selection. In all other cases, the program knows that you are setting up record selection.

To set up record selection using the Select Expert

1 On the Expert Tools toolbar, click **Select Expert**.

The Select Expert dialog box appears.



Note: If you click the Select Expert button without first highlighting a field in your report, the Choose Field dialog box appears. Highlight the field on which you want to base record selection and click OK. The Select Expert appears.

2 Use the drop-down lists to enter your selection criteria for the indicated field.

3 Click **OK** when finished.

Tip: To base record selection on more than one field, click the New tab. Select the next field from the Choose Field dialog box.

A selection formula is generated based on your specifications, limiting the report to the records you indicated.

Note: To view the selection formula, click the Show Formula button. The Select Expert expands to show the formula. You can modify the formula in the area that appears, or you can click the Formula Editor button to modify it in the Formula Editor.

Using formulas

To set up record selection using a formula

1 On the **Report** menu, point to **Selection Formulas** and click **Record**.

2 In the Record Selection Formula Editor, enter the formula by typing in the components or selecting them from the component trees.

Note: The resulting formula must be Boolean; that is, it must return either a True or False value.

3 Click **Check** to identify any errors in the formula.

4 Fix any syntax errors the Formula Checker identifies.

5 When the formula has the correct syntax, click **Save**.

Related topics

For sample record and group selection templates, see "Using formula templates" on page 112.

For complete instructions on creating formulas, see "Using Formulas" on page 329.

For advanced record selection strategies and performance tips, see "Using enhanced record selection formulas" on page 98.

Interaction of the Select Expert and the Selection Formula Workshop

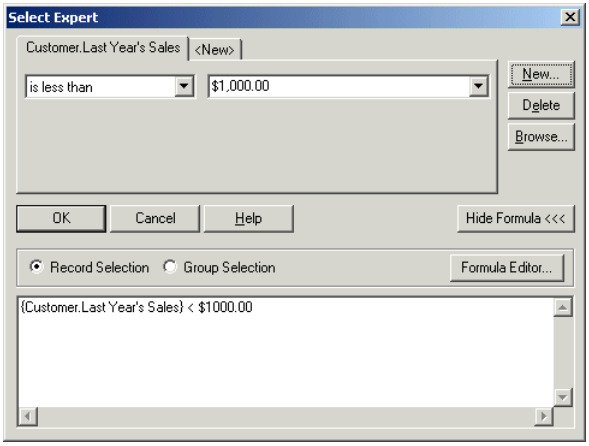
The Select Expert and the Record / Group Selection Formula Editor are interactive. That is, record selection criteria you enter via the Select Expert automatically generates a record selection formula which can be reviewed and modified. Likewise, record selection formulas and modifications to existing record selection formulas automatically update the selection criteria in the Select Expert.

Because of this interactivity, you can use the two tools together as a tutorial for learning the formula language.

To view the Select Expert formula

1. Right-click the field on which you want to view record selection.
2. Click **Select Expert**.
3. Click the **Show Formula** button.

   The Select Expert expands so you can review the formula the program generated based on your selection criteria.



4. Click the **Hide Formula** button when you are done with your review.
5. Use the Select Expert to change your selection formula.
6. Review the updated formula by clicking the **Show Formula** button again.
7. To make changes to the formula, click the **Formula Editor** button in the expanded Select Expert and use the tools in the Formula Workshop to make your formula changes.

Note: Selection formula components that do not fit any of the fixed criteria in the Select Expert will not be translated. For example, if part of your record selection formula extracts the last four characters in a customer number, the section of the formula code that performs that extraction will not be converted to Select Expert selection criteria.

Using formula templates

Record selection formula templates

The following sample formulas can be used as templates to help you create your own selection formulas using the Formula Workshop. These examples illustrate different selections that you *could* do, not necessarily the best selections.

Selecting records using character strings

`{file.FIELD} startswith "C"`

Selects those records in which the value in the {file.FIELD} field begins with the character “C” (includes values like CyclePath, Corp. and Cyclist’s Trail Co.; excludes values like Bob’s Bikes Ltd. and Feel Great Bikes, Inc.).

`not ({file.FIELD} startswith "C")`

Selects those records in which the value in the {file.FIELD} field does not begin with the character “C” (includes values like Bob’s Bikes Ltd. and Feel Great Bikes, Inc.; excludes values like CyclePath, Corp. and Cyclist’s Trail Co.).

`"999" in {file.FIELD}[3 to 5]`

Selects those records in which the 3rd through 5th digits of the {file.FIELD} field is equal to “999” (includes values like 10999, 70999, and 00999; excludes values like 99901 and 19990).

`"Cycle" in {file.FIELD}`

Selects those records in which the value in the {file.FIELD} field contains the string “Cycle” (includes values such as CyclePath Corp. and CycleSporin, Inc.; excludes values like Cyclist’s Trail Co. and Feel Great Bikes, Inc.).

Selecting records using numbers

Single values

`{file.FIELD} > 99999`

Selects those records in which the value in the {file.FIELD} field is greater than 99999.

`{file.FIELD} < 99999`

Selects those records in which the value in the {file.FIELD} field is less than 99999.

Range of values

`{file.FIELD} > 11111 and {file.FIELD} < 99999`

Selects those records in which the value in the {file.FIELD} field is greater than 11111 but less than 99999 (neither 11111 or 99999 is included in the range of values).

```
{file.FIELD} >= 11111 and {file.FIELD} <= 99999
```

Selects those records in which the value in the {file.FIELD} field is greater than 11111 but less than 99999 (both 11111 and 99999 are included in the range of values).

Selecting records using dates

The Month, Day, and Year functions can all be used in examples like the following:

```
Year ({file.DATE}) < 1999
```

Selects those records in which the year found in the {file.DATE} field is earlier than 1999.

```
Year ({file.DATE}) > 1992 and Year ({file.DATE}) < 1996
```

Selects those records in which the year found in the {file.DATE} field falls between 1992 and 1996 (1992 and 1996 not included).

```
Year({file.DATE}) >= 1992 and Year({file.DATE}) <= 1996
```

Selects those records in which the year found in the {file.DATE} field falls between 1992 and 1996 (1992 and 1996 are included).

```
Month({file.DATE}) in 1 to 4
```

Selects those records in which the month found in the {file.DATE} field is one of the first four months of the year (includes January, February, March, and April).

```
Month({file.DATE}) in [1,4]
```

Selects those records in which the month found in the {file.DATE} field is the first or fourth month of the year (includes January and April, excludes February and March).

Selecting records using preset date ranges

The preset date ranges can be used to create selection formulas similar to these:

```
{file.DATE} in LastFullMonth
```

Selects those records in which the date found in the {file.DATE} field falls within the last full month. (If the month is May, this selects all records with an April date.)

```
not({file.DATE} in LastFullMonth)
```

Selects all records except those in which the date found in the {file.DATE} field falls within the last full month (if the month is May, this selects all records except those with an April date).

```
{file.DATE} < CurrentDate
```

Selects all records in which the date found in the {file.DATE} field falls before today's date.

Selecting records using date/number/character combinations

These formulas simply “mix and match” formulas from the categories above.

```
"C" in {file.FIELD}[1] and Month({file.DATE}) in [1,4]
```

Selects those records in which the value in the {file.FIELD} field begins with “C” and the month is either January or April. For example, if this kind of formula was used with an order database, you could be asking for a report showing all customers whose names begin with “C” and who placed orders in January or in April.

```
"AOK" in {file.HISTORY}[3 to 5] and {file.OPENCRED} >= 5000
```

Selects those records in which the {file.HISTORY} field shows the characters “AOK” as the 3, 4, and 5 characters and the {file.OPENCRED} field (the amount of available credit) is at least 5000.

These templates can be used as they are (with your own data), or they can be combined to create complex formulas.

Pushing down record selection to the database server

The drivers provided with Crystal Reports for SQL data sources allow “pushing down” record selection to the database server. When you specify a record selection formula in a report based on an SQL data source, Crystal Reports analyzes it, generates a SQL query from it and passes the SQL query to the database server. Record selection is then carried out in two stages:

- The first stage of record selection takes place when the database server processes the SQL query and returns a set of records to Crystal Reports.
- In the second stage, Crystal Reports further evaluates locally the record selection formula for the set of records returned from the database server.

Since database servers are usually faster machines than workstations, it is to your advantage to specify record selection formulas that can be processed by the server in the first stage. Doing so minimizes record selection on the local machine in the second stage. This process is commonly known as “pushing record selection down to the database server.” The following kinds of record selections can be pushed down to the server:

- Selections with indexed and non-indexed fields (indexed fields provide faster response).
- SQL queries with AND and OR clauses.
- SQL expression fields that carry out formula calculations for record selection. (For the types of SQL expressions supported by your SQL server, consult the documentation provided with the server.)

For complete details on pushing your record selection formulas down to the database server, see “Using enhanced record selection formulas” on page 98.

Troubleshooting record selection formulas

To troubleshoot your selection formula, you should first begin by making sure that all of the fields referenced in the selection formula are placed on your report. Then delete the selection formula and test it as you rebuild it, step by step.

To troubleshoot record selection formulas

1. Write down the record selection formula on paper. You will use this written copy to help you reconstruct the selection formula one step at a time.
2. Remove the record selection formula from your report by deleting the formula from the Record / Group Selection Formula Editor in the Formula Workshop.
3. Click **Close** when finished in the Formula Workshop.
4. Make certain that all fields referenced in the record selection formula (the selectors) are on the report physically and are not hidden.

   For example, if one of the selectors is:

   ```
   {customer.POSTAL CODE} > "80000"
   ```

   but the {customer.POSTAL CODE} field is not used on your report (as in the case of a sales report that uses the postal code to define territories but does not include the postal code in the report data), then insert the {customer.POSTAL CODE} field into the report.

   Or, if one of the fields referenced in the selection formula is on the report but is hidden, unhide it by deselecting the Suppress option in the Format Editor for that field.
5. Print the report and verify that the data in those fields which are referenced in the selection formula print satisfactorily. Make certain that all the data prints. For example, if there are *x* total records in the database you should have *x* records printing for each of the referenced fields. This establishes a baseline against which you can compare the results of printing with the selection formula.
6. When you are sure that you are getting satisfactory results without using the selection formula, you can enter the selection formula using only one of the selectors.

   For example, to use this as the final selection formula:

   ```
   {customer.POSTAL CODE} > "80000" and {customer.CONTACT LAST NAME}[1] =
   "C" and {customer.LAST YEAR'S SALES} >= 5000
   ```

   this formula will select all of those records that show a Postal code greater than 80000, a value in the {customer.CONTACT LAST NAME} field beginning with “C”, and a value in the {customer.LAST YEAR'S SALES} field greater than or equal to 5000.

   You might start with this as the first test selection formula:

   ```
   {customer.POSTAL CODE} > "80000"
   ```

Print the report and evaluate the data that prints when you have only one selector activated. Does the {customer.POSTAL CODE} field show only ZIP codes greater than 80000?

- If it does, then you know that this part of the selection formula is working.
- If it does not, then troubleshoot this part of the selection formula.

7 Once the selection formula with one selector activated is working properly, add a second selector. For example, the new selection formula might look like this:

```
{customer.POSTAL CODE} > "80000" and {customer.CONTACT LAST NAME}[1] = "C"
```

8 Preview the report and evaluate the data that prints when you have two selectors activated. Evaluate the data in the {customer.CONTACT LAST NAME} field (since you already evaluated {customer.POSTAL CODE} in the last step).

Does the {customer.CONTACT LAST NAME} field show only text strings beginning with the letter "C"?

- If it does, then this part of the selection formula is working.
- If it does not, then troubleshoot this part of the selection formula.

9 Once the selection formula with two selectors activated is working properly, add a third selector, then a fourth, and so on, until you have tested each selector in the selection formula.

Correcting selections that do not generate data

You may encounter a situation in which you create a record selection formula and, while header and footer information prints on your report, no detail information appears. The problem is the selection formula is rejecting all records. This usually occurs because of an error in the creation of the selection formula.

There are a couple of potential causes of your problem in the selection formula:

- See "Correcting uppercase/lowercase inconsistency" on page 116.
- See "Unwanted spaces appear in selection formula" on page 117.

Correcting uppercase/lowercase inconsistency

Record selection formulas are case sensitive. That is, "Bob" matches only with "Bob". It does not match with "bob", "BOB", "BoB", "bOB", "boB", or "BOb". Thus, if your selection formula is set to include only those records with "BOB" in the {customer.CONTACT FIRST NAME} field, but all the entries in the {customer.CONTACT FIRST NAME} field are mixed case ("Bob", for example), the selection formula will find no matches and thus will not print any details for the report.

You can solve this problem by using the UpperCase (str) or LowerCase (str) functions in your selection formula to convert field data to a consistent case before the program begins its selection. For example, if you were using this formula:

```
{customer.CONTACT FIRST NAME} = "BOB"
```

you can change the formula to this:

```
UpperCase({customer.CONTACT FIRST NAME}) = "BOB"
```

This second formula first converts the value of the {customer.CONTACT FIRST NAME} field to upper case characters and then checks to see if the resulting value in that field is equal to “BOB”. Using this formula, any instance of the three letters “b,” “o,” and “b” will be a match, regardless of case, because the case will be converted to uppercase.

You could use the LowerCase function in a similar manner to match with “bob.”

Check your selection formula closely and make sure you have the correct case for any text you are trying to match. If in doubt, use the UpperCase (or LowerCase) function to assure consistency and proper matching.

Another formula which performs much the same functions is the following:

```
"BOB" in UpperCase({customer.CONTACT FIRST NAME})
```

Unwanted spaces appear in selection formula

Spaces are characters, and when you include spaces in the search key of a record selection formula, the formula looks for records with the exact match in the selected field, spaces and all. For example, the following formula:

```
"Mr . " in {customer.TITLE}
```

will not find any matches with the form of address “Mr.” because there is an extra space in the search key between the letter “r” and the period. Likewise, “Ph. D” will not match “Ph.D”.

Check your selection formula closely, and make sure that the spaces in the selection formula match the spaces in the fields you are trying to match.

Sorting, Grouping, and Totaling 9

Sorting, grouping, and totaling are the steps that turn disorganized data into useful information on a report. This chapter describes the types of sorting, grouping, and totaling you can do within a report.

Sorting data

Sorting means placing data in some kind of order to help you find and evaluate it.

When you first insert a database field into your report, the data within the fields appears in the order in which it was originally entered into the database. Locating information in this kind of report is difficult. It is much easier to review or find information when you can see it sorted in a logical format. For example, you may want to have a customer list sorted alphabetically by name or by country.

Understanding sort options

When you sort, the program asks you to define two things:

- The field you want the sort to be based on (sort field).
- The sort direction.

Sort field

A sort field is the field that determines the order in which data appears on your report. Almost any field can be used as a sort field, including formula fields. A field's data type determines the method in which the data from that field is sorted.

Note: You cannot sort on memo or BLOB fields.

Field Type	Sort Order
Single-character string fields	blanks punctuation numbers uppercase letter lowercase letters
Multiple character string field	two letters three letters four letters, and so on For example: • "BOB" comes before "bob" • "123" comes before "124" • " " (blank) comes before "a" • "aa" comes before "aaa"
Currency fields	numeric order
Number fields	numeric order
Date fields	chronological order
DateTime fields	chronological order same-date values sorted by time

Field Type	Sort Order
Time fields	chronological order
Boolean comparison fields	False values (0) True values (1)
Null values	null values non-null values

Sort Direction

Direction refers to the order in which the values are displayed, once sorted.

- **Ascending**
  Ascending order means smallest to largest (1 to 9, A to Z, False to True). The program sorts the records in ascending order based on the values in the sort field you select.
- **Descending**
  Descending order means largest to smallest (9 to 1, Z to A, True to False). The program sorts the records in descending order based on the values in the sort field you select.

Sorting single and multiple fields

In single field sorting, all the records used in the report are sorted based on the values in a single field. Sorting an inventory report by stock number or sorting a customer list by customer number are examples of single field sorts.

In multiple field sorts, the Report Designer first sorts the records based on the values in the first field selected, putting them in ascending or descending order as specified. When two or more records have the same field value in the first sort field, the program then sorts those records based on the value in the second sort field.

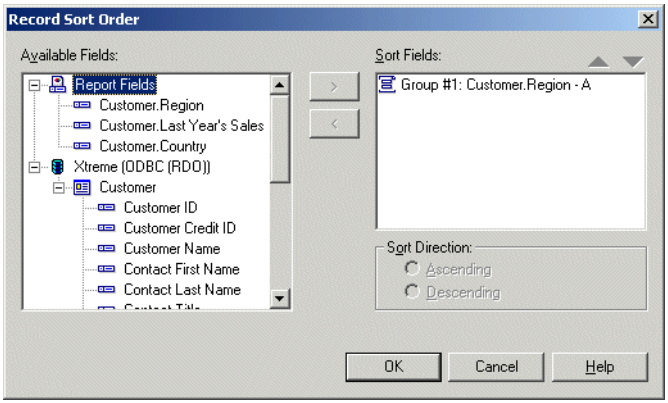
For example, if you choose to sort first by the {customer.COUNTRY} field and then by the {customer.REGION} field, both in ascending order, the report would appear with countries listed in alphabetic order, and regions within each country listed in alphabetic order. Any other fields, such as the postal codes within each region, would remain unsorted.

You create single or multiple field sorts using the same procedure.

To sort your data

1 On the **Report** menu, click **Record Sort Expert**.

The Record Sort Order dialog box appears.



2 Highlight the field to be sorted from the Available Fields area.

3 Click the > arrow.

The selected field is added to the Sort Fields list.

4 Specify the sort direction.

5 If sorting by more than one field, highlight the second field you want the data to be sorted by and add it to the Sort Fields list.

6 If you want to change the order of fields in the Sort Fields list, highlight the field you want to move and click the arrow buttons to move it up or down.

Tip: The order of the fields listed in the Sort fields box is the order by which data will be sorted.

7 As you add each field to the Sort Fields list, specify the sort direction.

8 Click **OK** when finished.

Records are sorted based on the values in the Sort Fields list.

Grouping data

Grouped data is data that is sorted and broken up into meaningful groups. In a customer list, for example, a group might consist of all those customers living in the same Zip Code, or in the same Region. In a sales report, a group might consist of all the orders placed by the same customer, or all of the orders generated by a particular sales representative.

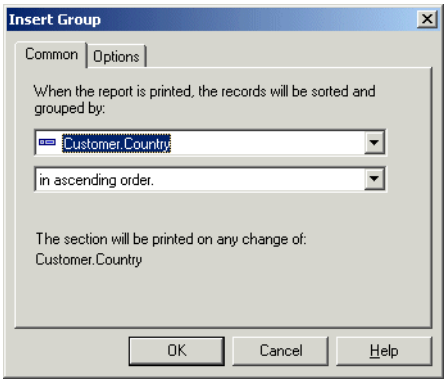
Group and sort direction

When data is grouped, four sort and group direction options are available. Direction refers to the order in which the values are displayed.

- **Ascending**
  Ascending order means smallest to largest (1 to 9, A to Z, False to True). The program sorts the records in ascending order and then begins a new group whenever the value changes.
- **Descending**
  Descending order means largest to smallest (9 to 1, Z to A, True to False). The program sorts the records in descending order and then begins a new group whenever the value changes.
- **Specified order**
  Specified order is a user-defined order. The program places each record into the custom group you specify, leaving the records in each group in original order or it sorts them in ascending or descending order, depending on your instructions. See "Creating custom groups" on page 124.
- **Original**
  Original order is the order the data was originally saved in the database. The program leaves the records in the order in which they appear in their originating database table, and begins a new group whenever the value changes in the group field you select.

To group data

1. On the **Insert** menu, click **Group**.
   The Insert Group dialog box appears.



2. Select the field you want the data grouped by from the top drop-down list.
3. Select the sort direction from the second drop-down list.

4 If you want to show a different value in the group header, click the **Options** tab.

Note: By default, the group header of the report will display the value of the field you are grouping on.

5 Select the **Customize Group Name Field** check box and choose a new group name.

For example, if you grouped by {Customer.CustomerID}, at each change of a group you will see the corresponding customer ID. If you want to display a different value (customer ID as well as customer name), customize the group name field by choosing an alternate data field, or creating a formula.

Tip: To hide the group header name, right-click the group header, select Format Field, and click Suppress on the Common tab of the Format Editor.

6 Click **OK**.

If your records within each group are unsorted, you need to sort the records within each group. See "Sorting records within groups" on page 125.

Related topics

"Grouping data in intervals" on page 130

"Creating group headers" on page 142

Creating custom groups

Usually data is sorted based on the values from a field in the report. Sometimes, however, you may not want to group data based on the values found in one of the fields on your report. For example:

- The field you want to group on does not exist.

  For example, your report may contain a City field and a State field, but no Country field, although you need to group by country.
- The field exists, but you do not want to group on the values in that field.

  For example, you may have a Color field on your report that includes specific color names (Logan Green, Sky Blue, Emerald Green, Navy Blue, and so on) but you want all shades of each color to appear as a single group (Greens, Blues, Reds, and so on). In this case you can build custom groups and manually assign the records you want to be in each group.
- The field exists, but you want to select specific values or ranges of values for each group.

  For example, you might want one group to contain records where gross sales are less than a certain value, a second group where gross sales are greater than a certain value, and a final group where gross sales fall between two values. In this case, you can build your groups using the same range of selection facilities that are available to you for building record selection queries.

Specified order grouping provides a solution to these custom sorting and grouping challenges. It enables you to create both the customized groups that will appear on a report and the records that each group will contain. The only limitation is that a record can be assigned to only one group.

To follow a tutorial where you create custom groups to rank customers by the amount of business they did in the previous year, see "Grouping data in intervals" on page 130.

To create a custom group

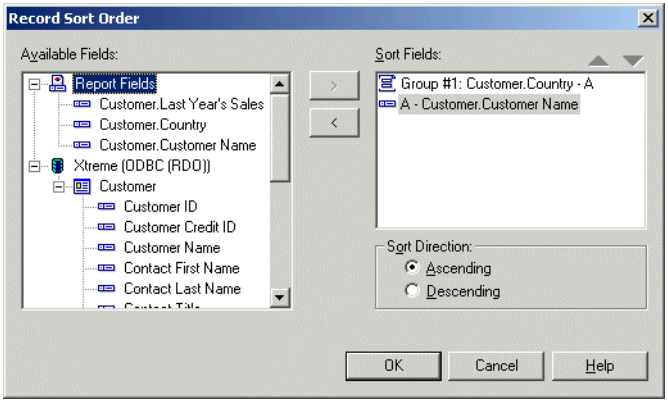
1. On the **Insert** menu, click **Group**.
   The Insert Group dialog box appears.
2. Select the field you want the data grouped by from the top drop-down list.
3. Select **in specified order** as your sort option from the second drop-down list.
4. On the **Specified Order** tab, enter the name of the group in the **Named Group** field.
5. Click **New**.
6. In the Define Named Group dialog box, use the drop-down lists to select the data to be part of the group.
7. Click the **<New>** tab to add more selection criteria to your specified group, if necessary.
8. Click **OK**.
9. Click **New** to create more custom groups as necessary.
10. Click the **Others** tab to specify how you want to organize the data that is not part of the group(s) you defined.
11. Click **OK**.

Sorting records within groups

Once you have grouped your data, you can easily sort the records within the groups to further organize the information.

To sort records within groups

1 Once the data is grouped, on the **Report** menu, click **Record Sort Expert**.
The Record Sort Order dialog box appears.



Note: Sort fields that begin with "Group" specify that the sort was done automatically when the data was grouped.

2 Highlight the field you want the records within the groups sorted by and click the > arrow to add it to the **Sort Fields** list.

Tip: The order of the fields in the Sort Fields box is the order by which data will be sorted.

3 Specify the sort direction.

4 Click **OK** when finished.

Group selection

When you group or summarize data, all the groups in the report are included by default. There may be times, however, when you do not want to include all the groups. For example:

- You might want to see only those groups that have certain group names, or whose summarized values meet a certain condition.
- You might want to see only the groups with the highest summary values, or the lowest.

You can select the groups that appear in the report in two different ways:

- See "Using the Select Expert" on page 127.
- See "Using selection formulas" on page 128.

Note: For faster results, limit records through record selection before creating groups. See "Selecting records" on page 108.

Using the Select Expert

The Select Expert can be used to select groups of records in the same way that you select individual records.

When you are setting up group selection criteria, instead of basing the selection criteria on standard fields, as you do for record selection, you base the criteria on group name fields or summary fields.

- If you have grouped your data but have not summarized it, you can only set up group selection based on the group name field. For example, you may want to select only those groups whose Region is Massachusetts:

  ```
  GroupName ({Customer.REGION}) = "MA"
  ```

- If you have summarized your data, you can set up group selection based on either the group name field or the summary field. For example:

  ```
  Sum({Customer.LAST YEAR'S SALES}, {Customer.REGION}) > 10000
  ```

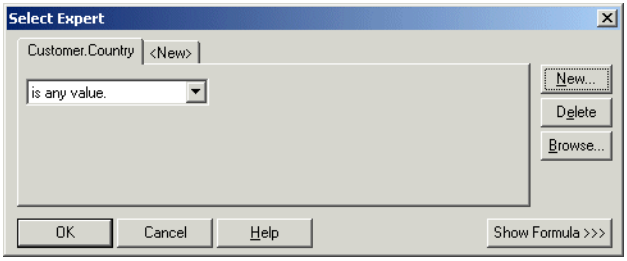
Note: The Select Expert can be used to set up record selection and group selection requests. When a group name or summary field is selected, the program knows that the selection criteria you set up is intended for group selection. In all other cases, the program knows that you are setting up record selection.

To set up group selection using the Select Expert

1. Right-click the summary field on which you want to base group selection and choose **Select Expert** from the shortcut menu.

   The Select Expert dialog box appears.

   Note: If you click the Select Expert button without first highlighting a summary field in your report, the Choose Field dialog box appears.



2. Use the drop-down list to enter your selection criteria for the indicated field.
3. To base the group selection on more than one field, click the **New** tab and choose the next field from the Choose Field dialog box.

Note: If you have not already previewed the report or refreshed the data, there will not be any data saved with the report. Without the data, the program cannot calculate group values; thus, no values appear when you click the arrow in the right drop-down list. In this situation, you will have to type in the values you want. If you want real values to work with, you will need to preview your report first. This will calculate the actual summary values available for you to work with.

4 When you are finished, click **OK** in the Select Expert to return to the report.

Using selection formulas

With the Group Selection Formula Editor, you can build your group selection request using group fields, group name fields, and other formulas. As with record selection formulas, your only restriction is that the formula you create must be Boolean; that is, it must return either a True or False value.

To create a record or group selection formula

1 On the Report menu, point to **Selection Formulas**.

2 Click **Record** to create a record selection formula.

- or -

Click **Group** to create a group selection formula.

The Formula Workshop appears.

3 Enter your selection formula in the Group Selection Formula Editor.

Note: The resulting formula must be Boolean; that is, it must return either a True or False value.

4 Click **Check** to identify any errors in the formula.

5 Fix any syntax errors the Formula Checker identifies.

6 When the formula has the correct syntax, click **Save**.

When the program runs the report, it will include only those records or groups of records that you specified.

Troubleshooting group selection formulas

In some cases, no values will print when using a group selection formula, even though there are values that match the selection criteria. Typically, in these cases:

- The group selection formula references another formula.
- The referenced formula is one that calculates the value of each group as a percentage of the total value of all groups (in other words a subtotal as a percentage of a grand total).

To correct a group selection formula

1 Use **Xtreme.mdb** to create a report that includes the following fields:

```
{customer.CUSTOMER NAME}
{customer.REGION}
{orders.ORDER ID}
{orders.ORDER AMOUNT}
```

For each order, the report shows the customer that placed the order, the region in which that customer is located, the order ID number, and the amount of the order.

2 Group the report by the {customer.REGION} field.

3 Insert a summary that subtotals the {orders.ORDER AMOUNT} field for each {customer.REGION} group.

The program calculates a subtotal in the {orders.ORDER AMOUNT} field every time the region changes. See "Subtotaling data" on page 139.

4 Insert a grand total on the {orders.ORDER AMOUNT} field to see the total value of all orders placed.

5 Create a formula named Percent that calculates each subtotal as a percentage of the grand total to see the value of the orders for each region group as a percentage of all orders placed.

```
Sum({orders.ORDER AMOUNT}, {customer.REGION})
     % Sum({orders.ORDER AMOUNT})
```

6 Place the formula in the Group Footer section of the report.

7 Reference the formula (@Percent) in a group selection formula that selects only those groups for which the percentage (of subtotal to grand total) is less than 5% in order to find out which regions individually contributed less than 5% of total sales:

```
{@Percent} < 5
```

When you click **Check**, you will receive the following error message:

```
This formula can not be used because it must be evaluated later.
```

8 Instead of using the formula name (in this case @Percent) in the group selection formula, enter the formula itself (the formula named @Percent). Thus, instead of using the group selection formula:

```
{@Percent} < 5
```

use the group selection formula:

```
Sum({orders.ORDER AMOUNT}, {customer.REGION})
     % Sum({orders.ORDER AMOUNT}) < 5
```

Now when you print, only the regions that contributed less than 5% will print.

Grouping data in intervals

You may want to group your data into intervals. Age groups, time periods, and sales categories are some of the interval groupings that can be created. In this example, you will rank customers by the amount of business they did in the previous year.

This example uses specified order grouping. This kind of grouping lets you specify the records that will be included in each group. You define the intervals you want and the program will do the rest.

To group data in intervals

1. Create a report using the sample data, **Xtreme.mdb**, and place the following fields from left to right in the Details section:

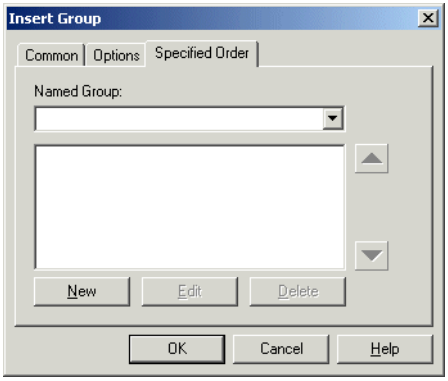
   ```
   {customer.CUSTOMER NAME}
   {customer.REGION}
   {customer.POSTAL CODE}
   {customer.COUNTRY}
   {customer.LAST YEAR'S SALES}
   ```

2. On the **Insert** menu, click **Group**.

3. Because you want to set up intervals based on the previous year's sales, select Last Year's Sales as the sort-and-group-by field from the drop-down list on the Insert Group dialog box.

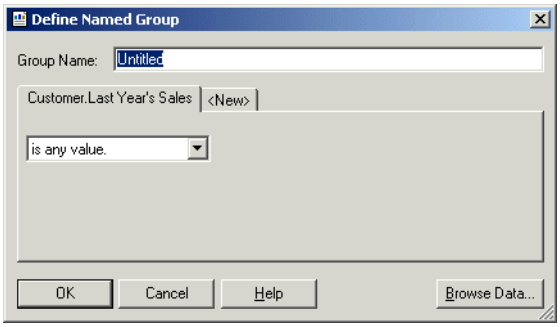
4. Select **in specified order** from the second drop-down list.

   The Specified Order tab appears in the Insert Group dialog box.



5. Click the **New** button.

The Define Named Group dialog box appears.



6 Type “Less than $10,000” in the **Group Name** field.

This is the name that will appear as the Group Name field value for the group.

7 Since the first group is to contain only those records that have a Last Year’s Sales figure of less than $10,000, set the fields so your condition reads:

`is less than 10000`

8 Click **OK** to return to the Specified Order Tab.

9 Click **New**.

The Define Named Group dialog box reappears.

10 Set up a second group that contains values between $10,000 and $25,000.

- Type “$10,000 to $25,000” in the Group Name field.
- Set the first field so the condition reads: `is between`.
- Specify a range of values:
  - Type “10000” in the top field.
  - Type “25000” in the bottom field.

You have now set up the group to contain all values between $10,000 and $25,000.

11 Click **OK** to return to the Specified Order tab.

12 Click **New**.

The Define Named Group dialog box reappears.

13 Set up the final group that contains all values over $25,000.

- Type “Over $25,000” in the Group Name field.
- Set the first field so the condition reads: `is greater than`.
- Type “25000”.

14 Click **OK** to return to the Specified Order Tab.

15 Click **OK**.

The report is grouped by interval in a specified order.

Grouping by the first letter of a company name

You might want to break your data into groups based on the first letter of the company name. For example, in a customer list you might want all the "A" customers in a group, then all the "B" customers, and so forth. To do this requires the use of a formula.

Do not worry if you are unfamiliar with formulas. This text will show you what formula you need here and how to enter it.

You can learn more about creating and editing formulas in "Formulas overview" on page 330.

You are going to create a formula that will extract the first letter of each customer's name. Then you will group the data using that formula field as the sort-and-group-by field. The program will sort the data based on the first letter of each customer name and start a new group whenever that letter changes.

To group data by the first letter of a company name

1. Create a report using the sample data, **Xtreme.mdb**, and place the following fields from left to right in the Details section:

   ```
   {customer.CUSTOMER NAME}
   {customer.REGION}
   {customer.POSTAL CODE}
   {customer.COUNTRY}
   ```

2. On the **View** menu, click **Field Explorer**.

   The Field Explorer dialog box appears.

3. Select **Formula Fields** and click **New**.
4. In the Formula Name dialog box, enter the name you want to identify the formula, for example "First Letter."
5. Click **Use Editor**.

   The Formula Workshop appears with the Formula Editor active.
6. Type the following formula into the Formula text box:

   ```
   {Customer.Customer Name}[1]
   ```
7. Click the **Close** button on the Formula Workshop.

   You return to your report.

8. On the Insert menu, click **Group**.

   The Insert Group dialog box appears.
9. Select the formula field as the field you want your data grouped by from the first drop-down list.
10. Select the sort direction from the second drop-down list.

11 Click **OK**.

You return to your report with the data grouped by the formula field as specified. The data is broken into groups based on the first letter in the customer's name. The formula provides a live header for every group.

For more information on live headers, see "Live headers" on page 143.

Related topics

"Formulas overview" on page 330

"Grouping data" on page 122

"Grouping data in intervals" on page 130

"Creating group headers" on page 142

Grouping data hierarchically

You have the option to group data in a report to show hierarchical relationships. When you group data hierarchically, you sort information based on the relationship between two fields.

For example, if you want to show the hierarchical structure of a department, you could group data by employee ID and specify the hierarchy using the data field listing whom the employee reports to.

To group data hierarchically

1 On the **Insert** menu, click **Group**.

2 In the Insert Group dialog box, select the field to be the basis of your hierarchy.

For example, if you want to view the hierarchical structure of a company's employees, select the employee ID data field. If you want to view the hierarchical structure of region sales offices, select office names.

3 Select **in ascending order**.

By default, the group header of the report will display the value of the field you are grouping on.

4 Click the **Options** tab and select the **Customize Group Name Field** check box if you want to show a different value in the group header.

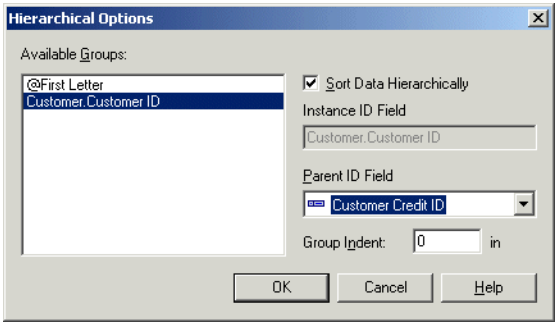
For example, if you grouped by {Employee.EmployeeID}, at each change of a group you will see the corresponding employee ID. If you want to display a different value (employee name instead of employee ID), customize the group name field by choosing an alternate data field, or creating a formula.

5 Click **OK**.

The group you created is added to the report.

6 On the **Report** menu, click **Hierarchical Grouping Options**.

The Hierarchical Options dialog box appears.



7 In the **Available Groups** list, select the group you want to organize hierarchically.

8 Select the **Sort Data Hierarchically** check box.

9 In the **Parent ID Field** list, select the field by which you want the Instance ID Field organized.

For instance, for a company hierarchical report, you might select the data field listing the supervisor to whom the employee reports.

Note: The Instance ID Field and Parent ID Field must be of the same data type. For example, if the Instance ID Field holds string data, then the Parent ID Field must also hold string data.

10 In the **Group Indent** field, enter the amount you want to indent for each subgroup.

11 Click **OK**.

The report data is now grouped hierarchically. If necessary, you can now calculate summary fields across your new hierarchical grouping. When inserting a subtotal, grand total, or summary in the usual manner, select the "Summarize across hierarchy" option. For more information, see "Summarizing grouped data" on page 135 and "Subtotaling" on page 139.

Note:

- The level of the hierarchy is determined by group instances that match the Instance ID and Parent ID.
- If a group instance is not connected to any Parent ID, it appears at the top of the hierarchy.

Editing groups

To edit a group

1. On the **Report** menu, click **Group Expert**.
2. In the **Group By** list of the Group Expert dialog box, select the group you want to edit.
3. Click **Options**.
4. In the Change Group Options dialog box, edit the group as necessary.
5. Click **OK** to close the Change Group Options dialog box and again to close the Group Expert dialog box.

   The report reflects the changes you have made to the group.

Summarizing grouped data

One of the primary purposes for breaking data into groups is to run calculations on each group of records instead of on all the records in the report.

When the program summarizes data, it sorts the data, breaks it into groups, and then summarizes the values in each group. It does this all automatically.

The program includes a number of summarizing options. Depending on the data type of the field you plan to summarize, you can:

- Sum the values in each group.
- Count all the values or only those values that are distinct from one another.
- Determine the maximum, minimum, average, or Nth largest value.
- Calculate up to two kinds of standard deviations and variances.

For example:

- Customer list reports: determine the number of customers in each state. The summary would count the distinct customers in each state group.
- Order reports: determine the average order placed each month. The summary would calculate the size of the average order for each month group.
- Sales reports: determine the total sales per sales representative. The summary would sum or subtotal the order amounts for each sales representative group.

Note: You can also calculate summary fields across hierarchical groupings. To do so, select “Summarize across hierarchy” in the Insert Summary dialog box.

To summarize grouped data

1. On the **Insert** menu, click **Summary**.

   The Insert Summary dialog box appears.

2. Select the desired field to summarize from the **Choose the field to summarize** list.

3. Select a summary operation from the **Calculate this summary** list.

4. Select a location in which to place the summary from the **Summary location** list.

   Tip: You can create a new group for you report by clicking the Insert Group button.

5. If you want to display your summary value as a percentage of a total, select **Show as a percentage of** from the Options area, and then select a total field from the list.

   For more information about percentages, see "Percentages" on page 141.

6. If you want to summarize across a hierarchy, select **Summarize across hierarchy**.

   For more information about hierarchies, see "Grouping data hierarchically" on page 133.

7. Click **OK** when finished.

Related topics

"Subtotaling data" on page 139

Ordering groups by summarized values

Groups can be organized in either ascending or descending order by summary value. For example, in an orders report, if you subtotal the order amount by state, you could order the groups:

- From lowest to highest order amount (ascending).
- From highest to lowest order amount (descending).

You order the groups in a report by summary values using the Group Sort Expert command on the Report menu.

To order groups by summary value

1. Click **Group Sort Expert** on the Expert Tools toolbar.

   The Group Sort Expert dialog box appears with a tab for each of the groups in the report that has a summary.

2. Click the tab for the group you want to sort.

3. Select the **All** option from the drop-down list on the left.

4 Choose the summary that you want to base your selection on from the summary drop-down list on the right.

The summary drop-down list on the right is for those cases in which you have multiple summaries within a single group section. For example, in an orders report, you might sum and average the orders for each customer and then display both the sum and the average in the same group section. In such a case, you would select the sum or the average from this drop-down list.

5 Specify the sort direction.

6 To select a second group sort, repeat Steps 2-5.

When you run the report, the program will order the groups based on the summary value(s) specified.

Selecting top or bottom N groups or percentages

At times, you might want to show only the top or bottom groups or percentage values in a report: the fastest selling product lines, the countries that account for the top 25 percent of sales, the states that generate the most orders, and so on. Because this kind of selection is so popular, the program includes the Group Sort Expert for setting it up easily.

There is one other element to consider when setting up top N: what to do with all the records from other groups that do not fit the top N or bottom N criteria you have set. You need to decide whether to eliminate those records from your report entirely or to lump them all together into a single group. The program enables you to do choose either option.

Note: A report must contain a summary value in order to be able to perform a top N or bottom N selection. See "Summarizing grouped data" on page 135.

To select the top or bottom N groups

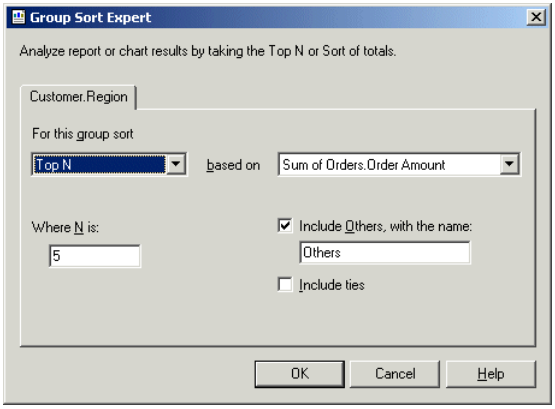
Note: This procedure shows you how to select top or bottom N groups. Top or bottom percentages work the same way except that you define the percentage value instead of the number of groups.

1 Create a report and summarize the data as desired. When you summarize the data, the program breaks the data into groups and summarizes each group.
- With top N groups, you are instructing the program to display those groups that have the highest summary values (Top N).
- With bottom N groups, you are instructing the program to display those groups that have the lowest summary values (Bottom N).

2 Click the **Group Sort Expert** button on Expert Tools toolbar.

The Group Sort Expert appears with a tab for your group.

Note: If you have multiple groups, the program will display a tab for each of the groups.



3 Choose **Top N** or **Bottom N** from the first drop-down list.

4 Choose the summary that you want to base your selection on from the summary drop-down list on the right.

The summary drop-down list on the right is for those cases in which you have multiple summaries within a single group section. For example, in an orders report, you might sum and average the orders for each customer and then display both the sum and the average in the same group section. In such a case, you would select the sum or the average from this drop-down list.

5 In the **Where N is** text box, enter the number of groups you want to display. For example:

- To report on the three fastest selling product lines, set N to be equal to three.
- To report on the five least productive sales regions, set N to be equal to five.

6 Click the **Include Others, with the name** check box and enter a name if you want to group all the other records into a single group.

7 Select **Include ties** to accommodate groups whose summarized values are equal.

For example, suppose you had the following groups:

- Order 1 = 100
- Order 2 = 90
- Order 3 = 80
- Order 4 = 80

If you set your top N to be three, but you *do not* select “Include ties,” your report will show Order 1, Order 2, and Order 3.

If, in the same scenario, you *do* select “Include ties,” your report will show Order 1, Order 2, Order 3, and Order 4 even though N is set as three. In this way, the program accommodates the equal values of orders 3 and 4.

8 Click **OK** when finished.

When the program runs the report, it will include only those groups that you specified.

Subtotaling

A subtotal is a summary that totals or sums numeric values in a group.

Note: If you are creating a subtotal using database tables that are grouped in a one-to-many linking relationship, you may need to use a running total instead of a subtotal. See "Creating running totals in a one-to-many linking relationship" on page 152.

Subtotaling data

In this example you will subtotal Last Year's Sales by Country.

To subtotal data

1 Create a report using the sample data, **Xtreme.mdb**, and place the following fields from left to right in the Details section:

```
{customer.CUSTOMER NAME}
{customer.REGION}
{customer.POSTAL CODE}
{customer.COUNTRY}
{customer.LAST YEAR'S SALES}
```

2 Right-click the Last Year's Sales field, point to **Insert** and choose **Summary** from the shortcut menu.

The Insert Summary dialog box appears with the chosen field listed as the field to be summarized.

3 Click **Insert Group**.

The Insert Group dialog box appears so you can specify the group you want to add to your report.

4 Choose the field you want the data grouped by, specify a sort direction and then click **OK** when finished.

5 On the Insert Summary dialog box, select the group you just created from the **Summary location** list and then click **OK**.

The values in each group are now subtotaled.

Extending prices and subtotaling the extensions

In an orders report or invoice, you may need to extend the prices for individual line items and then subtotal the extensions. You do this by using a simple formula to extend the prices, and then you subtotal the formula field.

To extend the price and subtotal the extensions

1. Create a report using the sample data, **Xtreme.mdb**, and place the following fields from left to right in the Details section:

   ```
   {orders.CUSTOMER ID}
   {orders detail.PRODUCT ID}
   {orders detail.QUANTITY}
   {orders detail.UNIT PRICE}
   ```

2. To create the formula for extending the prices, go to the **View** menu and select **Field Explorer**.

   The Field Explorer dialog box appears.

3. Select **Formula Fields** and click **New**.

   The Formula Name dialog box appears.

4. Enter the name you want to identify the formula.

5. Click **Use Editor**.

   The Formula Workshop appears with the Formula Editor active.

6. Enter the following formula into the Formula text box:

   ```
   {Orders_Detail.Quantity} * {orders detail.Unit Price}
   ```

7. Click the **Close** button on the Formula Workshop.

   You return to the Field Explorer dialog box with the name of your formula listed in the Formula Fields node.

8. Drag your formula field to the right of the Unit Price field in the Details section of the report.

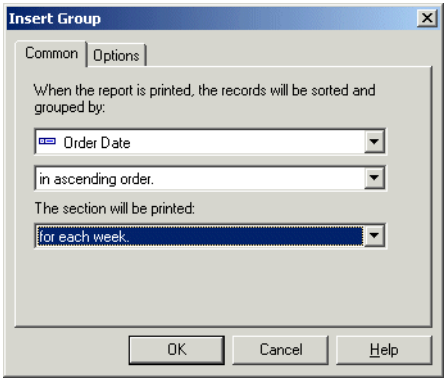
9. To subtotal your extension formula, right-click the formula field, point to **Insert** and choose **Summary** from the shortcut menu.

   The Insert Summary dialog box appears.

10. Click **Insert Group** and create a group on the {orders.ORDER DATE} field.

11. Choose **for each week** as the group interval.

Note: "The section will be printed" drop-down box will not be activated until you choose the Order Date field.



12 Click **OK** to return to the Insert Summary dialog box

13 Select the group you just created from the **Summary location** list and then click **OK**.

Your data will be sorted by date and grouped in one week intervals.

Percentages

Calculating a percentage

You can calculate the percentage of one group within a broader grouping. For example, you can show the percentage of sales in each city based on the total sales for each country. Or, you can see what percentage of the grand total of sales each country contributes.

To calculate a percentage

1 On the **Insert** menu, click **Summary**.

The Insert Summary dialog box appears.

2 Select the field for which you want to calculate the sum.

For example, you may want to insert a field which calculates the sum of last year's sales.

3 Select **Sum** in the **Calculate this summary** list.

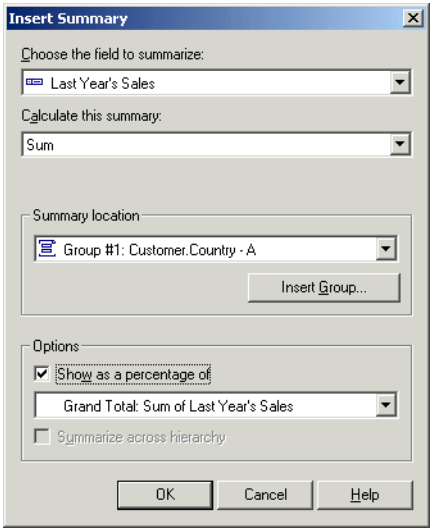
4 Select the summary location.

Note: The summary location cannot be Grand Total (Report Footer) when you are calculating a percentage.

5 Click the **Show as a percentage of** check box.

6 Select the group you want the percentage based on.

You can choose to show a percentage of a group within another group, or show a percentage of the grand total.



7 Click **OK**. The summary percentage field is added to your report.

Group headers

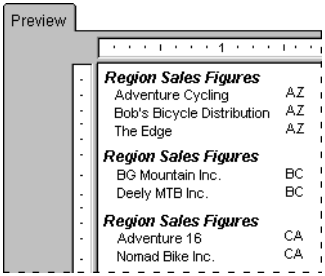
Creating group headers

Whenever you create a group, a subtotal, or a summary, the program creates both a Group Footer (GF) section (where it places any subtotal or summary value), and a Group Header (GH) section (where it automatically places the group name/header). Group Headers are useful, even necessary, if you want your report data to be clear and easily understood. Though the program creates a group header automatically, you may find that you would like to change or modify the header to suit your needs. You will learn how to create the four most common kinds of group headers:

- "Standard headers" on page 143
- "Live headers" on page 143
- "Live headers for groups based on a formula" on page 144
- "Headers for custom groups" on page 145

Standard headers

A standard header is a block of text that is used to identify each group in a rather generic kind of way. “Customer,” “State,” and “Monthly Orders” are all examples of this kind of header.



While the header is somewhat descriptive (“Region Sales Figures” means it is a regional group), you never know what region is in the group without first looking at the details in the group.

To create a standard header

1. Click the **Insert Text Object** button on the Insert Tools toolbar.
2. When the object pointer appears, move the object frame into the Group Header section.
3. Enter the text you want to use for the header.
4. Click outside the frame when finished to complete the process. Now when you run the report, the same header will appear at the beginning of each group.

Live headers

A live header is a header that changes based on the content of the group. If the data is subtotaled by region, for example, a live header would typically identify the region detailed in each group. Thus, the Arizona group would have a header identifying the group as Arizona data, the California group would have a header identifying the group as California data, and so on.

Note: When creating a group, the program automatically inserts a group name field in the Group Header section unless you have toggled the option off using the Options command on the File menu. The information that follows details how to manually insert such a section (if you do not have the program insert one automatically) and how to create different kinds of live headers for different needs.

Group name only

The easiest live header to create is one based on identifying the value of the group field.

To create a live header by group name only

1. On the **View** menu, click **Field Explorer**.
2. In the Field Explorer dialog box, expand the **Group Name Fields** folder.
3. Select the Group Name field that matches the group you are working with and drag it into the Group Header section for that group.

   When you run the report, the group field value identifier will appear as the group header for each region group.

Group name with text

A more complex type of live header combines a field value and text. A typical group header of this kind for data broken down by region would be, “Sales for California” or “Customers in Postal Code 60606“. Creating these headers involves three steps:

- Insert a text object in the Group Header section.
- Type in the text you want to appear.
- Enter the Group Name field in the text field where you want it to appear in the Group Header.

For example, if you want your header to read “Sales for” and then the name of the region in the current group (Sales for AZ, Sales for CA, and so forth), follow these steps:

To create a live header with group name using text

1. Click the **Insert Text Object** button on the Insert Tools toolbar.
2. Place the object frame into the Group Header section for the group.
3. Type in the desired text with a space after it.

4. Click the **Field Explorer** button on the Standard toolbar.
5. In the Field Explorer dialog box, expand the **Group Name Fields** folder.
6. Select the Group Name field that matches the group you are working with and drag it into the text object, immediately after the text and the space you entered.

   Tip: Expand the size of the text box to fit both the text and the group field.
7. Format the text object as you want it to appear.

When you run the report, the program will create a live header (with text) for each of your groups.

Live headers for groups based on a formula

When you create a group and use a formula field as a sort-and-group-by field, the program automatically creates a group name field based on the value returned by the formula.

For example, if you create this formula:

```
{customer.CUSTOMER NAME}[1]
```

and then group on the formula, the program will group your data based on the first letter in the Customer Name field.

To create a live group header for a group based on a formula, simply insert the group name field in the Group Header section.

When you run the report, the "A" group will have the letter "A" as a header, the "B" group will have the letter "B", and so on. For more information, see "Grouping by the first letter of a company name" on page 132 and "Grouping data in intervals" on page 130.

To create a more descriptive header such as "Customers beginning with the letter B", see "Group name with text" on page 144.

Headers for custom groups

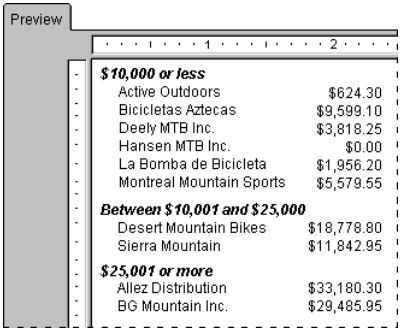
The final type of header is a header for the types of custom groups created when data is grouped in a specified order. When using specified order grouping, both the name for each group and the records that belong in it are specified. As in the other grouping situations, the program creates a group name field for each group based on the group names specified.

To create a header for custom groups

1. On the **View** menu, click **Field Explorer**.
2. In the Field Explorer dialog box, expand the **Group Name Fields** folder.
3. Select the Group Name field for the custom group and drag it into the Group Header section for that group.

The program automatically applies each of the group names you assigned to the appropriate groups.

Note: Make certain that when you assign the names to the groups using the Define Named Group dialog box, the names you assign are the names you want to appear as group headers.



Suppressing group headers

You have the option to hide group headers in your report.

To suppress group headers

1. Right-click the group header and select **Format Field**.
2. In the Format Editor, on the Common tab, click the **Suppress** check box.
3. Click **OK**.

   Tip: To show the group header again, clear the Suppress check box.

Drilling-down on group headers

To make report viewing easier, you can hide the details of your report and only have the group headers visible. When necessary, you can click on the group header to view the report details.

To drill-down on group headers

1. Right-click the grey Details section to the left of the report.
2. Select **Hide (Drill-Down OK)**.

3. Click **Refresh**.

   When you place the cursor over a group header, the cursor becomes a magnifying glass.
4. Double-click the group header to drill-down to the detail information.

   A drill-down tab appears in the Report Designer. Click the Design or Preview tab to return to that view.

Related topics

"Using the drill-down option on summarized data" on page 75

Running Totals 10

Running totals are a flexible and powerful way to create specialized summaries and continually incrementing totals.

This chapter shows you how to add a basic running total and a running total within a group to your report. You will also learn how to create conditional running totals and running totals using formulas.

Understanding running totals

Running total fields are similar to summary fields but allow more control over how the total is calculated and when it is reset. Running total fields are specifically suited to perform the following totaling functions:

- Show values of a total accumulate as it is calculated record by record.
- Total a value independent of the report's grouping.
- Total a value conditionally.
- Total a value after a group selection formula has been applied.
- Total a value from the driving table in a one-to-many linking relationship.

How running totals work

A running total field is created with the Running Total Expert. The Running Total Expert creates a running total field by asking you to select a field to summarize, the summary operation to use, a condition upon which to base the evaluation, and a condition upon which to reset the evaluation.

Note: A running total field can be used on database fields and first-pass formulas, but cannot be used on second-pass formulas or formulas that reference other second-pass formulas. For more information see "Multi-pass reporting flow chart" on page 420.

Placement of running total fields

The calculation of a Running Total field is determined by the settings selected in the Running Total Expert. However, where you place the running total affects the value that appears on the report. For example, if a Running Total field that evaluates every record and never resets (a grand total) in the Report Header, only the value of the first record will appear. Placing the same Running Total field in the Report Footer returns the desired value. The Running Total field is properly calculated in both cases, but it is displayed too soon in the first case.

The following list summarizes the records that are included in the calculation when a running total is placed in various report sections. This list assumes the running total is not reset.

Report Header	The first record in the report
Page Header	All records up to and including the first record on the current page
Group Header	All records up to and including the first record in the current group
Details	All records up to and including the current record
Group Footer	All records up to and including the last record in the current group
Page Footer	All records up to and including the last record on the current page
Report Footer	All records in the report

Creating running totals

Creating running totals in a list

Running totals are totals that can be displayed on a record by record basis. They total all records (in the report, in the group, and so on) up to and including the current record.

The most basic form of a running total is a running total maintained throughout a list. In this tutorial, you will create this kind of report by setting up a running total for a list of order amounts.

Note: Running total fields are prefixed by the # sign.

To create a running total in a list

1. To get started, create a report using the sample database, **Xtreme.mdb**. Link the Customer and Orders tables, and then place the following fields from left to right in the Details section:

   ```
   {customer.CUSTOMER NAME}
   {orders.ORDER ID}
   {orders.ORDER AMOUNT}
   ```

2. On the **View** menu, click **Field Explorer**.

   The Field Explorer dialog box appears.

3. Select **Running Total Fields** and click **New**.

   The Create Running Total Field dialog box appears.

4. Enter the name “TotalOrders” in the **Running Total Name** box.
5. Highlight {orders.ORDER AMOUNT} in the **Available Tables and Fields** box, and use the first arrow button to move it over to the **Field to summarize** box.
6. Select **sum** from the **Type of summary** list.
7. In the **Evaluate** section of the dialog box, click **On change of field**, and select {orders.ORDER ID} as the **On change of** field.

   The running total will execute each time this field changes.

8. In the **Reset** section of the dialog box, click **Never** (this gives you a running total that never resets; that is, the running total continues throughout the report).
9. Click **OK** to save the running total field.

   The program returns you to the Field Explorer dialog box.

10. Insert the running total field in the Details section of the report, just to the right of {orders.ORDER AMOUNT}.

    On your report, each row in the running total column displays the current record value added to the previous values. This total continues, unbroken, through the report.

Creating running totals for a group

Another common use for running totals is tallying items in a group. The running total starts with the first item in the group and ends with the last. Then it starts all over again for the next group, then the next, and so on.

In this tutorial, you will create a report that:

- Maintains a running total of customer orders.
- Groups customer orders and resets the running total for each group.
- Displays the subtotal for each customer group.

To create a running total for a group

1. To get started, create a report using the sample data, **Xtreme.mdb**.

   Link the Customers and Orders tables and place the following fields from left to right in the Details section:

   ```
   {customer.CUSTOMER NAME}
   {orders.ORDER ID}
   {orders.ORDER AMOUNT}
   ```

2. On the **Insert** menu, click **Group** and group on the {customer.CUSTOMER NAME} field.

3. On the **View** menu, click **Field Explorer**.

   The Field Explorer dialog box appears.

4. Select **Running Total Fields** and click **New**.

   The Create Running Total Field dialog box appears.

5. Enter the name “GroupRunningTotal” in the **Running Total Name** box.

6. Highlight {orders.ORDER AMOUNT} in the **Available Tables and Fields** box, and use the first arrow button to move it over to the **Field to summarize** box.

7. Select **sum** from the **Type of summary** list.

8. In the **Evaluate** section of the dialog box, click **For each record**.

9. In the **Reset** section of the dialog box, click **On change of group** and accept the default group name.

10. Click **OK** to save the running total field.

    You return to the Field Explorer dialog box.

11. Place the running total field in the Details section of your report, just to the right of the {orders.ORDER AMOUNT} field.

    Note: If you want to view a grand total of each group, place the running total field in the Group Footer section of your report.

Creating conditional running totals

There may be times when you have a list of values, and you only want to subtotal some of the values in the list. For example:

- You have a list that contains both Canadian and U.S. customers.
- You want to keep customer records sorted alphabetically based on customer name.
- You do not want to break the data into groups based on the country.
- You want a total of the values from just the Canadian records.
- You also want a total of the values from just the U.S. records.

To accomplish this, create two running totals, one to keep a running total of the U.S. records, and one to keep a running total of the Canadian records.

- **USTotal**
  maintains a running total of the U.S. records
- **CanadaTotal**
  maintains a running total of the Canadian records.

To create a conditional running total

1. To get started, create a report using the sample data, **Xtreme.mdb**. Place the following fields from left to right in the Details section:

   ```
   {customer.CUSTOMER NAME}
   {customer.COUNTRY}
   {customer.LAST YEAR'S SALES}
   ```

2. On the **Report** menu, click **Record Sort Expert**.

   Tip: Another way to do this is to click the Record Sort Expert button on the Expert Tools toolbar.

3. Sort the records based on the {customer.CUSTOMER NAME} field.
4. On the **View** menu, click **Field Explorer**.

   The Field Explorer dialog box appears.

5. Select **Running Total Fields** and click **New**.

   The Create Running Total Field dialog box appears.

6. Enter the name “USTotal” in the **Running Total Name** box.
7. Highlight {Customer.LAST YEAR’S SALES} in the **Available Tables and Fields** box, and use the first arrow button to move it over to the **Field to summarize** box.
8. Select **sum** from the **Type of summary** list.

9. In the **Evaluate** section of the dialog box, click **Use a formula** and then click the **Formula** button.

   The Formula Workshop appears with Running Total Condition Formula active.

10 Enter the following formula in the Formula box:

```
{Customer.Country} = "USA"
```

This tells the program to evaluate the running total each time it comes to a record where {Customer.COUNTRY} is equal to “USA.” The running total will ignore all other records (such as records for Canada).

11 When the formula has the correct syntax, click **Save and close**.

You return to the Create Running Total Field dialog box.

12 In the **Reset** section of the dialog box, click **Never**.

13 Click **OK** to save the running total field.

The program returns you to the Field Explorer dialog box.

14 Place the running total field in the Details section of your report.

15 Now create the “CanadaTotal” running total field using the process outlined in steps 5-13. The only difference is that this time you will set the evaluation formula to:

```
{Customer.Country} = "Canada"
```

16 When you are finished, place the **#CanadaTotal** field in the Details section of your report, just to the right of the {Customer.LAST YEAR’S SALES} field.

Note: If you only want to see a grand total of the Canadian and American sales, place the two running total fields you created in the Report Footer section of your report.

Creating running totals in a one-to-many linking relationship

A one-to-many linking relationship occurs in linked tables when a single record in one table can be matched with many records in another table. For example, a one-to-many link might occur when you link a customer table to an orders table. It wouldn’t be unusual in such a case for each customer in the primary table to have many orders in the second (lookup) table. In your report, the field values from the primary table are repeated for each new field value in the lookup table.

Creating a subtotal on a field from the primary table provides an incorrect result because the repeated values are included in the calculation. You can avoid this problem by creating a running total.

This concept can be demonstrated with the sample Xtreme database by using the Customer and Orders tables.

To create a running total in a one-to-many linking relationship

1 From the Field Explorer dialog box, place the following fields from left to right in the Details section of your report:

```
{customer.CUSTOMER NAME}
{customer.LAST YEAR'S SALES}
{orders.ORDER ID}
{orders.ORDER AMOUNT}
```

2 On the Insert Tools toolbar, click **Insert Group** and create a group based on the {Customer.CUSTOMER NAME} field.

3 Right-click the {Customer.LAST YEAR'S SALES} field and select **Summary** from the **Insert** submenu.

4 Choose **Group #1: Customer.Customer Name - A** as the location for your summary.

If you look at the subtotals for each group, you will notice they aren't accurate. This is because the {Customer.LAST YEAR'S SALES} field is duplicated for each order in the report. Follow the rest of this procedure to see how a running total produces an accurate result in the same situation.

5 In the Field Explorer dialog box, select **Running Total Fields** and click **New**.

The Create Running Total Field dialog box appears.

6 Enter the name "LYSrunning" in the **Running Total Name** box.

7 Highlight {Customer.LAST YEAR'S SALES} in the **Available Tables and Fields** box, and use the first arrow button to move it over to the **Field to summarize** box.

8 Select **sum** from the **Type of Summary** list.

9 In the **Evaluate** section of the dialog box, click **On change of field** and add the {Customer.CUSTOMER NAME} field from the **Available Tables and Fields** box.

10 In the **Reset** section of the dialog box, click **On change of group** and choose **Group #1: Customer.Customer Name - A**.

11 Click **OK** to save the Running Total field.

12 Place the running total in the **Group Footer** section.

Compare the running total amount with the subtotal amount for each group. You will see the running total is accurate while the subtotal is not.

Creating running totals using a formula

If you have suppressed data, or your data is based on a formula that occurs WhilePrintingRecords, you should create a running total formula rather than using the Create Running Total Field dialog box.

When you create a running total manually, you need to create three formulas:

- A summary formula.
- A reset formula to set a variable to zero.
- A display formula to display the variable.

In the following procedure, you will create a report that performs the following functions:

- Maintains a running total of customer orders.
- Groups customer orders and resets the running total for each group.
- Displays the subtotal for each order (the last running total for that order).

To create running totals using a formula

1. Create a report using the sample data, **Xtreme.mdb**. Link the Customers and Orders tables and place the following fields from left to right in the Details section:

```
{customer.CUSTOMER NAME}
{orders.ORDER ID}
{orders.ORDER AMOUNT}
```

2. On the **View** menu, click **Field Explorer.**

3. Select **Formula Fields** in the Field Explorer dialog box and click **New**.
4. Name the formula “RunningTotal” and click **Use Editor**.

   The Formula Workshop appears with Formula Editor active.
5. Enter the following into the Formula box:

```
WhilePrintingRecords;
CurrencyVar Amount;
Amount := Amount + {orders.ORDER AMOUNT};
```

6. Click the **Close** button on the Formula Workshop.
7. Place this formula in the Details section of your report, just to the right of the {orders.ORDER AMOUNT} field.

   This formula prints the running total of the values in the Order Amount field.

8 On the **Insert** menu, click **Group** and group the report on the {customer.CUSTOMER NAME} field.

9 In the Formula Workshop, create “AmountReset”:

```
WhilePrintingRecords;
CurrencyVar Amount := 0;
```

This formula says:

Set the value in the Amount variable to 0.

10 Place this formula in the Group Header #1 section of your report.

Because the Group Header #1 section appears once for every group, @AmountReset will execute each time the group changes. Thus, the Amount variable is reset to 0 each time a new group begins.

11 Select the @AmountReset formula on the report and use the Format Editor to suppress it so that it will not appear in the final print-out.

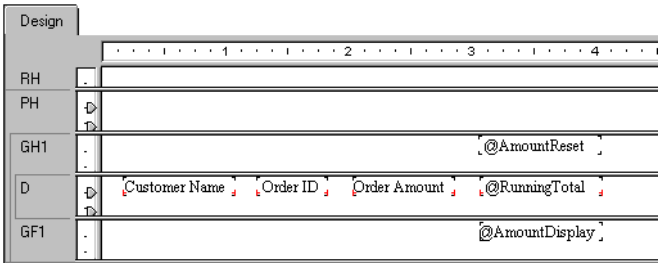
12 In the Formula Workshop, create “AmountDisplay”:

```
WhilePrintingRecords;
CurrencyVar Amount;
```

This formula simply displays the current value of the Amount variable at any time.

13 Place this formula in the Group Footer #1 section of your report.

Because the Group Footer #1 section appears once for every group, @AmountDisplay will execute each time a group ends. Thus, the value stored in the Amount variable will be printed each time the group changes.



Note: This formula prints the same value that @RunningTotal prints as the running total for the last record in each group. But since it is printing it in the Group Footer section, it acts as a group subtotal, not as a running total.

On your report, each row in the running total column displays the current record value added to the previous values. The running total starts fresh with each new group, and the final running total for each group becomes the subtotal for that group.

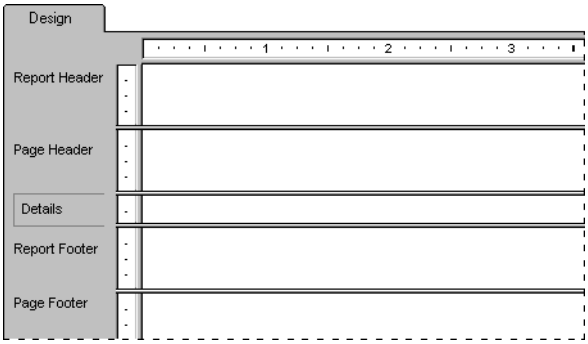
Multiple Section Reports 11

This chapter introduces the various types of sophisticated reports you can create using the multiple section reporting capabilities in Crystal Reports. These capabilities enable you to create reports that treat individual values differently based on sets of criteria you establish. These concepts are then applied to the creation of Form Letters.

About sections

Crystal Reports provides five design areas to use when building your report:

- Report Header
- Page Header
- Details
- Report Footer
- Page Footer



Each area contains a single section when you first create a new report. You cannot delete any of these original sections but you can hide them or add to them. Once you have added sections, you can delete them, move them in relation to other similar sections, or merge related sections together.

Working with sections

Many of the procedures in this section show you how to work with sections in the Section Expert. Sections can also be inserted, deleted, and so on by right-clicking the shaded area to the left of the section in the Design or Preview tabs and choosing the appropriate option from the shortcut menu.

Inserting a section

1 Click **Section Expert** on the Expert Tools toolbar.

The Section Expert appears with a list of all the sections in the report. When there are more than one of any kind of section, the sections are lettered A, B, C, and so on.

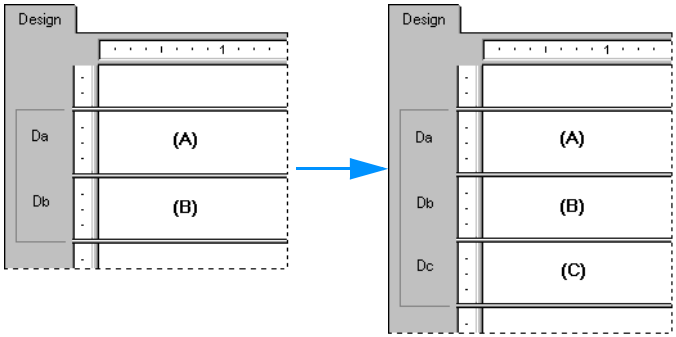
Note: The program enables only those options (free form, new page before, and so on) that apply to the highlighted section.

2 Highlight the section you want to insert a section after.

For example, to add another Details section, highlight the existing Details section.

3 Click **Insert**.

A new section is inserted immediately below the highlighted section.



Deleting a section

1 Click **Section Expert** on the Expert Tools toolbar.

The Section Expert appears with a list of all the sections in the report. When there are more than one of any kind of section, the sections are lettered A, B, C, and so on.

Note: The program enables only those options (free form, new page before, and so on) that apply to the highlighted section.

2 Highlight the section you want to delete.

3 Click **Delete**.

The program removes the highlighted section from the report.

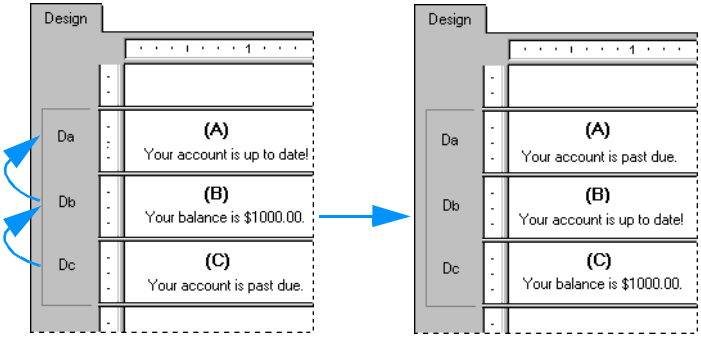
Moving a section

1 Click **Section Expert** on the Expert Tools toolbar.

The Section Expert appears with a list of all the sections in the report. When there are more than one of any kind of section, the sections are lettered A, B, C, and so on.

Note: The program enables only those options (free form, new page before, and so on) that apply to the highlighted section.

2 With Section (C) highlighted, click the up arrow twice.



The data originally in Section (C) is moved to Section (A). The data in the other sections is moved down.

Note:

- You can only move a section up or down within an area.
- The letters that identify the sections describe their relative (as opposed to original) position. Thus, if you move a “C” section up, it becomes a “B” section. It loses its original “C” designation.
- You can also move sections by dragging and dropping them in the Report Designer.

Merging two related sections

There may be times when you have placed objects in two sections (where they print sequentially) and you want to put them all in a single section (where they print simultaneously). You can merge the two sections and then rearrange the objects as needed in the new section.

To merge related sections

1 Click **Section Expert** on the Expert Tools toolbar.

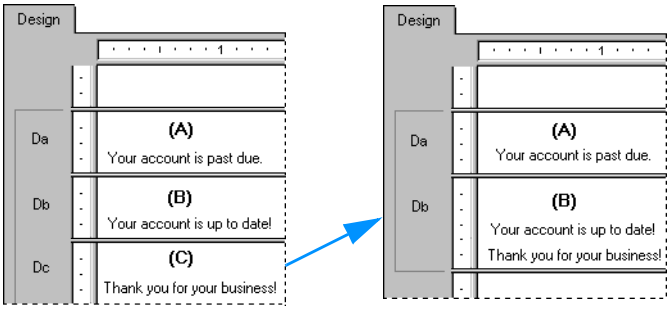
The Section Expert appears with a list of all the sections in the report. When there are more than one of any kind of section, the sections are lettered A, B, C, and so on.

Note: The program enables only those options (free form, new page before, and so on) that apply to the highlighted section.

2 Move the sections so the two sections you want to merge follow each other in the list.

3 Highlight the top section.

4 With Section (B) highlighted, click **Merge** and Section (C) will be merged with Section (B) to form one section.



5 Rearrange the objects as needed.

Splitting and resizing sections

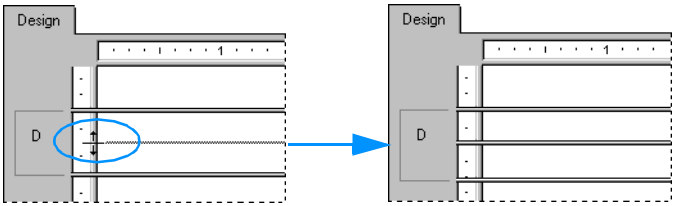
A section can be split into two or more sections and / or resized easily in the Design tab.

Splitting a section

1 Move the pointer over the left boundary of the section you want to split.

2 When the pointer becomes a Section Splitting cursor, click the boundary.

3 When a horizontal line appears, drag it up or down to split the section the way you want it.

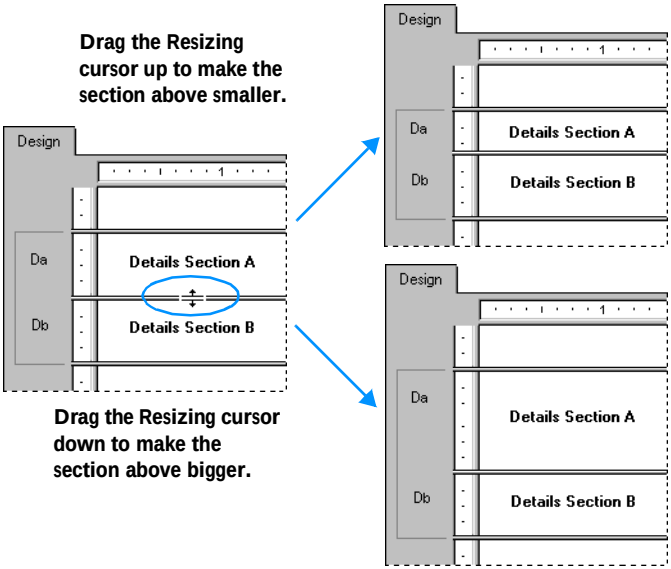


Resizing a section

1. Move the pointer over the bottom boundary of the section you want to resize.
2. When the pointer becomes a Resizing cursor, drag the boundary to make the section bigger or smaller as you wish.

Resizing a section to remove white space

If you have one or more objects in a section and you want to resize the section to remove unnecessary white space, right-click in the shaded area to the left of the section in the Design and Preview tabs and choose Fit Section from the shortcut menu. The program automatically resizes the section, moving the bottom boundary of the section to the baseline of the bottom object in the section.



A section automatically expands vertically in two instances:

- When you place an object and the object is bigger (vertically) than the section you place it in.
- When you expand an object (vertically) so it becomes bigger than the section it resides in.

Note: You cannot resize a section smaller than the combined height of all the objects in the section.

Using multiple sections in reports

Certain reporting tasks are performed most efficiently by creating multiple sections within an area, such as:

- Keeping variable length objects from overwriting each other (see "Keeping variable length objects from overwriting each other" on page 163).
- Eliminating blank lines when fields are empty (see "Eliminating blank lines when fields are empty" on page 164).
- Adding blank lines under specific conditions (see "Adding blank lines conditionally" on page 164).

Once you understand the power of multiple sections, you will discover even more ways to produce the effects you want.

Related topics

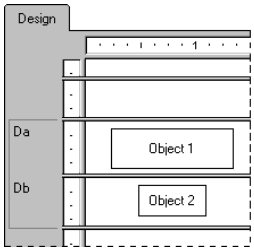
"Working with sections" on page 158

Keeping variable length objects from overwriting each other

When subreports or other variable length objects are placed above other objects in one section of the report while the Can Grow option for the variable length object is toggled on in the Format Editor, that object may overprint objects positioned directly below it unless you have:

- Expanded the section to fit the maximum size of the object.
- Spaced the objects, allowing enough space for the first object to complete printing before the second one begins.

You can eliminate this overprinting problem by creating multiple sections in an area and placing objects below the variable length object in their own section(s).



Now, when the report runs, the section with the variable length object will finish printing before the section below it prints and you will get the results you want. See "Combining two or more unrelated reports" on page 374.

Note: Many report objects can use the Can Grow option and can, therefore, cause overprinting:

- Text fields
- Formula fields
- Memo fields

- BLOB fields
- Subreports
- Cross-Tabs
- OLAP grids

Eliminating blank lines when fields are empty

It is very common to have two address lines in a customer table, one for street address (Address 1), and one that can be used for a suite number or mail stop (Address 2). Address 1 usually contains a value, but Address 2 is often blank. If you create a customer list using this data and stack the fields on top of one another in mailing label format, those customer records with an empty Address 2 field will print with a blank line. You can eliminate this blank line either by using multiple sections, or by suppressing blank lines.

To eliminate blank lines by using multiple sections

1. Use the Section Expert to create two new Details sections so that you have a total of three. See "Working with sections" on page 158.
2. Place the **Address 2** field in the middle section and the other data in the sections above and below it as you want it to appear in the report.
3. In the Section expert, highlight the middle section.
4. On the **Common** tab, select the **Suppress Blank Section** check box.

Now, when the report prints, if the Address 2 section is blank, the program will not print it and you will not get unwanted blank lines in the report.

Note: If the report section you want to suppress contains a blank subreport, use the Suppress Blank Subreport option, found on the Subreport tab of the Format Editor, as well as the Suppress Blank Section option.

Adding blank lines conditionally

Use multiple sections to print a blank line on your report under specific conditions. For example, you may want to insert a blank line after every fifth record in the report.

To add blank lines conditionally

1. Use the Section Expert to create two Details sections. See "Working with sections" on page 158.
2. Place the report detail data in the top section.
3. Leave the second section empty.
4. In the Section expert, highlight the second section.

5 On the **Common** tab, select the **Suppress (No Drill-Down)** check box then click the conditional formatting button to its right.

6 Enter the following formula in the Format Formula Editor:

```
Remainder (RecordNumber,5) <> 0
```

This formula divides each record number by 5. If the division produces a remainder, it suppresses the blank section. But if the division produces no remainder, a zero (which it will for every fifth record printed), the program prints the second section, thus inserting a blank line.

Note: To insert a blank line under different conditions, you can modify your formula appropriately. See "Working with conditional formatting" on page 208.

Form letters

While form letters themselves are not necessarily multi-section reports, they are often used in multi-section reports to generate custom mailings. The topic "Printing conditional messages in form letters" on page 172, explains how to use multiple form letters or multiple versions of the same form letter for custom mailings.

Form letters often use text objects to hold the content of the report. The following sections provide an introduction to text objects and demonstrate how to use them in form letters.

Working with text objects

You will use many of the capabilities of text objects when creating form letters. A brief discussion of text objects should make it easier for you to create the form letter in the next section. Some things to consider are:

- A text object can contain both text and fields; you will use both in this example.
- You can resize text objects; you will be resizing the text object so that it prints as a letter.

Text objects operate in two modes:

- Move/Resize mode
- Edit mode

Move/Resize mode

When the object is in the move/resize mode, it appears as a broken line frame with resizing handles.

In this mode, you can resize the object by dragging any of the resizing handles, or you can move it by placing the cursor inside the object and dragging it to a new location. You can also insert fields in this mode, but you cannot insert text. You put a text object into the move/resize mode by clicking it when it is inactive.

Edit mode

When the object is in the edit mode, it appears as a broken line frame without sizing handles, and an in-place ruler appears at the top of the tab (if you have selected the Show Rulers options in the Options dialog box).

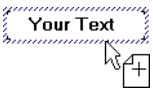
When you first place a text object, the program sets it in the edit mode. You can also put a text object into the edit mode by double-clicking it when it's inactive or in the move / resize mode. Finally, you can put a text object in edit mode by right-clicking it and selecting Edit Text from its shortcut menu.

Each text object contains word processor capabilities, including the ability to change the fonts for individual characters and fields, and automatic word wrap. In the edit mode you can insert text and such non-text objects as database fields and formulas. Whenever the object is in edit mode, it contains an insertion point, a flashing vertical line that indicates the position at which typed text or inserted fields will begin.

The insertion point moves as you type, automatically staying to the right of the last character. It also moves when you insert a field, automatically staying to the right of the field. It moves one character position at a time when you press the space bar. It moves down one line, to the inside left edge of the text object, when you press Enter (this action inserts a carriage return). It moves to the cursor position when you click anywhere within the existing text.

As you work through the tutorials for multiple sections, it is always expected that you will type or place fields at the existing insertion point, unless otherwise indicated.

- To select text within a text object (to delete it, to change a font, and so on), position the cursor over the text and, when the I-beam cursor appears, drag the cursor to highlight the text you want to select.
- To select a field which is inside a text object, position the cursor over the field and, when the I-beam cursor appears, right-click.
- To insert text, type in the desired text and it will appear at the insertion point.



Note: It is critical that you see the Drag and drop cursor before placing the field. If it does not appear, you might place the field so that it overlays the text object instead of being inserted into it. It may appear to be inside the text object, but if you move the object the field will not move with it.

- The insertion point is tied into the Drag and drop cursor. If you have existing text or fields in the text object, the insertion point moves as you move the Drag

and drop cursor, enabling you to pick the exact position where you want to place the field. The program always places the field at the insertion point.

As you move the drop cursor... **the insertion point moves.**

Creating a form letter using a text object

The following section shows you how to create a form letter.

You are going to use a text object to create a form letter. The form letter you create will be tied to a database table so that each letter will be customized with company information from a different record.

If you have difficulty performing any of the steps, please see "Working with text objects" on page 165.

To create a form letter

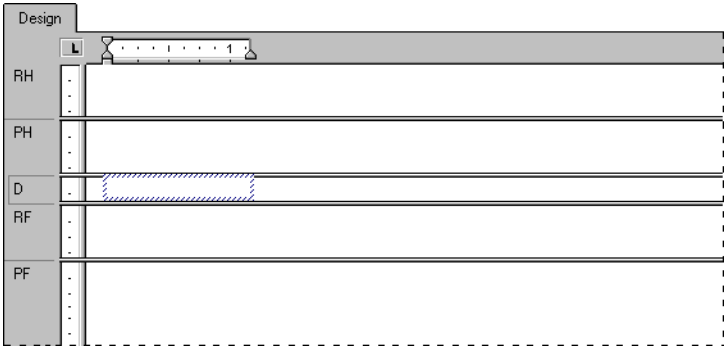
The letter will consist of a date, an inside address, a salutation, a one paragraph letter body, and a closing section.

1. Create a blank report. Use the **Customer** table of **Xtreme.mdb**.

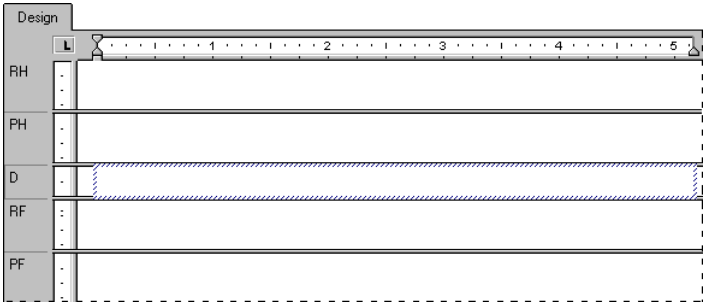
   The Design tab appears.

2. Since you do not want field titles to appear above the fields that you insert into the letter, clear the **Insert Detail Field Headings** check box on the **Layout** tab of the Options dialog box.

3. Insert a text object into the **Details** section of the report.
4. Click the text object frame to put the object in move/resize mode.



5 Drag the resizing handle on the right side of the object to the right edge of the **Design** tab. This will make the object about eight inches wide, the approximate width of a page. You may have to stop resizing, scroll the window, and resize some more to accomplish this.



6 Double-click inside the text object to place it in edit mode. It is now ready for you to begin your work. The insertion point appears at the extreme left, inside the object.

Inserting a date

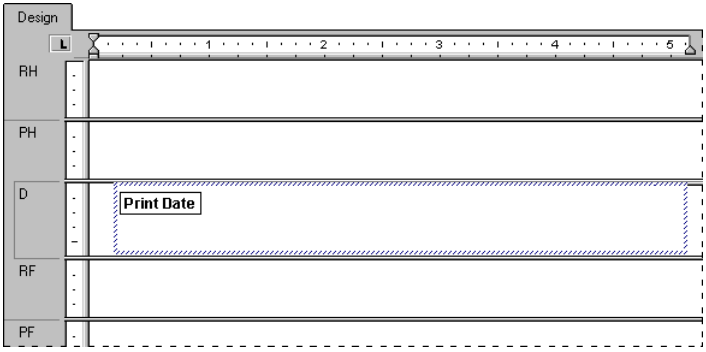
1 To insert a date into the letter, expand **Special Fields** in the Field Explorer dialog box and scroll until you find **Print Date**.

2 Click **Print Date** and drag the placement frame into the text object and place it at the insertion point.

Note: To change the way the date is formatted in the letter, double-click the text object to select it. Then right-click the Print Date field and choose Format {Print Date} from the shortcut menu. Make your modifications on the Date tab of the Format Editor when it appears.

3 Press Enter twice to insert some white space between the date and the inside address and to move the insertion point down within the text object.

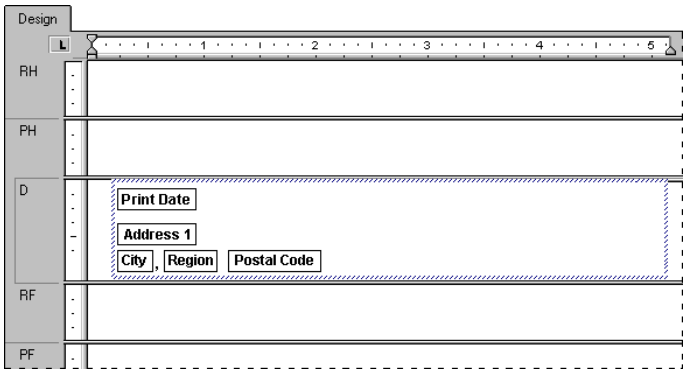
Tip: You might have to resize the Details section and the text object if you have not selected the Can Grow option for the text object in the Format Editor.



Creating an inside address

1. To create the inside address, drag database fields into the text object from the **Customer** table in the Field Explorer dialog box.
2. Drag in the **Address 1** field and place it at the insertion point, and press Enter. The insertion point moves down to the line below.
3. Drag in the **City** field and place it at the insertion point.
4. Type a comma, followed by a space.
5. Drag in the **Region** field and place it at the insertion point.
6. Type in two spaces.
7. Finally, drag in the **Postal Code** field, place it at the insertion point, and press Enter. The insertion point moves down to the line below.
8. Press Enter one more time to move the insertion point down one more line to the position where the salutation should start. This completes the inside address.

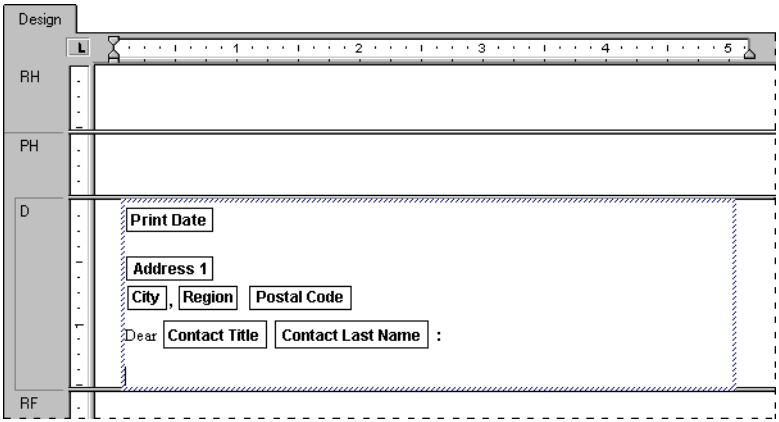
Note: When a field is placed within a text object, it is automatically trimmed on both the left and right sides, so that it contains no extra white space.



Creating a salutation

1. Press Enter four times to move the insertion point down.
2. Type in the word “Dear” and a space (do not include the quotation marks).
3. In the Field Explorer dialog box, highlight the **Contact Title** field from the **Customer** table and drag it into the text object, placing it immediately after the space.
4. Insert a space. The program positions the insertion point immediately after the space.
5. In the Field Explorer dialog box, drag the **Contact Last Name** field into the text object and place it at the insertion point. The insertion point moves to the right of the field.

6 Type a colon ":" at the insertion point (without the quotation marks) and press Enter twice to move the insertion point down two lines.



Creating the letter body

1 Now type "Your company" (do not include the quotation marks) and type a comma followed by a space.

2 Drag the **Customer Name** field into the text object and place it at the insertion point, just after the space.

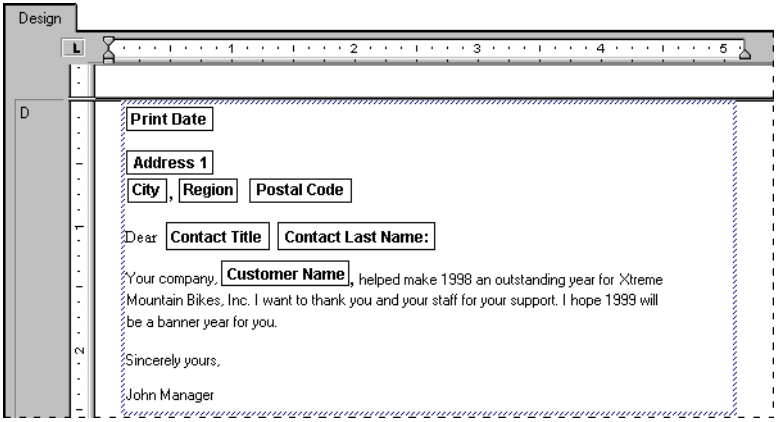
3 Type a comma, followed by a space.

4 Type the following text (do not include the quotation marks): "helped make this year an outstanding year for Xtreme Mountain Bikes, Inc. I want to thank you and your staff for your support. I hope 2000 will be a banner year for you."

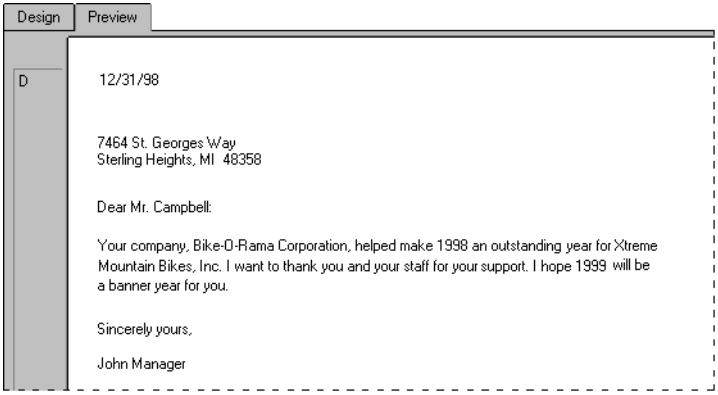
5 Press Enter twice.

6 Type "Sincerely yours" (without the quotation marks), followed by a comma, and then press Enter four times.

7 Finally, to complete the form letter, type your name.
The Design tab should look similar to this:



8 Click **Print Preview** on the Standard toolbar to preview the form letter.
It should look similar to this:

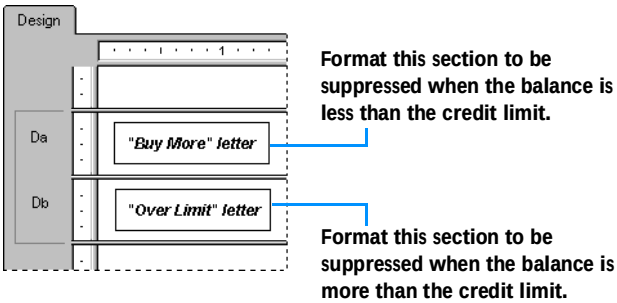


Printing conditional messages in form letters

It is likely that you will want to print conditional messages in form letters. For example, you may want to encourage customers with available credit to buy more and those who are over their credit limit to bring their accounts down below the limit once again. You can create both of these letters within a single report.

To create a conditional message

1. Use the Section Expert to insert a second Details section in the report. See "Working with sections" on page 158.
2. Create two form letters. Place a letter that encourages customers to buy more in the Details A section of the report; place a letter that encourages customers to bring their balance down in the Details B section of the report. See "Creating a form letter using a text object" on page 167.
3. Use the Section Expert to format the Details sections so that each is suppressed under certain conditions. For example:



Now, when a record indicates available credit, the *buy more* letter will print. When the account is over the credit limit, the *over limit* letter will print. And when the customer is right at the credit limit, nothing will print at all.

Related topics

"Working with conditional formatting" on page 208

Formatting 12

Formatting refers to changes in the layout and design of a report, as well as the appearance of text, objects, or entire report sections. This chapter details methods you can use to draw attention to data, change the presentation of dates, numbers, and other values, hide unwanted sections, and perform a variety of other formatting tasks to give a report a professional appearance.

Formatting concepts

This section explains how to format a report. Formatting refers to changes you can make to the layout and design of a report, as well as the appearance of text, objects, or entire report sections.

You can use formatting to do many things, including:

- Dividing sections of a report.
- Calling attention to certain data.
- Changing the presentation of dates, numbers, Boolean values, currency values, and text strings.
- Hiding unwanted sections.
- Giving the report a professional appearance.

The following topics describe the types of formatting you can do with Crystal Reports, giving step-by-step instructions for performing a variety of formatting tasks.

Using a template

A template is an existing report file whose formatting can be added to a new report. At the same time, the formatting of the template report's fields and report objects are applied to the new report. Use templates to give any number of reports a consistent look without having to format each one individually.

For additional information about templates, see Template Considerations in the online help.

Applying a template

When you create a new report in the Standard Report Creation Wizard, you can apply a template as an optional step. You can also apply a template later by using the Template Expert. You can choose from a number of predefined templates, or you can use an existing Crystal report as a template.

To apply a template in the Standard Report Creation Wizard

1. Click **New** on the Standard toolbar.
2. Select **Using the Report Wizard** on the Crystal Reports Gallery dialog box, ensure **Standard** is selected, and click **OK**.
3. Choose data, fields, grouping fields, and so on, until the wizard displays the Template screen.
4. In the **Available Templates** list, click a predefined template name to see an example of it in the Preview area.

   By default, the sample templates shipped with Crystal Reports are installed at `\Program Files\Crystal Decisions\Crystal Reports 9\Templates`.
5. If you want to apply a template based on an existing Crystal report, click **Browse**.

6 In the Open dialog box, select a Crystal report (.rpt) file and click **Open**.

The report is added to the list of Available Templates.

Note: If a template name and preview picture were saved (in the Document Properties dialog box) with the report you selected as a template, you will see this information on the Template screen.

7 Click **Finish**.

Your report appears with its data formatted to match the template you selected.

Note: Formatting is not applied if the report you chose does not meet the requirements for a template. Search the Crystal Reports Online Help for the topic "Template considerations."

To apply a template to an existing report

1 On the **Report** menu, click **Template Expert**.

The Template Expert appears.

As was the case with the Template screen, you can choose from a number of predefined templates, or you can click the Browse button to search for an existing report to use as a template.

2 Choose a template and click **OK**.

Removing an applied template

Perhaps, after applying a template, you don't like the changes made to your report. As long as you haven't exited Crystal Reports since you applied the template, you can remove it from your report.

To remove an applied template

1 On the **Report** menu, click **Template Expert**.

2 Choose **Undo the current template** and click **OK**.

The chosen template's features are removed and your report reverts back to the formatting it had when you first opened it.

Note: To remove a template, you must use this option; the Undo command is not available from the Edit menu.

Reapplying the last template selected

If you want to reapply the last template you selected during a session of Crystal Reports, you can simply select an option on the Template Expert.

To reapply the last template selected

1 On the **Report** menu, click **Template Expert**.

2 Choose **Re-apply the last template** and click **OK**.

Using Template Field Objects

You can use Template Field Objects to create more flexible report templates. These report objects do not refer to existing database fields; you simply put them in your template report and format them as you require. When you apply the template to another report, Crystal Reports displays that report's data with the formatting you specified. Therefore, when you design a template, you don't have to know what data might be in the report you'll eventually apply it to—you use Template Field Objects to take care of the possibilities for you.

Note: Template Field Objects are applied only to result fields: database fields, parameter fields, SQL statements, and formulas. Special Fields are not considered result fields.

To add a Template Field Object to a template report

1. On the **Insert** menu, click **Template Field Object**.

   A placement frame is attached to your cursor.

2. Place the Template Field Object frame on your template report.

   A Template Field Object can be placed in any report section.

3. Right-click the object and click **Format Template Field** from its shortcut menu.

   A list of formatting options appears. You can choose any of the options; when you do, the appropriate tab of the Format Editor appears.

   Tip: You can select multiple Template Field Objects and apply your formatting choices to all of the objects.

4. Specify the formatting for your Template Field Object as you require.

   For information about how to apply formatting, see "Working with absolute formatting" on page 199.

For each Template Field Object you create, a special formula field is created. You can see this formula field in the Formula Workshop. If you want to use sample data in your report to see how your formatting will look, you can reference database fields in these formulas.

To add sample data to a Template Field Object formula

1. In the Field Explorer, expand the **Formula Fields** node, select a Template Field Object, and click **Edit**.

   Tip: In the Formula Fields node of the Field Explorer and the Formula Workshop, Template Field Objects appear as `<TemplateFieldn>`.

2. In the Formula Editor, replace the `Space(10)` section of the argument with a database field of the type you want to see in your sample, save your change, and close the Formula Workshop.

3. Refresh your report's data.

Using the Report Design Environment

Design solutions

There are several things to keep in mind when designing reports that are distributed to different environments. For the best results, consider:

- Section characteristics
- Making an object underlay a following section
- Pre-printed forms
- Multiple columns
- Hiding report sections
- Hiding report objects
- Placing text-based objects
- Placing multi-line, text-based objects
- Importing text-based objects from a file
- Spacing between text-based objects
- Overflow Field Representation
- Selecting multiple objects
- Free-form placement
- Vertical placement
- TrueType fonts
- Page margins
- Default printer
- Printer drivers

Section characteristics

A report consists of several sections, including the Report Header, Page Header, Group Header, Details, Group Footer, Page Footer, and Report Footer.

Each report section is made up of a series of lines. When a text-based object is placed in a section, it is placed on a line in such a way that the text is aligned to the baseline. The line's height is then adjusted by the printer driver so that it is high enough to accommodate the object.

- If you place another text-based object on the same line with a font size larger than that of the first object, the line's height extends to accommodate the second object.
- If you place another text-based object on the same line with a font size even larger than the previous two objects, the line's height extends to accommodate the third object.

A line's height is determined by the text-based object with the largest font size on the line.

As you add text-based objects to a report, either in the same section or other sections, the line height adjusts to accommodate the various fonts. Since this vertical spacing is set up by the printer driver, it is difficult to create reports designed for pre-printed forms when they are printed in various environments.

When designing reports, you should do the following:

- Always print a test page.
- Keep all font sizes the same.
- Be sure to print pre-printed forms on the same machine.

Making an object underlay a following section

Using this example, you can make the Xtreme logo (Xtreme.bmp) underlay multiple sections. This procedure is similar to the one for inserting a company watermark to serve as a report background.

To make an object underlay a following section, first place the object in the section above the section you want it to underlay. Then select the Underlay Following Sections check box in the Section Expert for the section in which the object is placed.

To create a basic report

1. Create a report using the Customer table in **Xtreme.mdb**.

   The Xtreme.mdb file is located in the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory, or the directory in which the program resides.

2. Place {customer.CUSTOMER NAME} and {customer.LAST YEAR'S SALES} side-by-side in the Details section of the report.

3. To remove unnecessary objects from this example, delete the field titles that the program places in the Page Header section of each field.

4. On the **Insert** menu, click **Group** to break the data into regional groups.

5. On the **Common** tab of the Insert Group dialog box, select {customer.REGION}.

6. Click **OK**.

To insert a picture onto the report

1. On the **Insert** menu, click **Picture**.

   Tip: Another way to do this is to click the Insert Picture button on the Insert Tools toolbar.

2. Select the **Xtreme.bmp** file, then place it in the Page Header section, to the right of the fields in the report body.

   The Xtreme.bmp file is located in the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory, or the directory in which the program resides.

Note: In this example, the picture is placed to the right of the fields to avoid underlaying the text. When you are working with a watermark, a subdued picture designed to be nearly invisible, place it directly above the text.

3 On the Standard toolbar, click **Print Preview**.

The picture prints in the Page Header section of each report page.

To make the picture underlay the following section

1 On the **Report** menu, click **Section Expert**.

The Section Expert dialog box appears.

Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.

2 In the **Sections** list, click **Page Header**, then select the **Underlay Following Sections** check box.

3 Click **OK** to preview the report again.

The picture now prints in both the first Group Header and the following few Details sections, next to (instead of above) the text in the body of the report.

Note: Using the technique of placing a picture to the right of the body of the report, you can set up a chart or an employee picture to print beside the details pertaining to that chart or employee.

4 Once you are finished previewing the report, return to the **Design** tab.

5 Resize the picture vertically to make it two or three times larger, then preview the report again.

The bitmap file now underlays more sections.

The area in which the picture underlays depends on the following conditions:

- The size of the picture.
- The section in which the picture was originally placed.
- The position of the picture in the section.

By modifying size and placement of an object, you can create a variety of visual effects, using the underlay feature.

Pre-printed forms

If you print on pre-printed forms, you will be able to:

- Scan a form.
- Place it in the report as a bitmap.
- Use the underlay feature to line up the bitmap and report, as well as move objects anywhere you want them to appear.
- Eliminate the need to print the forms separately by printing the report and the form as a single unit.

Multiple columns

Instead of having your data print straight down the page, you can set up multiple columns and have the data flow from column to column.

To create a multiple-column report

1. Open then report you want to format with multiple columns.
2. On the **Report** menu, click **Section Expert**.

   Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.
3. In the Section Expert, highlight **Details**, and then select **Format with Multiple Columns**.

   A Layout tab is added to the Section Expert.
4. Click the **Layout** tab and set the **Width** you want your column to be.

   Keep in mind that the width of your paper when deciding your column width. For example, if you have three fields in your Details section, and they take up four inches of space, limit the width of the column to under four and a half inches so that all the field information can be seen.
5. Set the **Horizontal** and/or **Vertical** gap you want to maintain between each record in your column.
6. In the Printing Direction area, choose a direction.
7. If the report you're formatting contains grouping, select **Format Groups with multiple column**.
8. Click **OK**.

When you preview the report, you'll see that the field headers appear only for the first column. To have field headers for the second column, insert a text object.

Hiding report sections

Crystal Reports has three properties you can set in the Section Expert to hide report sections.

Hide (Drill-Down OK)

The Hide property hides a section whenever you run the report. For example, in a summary report, the Hide property can be used to display only the summaries, but not the details behind the summaries. When the Hide property is applied to a section, it becomes visible when the Drill-down cursor is used to drill down on the section contents. This property is absolute; it cannot be conditionally applied using a formula.

Suppress (No Drill-Down)

The Suppress property also hides a section when you run the report. Unlike the Hide property, however, you cannot apply the Suppress property, then drill down to reveal the section contents. This property can be applied absolutely, or conditionally using a formula. This is useful for writing form letters. For example, in a form letter, you might create two Details sections: one to suppress when sales are equal to or over $X and one to suppress when sales are under $X.

Suppress Blank Section

The Suppress Blank Section property hides a section whenever there is nothing in it. If something is placed within the section and it produces a value in your report, then it becomes visible.

Hiding report objects

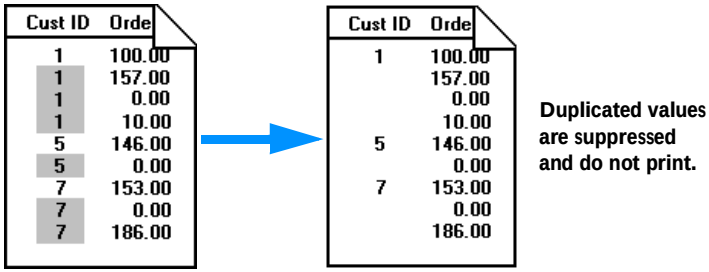
Crystal Reports has three formatting options in the Format Editor for hiding individual objects.

Suppress If Duplicated (Common tab)

The Suppress If Duplicated property prevents a field value from printing if it is identical to a duplicate of the value that comes immediately before it in an iteration of the same section.

The value does not print, but the space in which it would have printed remains.

Note: This option does not work for text fields that contain embedded fields.

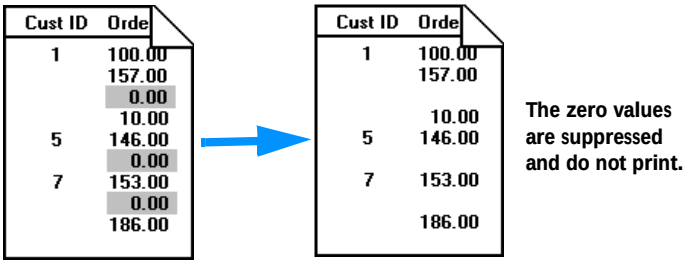


Suppress If Zero (Number tab)

Tip: To find this option, click the Number tab of the Format Editor, then click the Customize button.

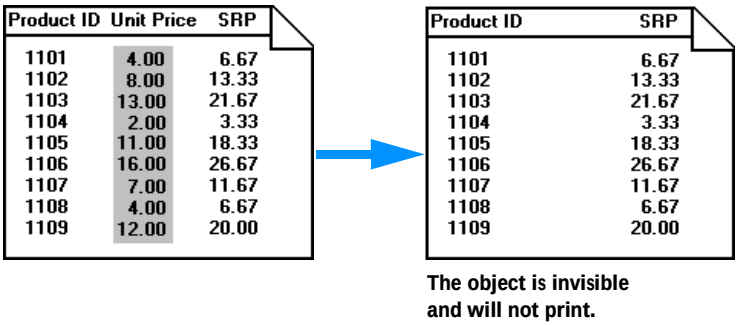
The Suppress If Zero property prevents a value from printing if it is a zero value. The value does not print, but the space in which it would have printed remains. To remove the blank space, select the Suppress Blank Section check box in the Section Expert.

Note: This will only work if there are no other objects in the section.



Suppress (Common tab)

The Suppress property hides an object when you run the report. For example, it is common to apply this property to formulas that are needed to do some report calculations, but that you do not want to print when you run the report. When you select this property, the selected object does not print.



Note: You can click the Conditional Formula button for any of these properties and create a formula that will make the setting conditional on some event. See "Working with conditional formatting" on page 208.

To set these properties, select the object, then click Format from the Expert Tools toolbar to open the Format Editor dialog box. When the Format Editor appears, set the properties.

Placing text-based objects

When a text-based object is placed on a report, the object is represented by an object frame. The height of the object frame is based on the height of the font. The width, however, is determined differently, depending on the object you are working on.

- For database fields that are not memo fields, the width is initially determined by the width of the field as defined in the database, and by the average character width as provided by the selected font and font size.

  For example, you have a database field called {customer.LAST NAME} and your database defines this field as a text field with a length of 35 characters. When you place this field on your report, the width of the boundary is 35 times the average character width of the font and font size that the database field is formatted to. Remember that this is the initial default boundary width. The width can always be resized to increase or decrease as you see fit.
- For text-based objects, the default width is approximately 19 average character widths wide. Text objects are different from database fields in that their width automatically expands as you enter in text and/or database fields into the object. As with all other text-based objects, the width can be resized by the user.
- For different number fields (double, single, integer, long integer, and byte) the default widths are all different. As with all objects, the width can be resized by the user.

Preventing the truncation of text inside an object

Whether the default widths are accepted or the text-based objects are resized, a problem could arise if the text inside the object prints right to the edge of the object frame. While the report may look fine on the machine it was designed on, when the report is printed using another printer driver that measures the font wider, the length of the text grows, but the object frame remains fixed. The resulting text is cut-off or truncated.

To prevent the truncation of text inside an object

1. Right-click the text object you want to format to bring up the shortcut menu.

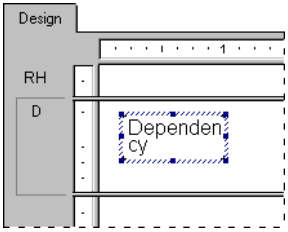
   Tip: Another way to do this is to click the Format button on the Expert Tools toolbar.
2. On the shortcut menu, click **Format Text**.

   The Format Editor dialog box appears.
3. On the **Common** tab, select the **Can Grow** check box.
4. Click **OK** to save your changes.

   The object is then formatted to print on multiple lines. If the text prints wider than the object, the text wraps onto additional lines.

Preventing breaks in non-spacing text inside an object

For text strings that do not contain spaces, such as single words, the text string is broken at the edge of the object frame before the line starts to wrap.



To prevent the breaks in non-spacing text inside an object

1. Select the object you want to format.
2. Expand the object frame to make it wider than the widest block of text inside the frame.

There are many times when the actual text in a database field is far less than the maximum amount the field can contain. For example, a {table.LAST NAME} field is designed with a field size of 80 and the longest name in the database is 28 characters. In this case, when you first place the field in your report, the field is 80 times the average character width. Reduce the width of the field, but include enough space to account for growth.

While each of these options offers an effective solution when dealing with a single text-based object in a section, there are still design considerations to take into account when placing more than one object in a section. When sizing an object, consider its placement with regard to other objects in the section.

Avoid designing reports where the space between each object is very tight. Leave room for growth by expanding the width of the object by approximately 5 per cent. Or, if this is not possible, consider reducing the size of the font or placing each object in its own subsection.

Suppressing blank lines in embedded fields

Because you can embed fields in text objects, you might encounter cases where an empty field causes a blank line in a text object. You can suppress blank instances of such embedded fields.

Note: The suppression of embedded field blank lines is designed to remove blank lines within a text object if the text object contains an entirely blank field alone on a line followed by a carriage return.

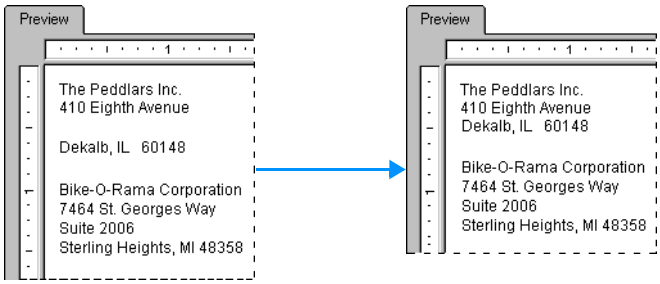
To suppress blank lines in embedded fields

1. Open your report in the **Design** tab, and click the desired text object—that is, the text object that causes blank lines to show for some records.

   Tip: To ensure that you have clicked a text object, look for the word Text in the status bar at the bottom left corner of the screen.

2. Right-click the text object and, on the shortcut menu, click **Format Text**.
3. Select the **Suppress Embedded Field Blank Lines** option in the Format Editor, and then click **OK**.

Now, when the report prints, unwanted blank lines will no longer appear in place of empty embedded fields. You can confirm your changes in the Preview tab.



Before suppressing blank lines, the Address2 field prints a blank line if it is empty.

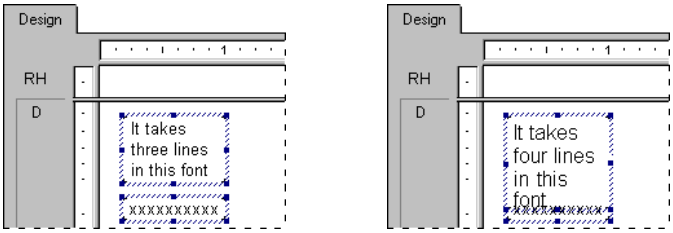
After suppressing blank lines, the Address2 field does not print if it is empty.

Placing multi-line, text-based objects

While text-based objects that are formatted to print on multiple lines follow the same design rules as other objects, they have one additional characteristic to be considered. If the printer driver expands or contracts the spacing of the text, word wrapping may differ, changing the number of lines necessary to print the object in order to accommodate growth or shrinkage.

When placing multi-line text-based objects, you could encounter problems if other objects in the same section are placed directly below them.

Unlike single-line text-based objects, expanding the object frame of a multi-line text-based object to accommodate growth is not a viable option. When you do this, the line width increases according to the expanded boundaries.



So, when possible, place multi-line text-based objects at the bottom of a section. If they require more lines to print, the section expands downward to accommodate the growth, and they do not endanger other objects.

Importing text-based objects from a file

Using Crystal Reports, you can import a formatted, text-based object from an existing file onto your report.

To import text-based objects from a file

1. Double-click the text-based object you want to format to put it in edit mode, then right-click it to bring up the shortcut menu.
2. On the shortcut menu, click **Insert from file**.
3. In the Open dialog box that appears, select the file in which your text-based object is stored, then click **Open**.

   The object is imported from the file into the text object on your report.

Spacing between text-based objects

Use the grid and guidelines options to help evenly align text-based objects.

You can select the Snap To Grid option, set the grid to a maximum of one inch, and make the grid visible or invisible on the Design tab, Preview tab, or both. For more information on working with grids, see "Using the grid" on page 186.

You can also work without a grid, placing objects wherever you want them on a report. You may want to work in a free-form environment while retaining the ability to align objects, or to move or resize them as a group. You can do this using guidelines. See "Designing with guidelines" on page 187.

Using the grid

The grid is a series of row and column coordinates. When the grid is selected and the Snap To Grid option is selected on the Layout tab of the Options dialog box, Crystal Reports enables you to place text-based objects only at these coordinates, not between them. You can then space data on your report and align objects as needed. If you attempt to place an object between grid coordinates, the object "snaps" to the grid; that is, the object automatically moves to the nearest set of row and column coordinates.

Each report contains a design grid. You can select the grid on or off, as well as set it to different sizes when required. By default, the grid is not selected. See "Selecting the grid" on page 187.

Once set, the grid remains the same size for all sections. It is measured from the upper-left corner of each section and continues down and to the right, until the end of the section. A new grid of the same size then starts from the upper-left corner of the next section, and so on, through the end of the report.

If you select the Snap To Grid option, the following conditions occur:

- The upper-left corner of all newly placed, text-based and OLE objects snap to a grid point.
- Objects placed on a report before the Snap To Grid option is selected do not snap to the nearest grid point. They remain in the same place.
- If you resize an object, the side or sides that you are resizing snap to the nearest grid point.

Selecting the grid

The Design and Preview tabs have an underlying grid structure that you can activate on the Layout tab in the Options dialog box.

To select the grid

1. On the **File** menu, click **Options**.

   The Options dialog box appears.

2. On the **Layout** tab, in the Grid options area, you can activate the snap to grid feature, show the underlying grid structure on the Design or Preview tab, or both, select free-form placement, and specify grid size.

3. Click **OK** to save your changes.

Tip: Another way to do this is to select the Grid commands from the View menu. You can also right-click an empty space on the Design or Preview tabs and select the command from the shortcut menu.

Designing with guidelines

Crystal Reports provides guidelines to help you align and size report objects with accuracy. Guidelines are non-printing lines that you can place anywhere on the Design and Preview tabs to aid in alignment. They have a snap property that automatically snaps objects to them.

Viewing guidelines

You can view guidelines on the Design and Preview tabs by selecting the view options in the Options dialog box.

To view guidelines

1. On the **File** menu, click **Options**.

   The Options dialog box appears.

2. On the **Layout** tab, in the Design View area, select the **Guidelines** check box and / or the **Guidelines** check box in the Preview area.

3. Click **OK** to save your changes.

Tip: Another way to do this is to select a Guidelines command from the View menu.

Inserting guidelines

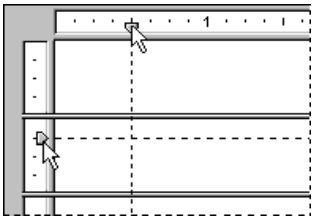
Although you can and should insert manual guidelines whenever necessary, Crystal Reports will automatically insert guidelines in certain situations:

- Whenever you add a field or formula field to a report, the program creates a guideline at the left edge of the field frame and snaps the field and field title to it.
- When a field is summarized, the program snaps the summary to the same guideline to ensure proper alignment.
- When you right-click the shaded areas to the left of a section, then select the Arrange Lines option from the shortcut menu, the program creates one or more horizontal guidelines in the section and snaps the fields to them.

To insert, move, and remove guidelines manually

1. In the Design or Preview tab, click the ruler at the top to activate a vertical guideline; click the ruler on the left to activate a horizontal guideline.

   Notice that each guideline is attached to an arrowhead on its originating ruler.



   Note: If guidelines don't appear, ensure that the appropriate Guideline option is selected on the View menu. On the Preview tab, you must select an object to see a guideline.

2. To position a guideline, drag its arrowhead along the ruler to the desired location.

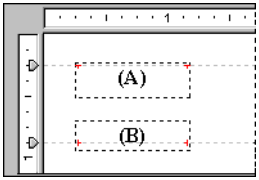
3. To delete a guideline, drag its arrowhead away from the ruler.

Note: If you select the Snap To Grid option, you can only insert or move guidelines in grid increments.

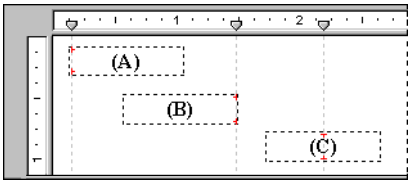
Snapping objects to guidelines

To snap an object to a guideline, drag the report object onto the guideline until the object's edge sits atop the guideline. Snapping enables you not only to align report objects accurately, but also to re-position and re-size multiple objects together. Once several objects are snapped to a guideline, you can move all the objects by moving the guideline.

You can snap either the top or the bottom of an object to a horizontal guideline.



You can snap an object's left side, right side, or vertical midline (the invisible line that bisects an object vertically) to a vertical guideline.



To snap objects to a guideline

1. Insert a guideline by clicking one of the rulers.
2. Drag the report object onto the guideline, so that one of the object's edges is on the guideline.

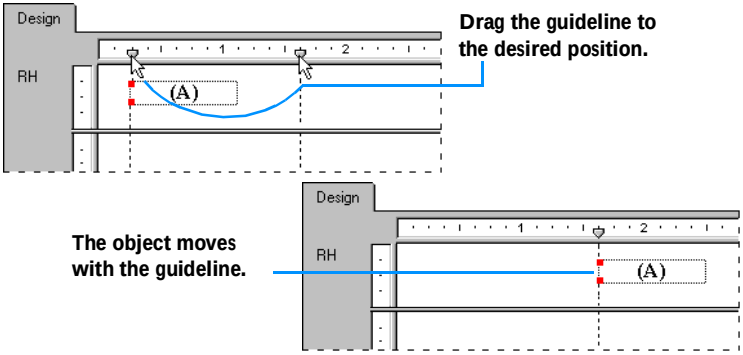
The snap property of guidelines works differently for text-based objects than for other objects, such as OLE objects. When a single-line text-based object snaps to a guideline, it is the baseline of the text, not the object frame, that snaps to the guideline. When a multi-line text-based object snaps to a guideline, either the baseline of the text or the object frame can be snapped to the guideline.

You can tell if a text-based object is snapped to a horizontal guideline by looking for the special indicators positioned on either side of the object directly at the baseline (as circled in the image below) in Design view. If the object is snapped to a vertical guideline, the special indicators appear along the side of the object.

To place several text objects of different font sizes on one line with their baselines lined up, snap each object's baseline to the same horizontal guideline.

Positioning objects using guidelines

Once you've snapped one or more objects to a guideline, you can move all the objects at once by moving the guideline. To move the guideline, drag its arrowhead along the ruler.



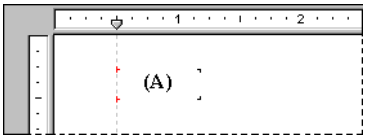
Note: When a guideline is moved, any object that is snapped to it is moved as well. But, if you move an object that is snapped to a guideline, the guideline does not move.

Resizing objects using guidelines

Use two guidelines to resize two or more objects that are either the same size (height or width) or different sizes. The process for resizing with two guidelines is the same in either case.

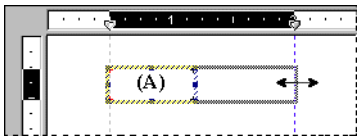
To resize objects using guidelines

1. Create a guideline.
2. Snap one side of the object to that guideline.



3. Create a second guideline to the right of the object.

   Note: The guideline should not be touching the object.
4. Click the object to activate the sizing handles.
5. Drag the resizing handle over to the second guideline so that the object snaps to the guideline.



6 For each additional object you want to snap to both guidelines, repeat Steps 2 through 5.

7 If the objects are not the desired size, drag one or both of the guidelines until the objects are the correct size.

Indenting lines

Using Crystal Reports, you can control line indentation for memo fields, string fields, and text-based objects. For objects, you have the option of indenting lines for a particular paragraph by positioning the cursor at the start of that paragraph. Or, if you select an object in its entirety, you can apply the same indenting specifications to all the paragraphs within that object.

Keep in mind that any line following a carriage return will be considered the first line of a new paragraph.

To indent lines

1 Right-click the field or object you want to format to bring up the shortcut menu.

2 On the shortcut menu, click **Format text**.
The Format Editor dialog box appears

3 Click the **Paragraph** tab.

4 In the Indentations area, you can indent the first line of the paragraph; indent every paragraph line from the left margin; and indent every paragraph line from the right margin.

Note: Only indentation values within the range of the field or object width are accepted.

5 Click **OK** to save your changes.

Overflow Field Representation

Crystal Reports uses Overflow Field Representation to assist users when working with numeric or currency values in report cells. Normally, if a numeric or currency value is larger than the field containing it, that value is truncated, or “clipped.” For example, values like 100,000,000 might appear on the report as 1,000, or as 000 (depending on the properties you have set). This could potentially cause confusion when the report is read.

When the Allow Field Clipping formatting option is cleared, numeric/currency field values that exceed the field size will be represented by number signs (######) in the Preview tab, letting you know immediately when the field is too small.

To allow for overflow field representation

1 Right-click the currency field or number field you want to format to bring up the shortcut menu.

2 On the shortcut menu, click **Format Field**.

The Format Editor dialog box appears with the Number tab open.

3 Click the **Customize** button.

The Custom Style dialog box appears with the Number tab open.

4 To allow overflow field representation, clear the **Allow Field Clipping** check box.

Note: You also have the option to click the Conditional Formatting button to enter a formula in the Format Editor. In the Format Formula Workshop, you can specify that field clipping will be disabled only when certain conditions are met.

5 Click **OK** to save your changes.

To view the results, refresh the report. If you disabled field clipping, any numeric/currency field values that are larger than the field objects containing them will be represented by number signs (######).

Selecting multiple objects

You can select multiple objects, including text, field, chart, map, bitmap, OLAP grid, Cross-Tab and OLE objects, to format them together.

Once you have selected multiple objects, you can move, align, size, cut, and copy and paste them as a group. You can also change their font, color, and paragraph style.

Objects are moved, aligned, and sized based on a "main" object, which is usually the last object you select. You can change the main object to another by right-clicking the desired object.

To select multiple objects

1 Click one object, and Ctrl+click the other objects you want to select.

Note: You can also lasso a number of objects by left-clicking and dragging your cursor over an area.

2 Right-click the main object.

3 On the short-cut menu, select the appropriate formatting option.

Free-form placement

You may also want to use Crystal Reports without a grid, in a free-form environment similar to that of a drawing program. Free-form means that you can place objects anywhere you want them to appear on your report. Your only restriction is that the program does not allow you to place chart and Cross-Tab objects on the Page Header, Page Footer, or Details section.

To work in a free-form environment, select the Free-Form Placement option on the Layout tab in the Options dialog box.

Free-form placement on

In order to create dynamic reports and reduce printer driver dependency as much as possible, it is recommended that all sections of the report be formatted with the Free-Form Placement option turned on. This is especially true if your report includes OLE objects such as charts, boxes, lines, and bitmapped images (e.g. company logos).

When a section is formatted with free-form placement, *all* objects can be placed *anywhere* on that section. Crystal Reports places objects within a section based on their absolute coordinates. These coordinates determine the vertical placement of objects on your report. What this means is that you control the vertical placement of single-line, text-based objects rather than the printer driver. In doing so, you can better protect your reports from printing inconsistencies caused by using different printer drivers.

However, while the printer driver no longer controls the vertical spacing of text-based objects within the sections, it still determines horizontal spacing of text within the text-based objects, as well as the inter-line spacing of multi-line, text-based objects. So, while free-form placement gives you better control, you must still take into account these considerations when designing your reports. See "Placing multi-line, text-based objects" on page 185.

Free-form placement off

If a section has the Free-Form Placement option turned off, the program no longer references the object's absolute coordinates to determine where it prints, except under the following conditions.

- The absolute *x* coordinate is still referenced to determine where each object begins printing horizontally (left and right placement).
- The *y* coordinate is still referenced for vertical placement of the object, but the coordinates may be adjusted by the Crystal Reports Designer when the printer driver changes.

So, if the report is printed using a printer driver that measures inter-line spacing *greater* than the original printer driver, the *y* coordinate increases and the text-based object prints farther down the page. With the Free-Form Placement option turned off, the user no longer controls the vertical placement of text-based objects; the printer driver determines that instead.

However, the placement of OLE objects such as graphics, boxes, and lines is *not* controlled by the printer driver. Therefore, when placing and printing, the Crystal Reports Designer always references the absolute coordinates of these objects. Due to the various placement methods, problems can arise when both text-based objects and OLE objects are placed on a report.

For example, a box (OLE object) is placed around a database string field (text-based object). Everything looks great and everything is aligned as it should be. However, if the report is printed using a printer with an especially high measurement for inter-line spacing, then the following conditions occur.

- The placement of the box does not change relative to the section in which it is placed (the *x* and *y* coordinates do not change).
- The vertical placement of the text-based object changes because the *y* coordinate is adjusted upward.

Note: The value is a measurement from the upper-left corner of the section; so, the greater the value, the farther down the page the object prints.

If you are distributing reports that have sections formatted with the Free-Form Placement option turned off, and you want the objects in those sections to be surrounded by a border or formatted with lines, it is better to modify the objects' border properties than to insert lines and boxes. This way, the borders always stay with the objects. See "Adding borders, color, and shading to a field" on page 200.

Remember, *any* section can be formatted with the Free-Form Placement option turned on or off. While it is better to turn the option on in some cases and not in others, it is highly recommended that every section of your report be formatted in the *same* manner.

Free-form placement using guidelines

You may want to work in a free-form environment while retaining the ability to align objects, or to move or resize them as a group. You can do this using guidelines.

Guidelines are lines that extend vertically or horizontally from the Design and Preview tabs. Guidelines have a snap property. When you move an object within a guideline's magnetic range, the object snaps, or attaches itself, to the guideline.

- Once an object is snapped to a guideline, if the guideline is moved, the object also moves.
- If you have several objects snapped to a guideline, they all move when the guideline is moved.
- If you have several objects snapped to a guideline on two sides (right and left, or top and bottom) and one of the guidelines is moved, all of the objects resize similarly.

Using guidelines in a free form environment provides flexibility with control. See "Designing with guidelines" on page 187.

Vertical placement

On the Common tab of the Format Editor, you can use the text rotation options to vertically align the fields and text-based objects on your report.

When you select a text rotation of 90 degrees, the text shifts 90 degrees in a counter-clockwise direction.

When you select a text rotation of 270 degrees, the text shifts 270 degrees in a counter-clockwise direction.

Note:

- If text rotation is left at 0 degrees, your report is horizontally formatted, left to right.
- For text rotation of text-based objects, the Can Grow option that prevents the truncation of text inside an object is automatically cleared. For more information on the Can Grow option, see "Preventing the truncation of text inside an object" on page 183.
- Vertically formatted text that spans over the edge of the page cannot be displayed as part of your report.

Inserting character and line spacing

With Crystal Reports, you can specify the amount of spacing between characters or lines for memo fields, string fields, and text-based objects.

To insert character and line spacing

1. Right-click the field or object you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Field**.

   The Format Editor dialog box appears.
3. Click the **Font** tab.

   You will use this tab to set up the character spacing values.
4. In the Spacing area, in the **Character spacing exactly** field, specify the value *n* that each character occupies.

   The value *n* is defined as the distance in number of points measured from the start of one character to the start of the next. When you change the character spacing, you change only the spacing between adjacent characters, not the font size of the characters.

   For example, if you specify a 14-point font with a character spacing of 14 points, each character will remain as a 14-point font size, occupying a space that is 14 points wide.
5. Click the **Paragraph** tab.

   You will use this tab to set up the line spacing values.
6. In the Spacing area, in the **Line spacing** field, specify the line spacing as a multiple of the font size you are using, or as an exact number of points.
7. Click **OK** to save your changes.

Setting fractional font sizes

On the Font tab of the Format Editor, you can select a fractional font size for database fields and text-based objects on your report.

To set fractional font sizes

1. Right-click the field or object you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Field**.
   The Format Editor dialog box appears.
3. Click the **Font** tab.
4. In the **Size** list, type the desired fractional font size for the field or object.

   Note: The number you type must be between 1 and 1638. Crystal Reports will automatically round all fractional entries to the nearest 0.5. Consequently, in your report, you can use the fractional font sizes 1.5, 2.5, 3.5, and so on, up to 1637.5.
5. Click **OK** to save your changes.

Note: When setting fractional font sizes for individual database fields and text-based objects that you've already placed on your report, you must make your changes manually—that is, by following these procedures. (This is because the existing font settings of objects in your report will override your default Options.) However, you can use the Fonts tab of the Options dialog box to adjust your default font settings: these default Options will affect the new reports that you create, along with any new objects that you add to an existing report.

TrueType fonts

Designing your report with printer-specific fonts can lead to problems when using different printers. The fonts may not be supported by the other printers, or if they are supported, the fonts may not be installed on the printers.

During the printing process, if you encounter printer-specific fonts that are unrecognizable to the printer driver, Crystal Reports substitutes the fonts, creating inconsistent results. To avoid this situation, only common TrueType fonts should be used when designing reports.

Page margins

Setting specific margins

Crystal Reports gives you the option of setting margins to meet your specifications.

To set specific margins

1. On the **File** menu, click **Page Setup**.
   The Page Setup dialog box appears.
2. Change the default page margins to fit your needs.
3. Click **OK** to save your changes.

Note: All margins are calculated from the paper edge. Thus, a left margin of .25 inches causes the printing to start exactly one quarter inch in from the edge of the paper.

Using default margins

If you decide to design your report using the default margins, the following problems may occur.

- When printing a report in another environment where the printer's default margins are *greater* than its setting, the report objects on the right side of the report print off the page.
- When printing a report in another environment where the printer's default margins are *smaller* (allowing a larger printing area), the entire report moves to the left side of the page.

It is recommended that you always set your own margins. Even if the margins you want to use are the same as the default margins, be sure that the Use Default Margins option in the Page Setup dialog box is not selected, and you set your margins manually using Page Setup.

Note: You can select the Adjust Automatically check box if you want Crystal Reports to adjust the report's margins automatically when the paper size changes. This option preserves the report's printable area by enlarging or reducing the left/right and top/bottom margins by the same factor.

Default printer

In general, it is a good idea not to choose a specific printer. Even though the printer may be identical to the default printer, how the printer is recognized can still vary for different operating systems.

For example, an HP Laser III printer is being installed on three different operating systems.

- With Microsoft Windows 98, the printer name can be changed so that HP Laser III is Front Reception Printer, but the printer driver will be listed as HPPCL5MS.DRV.
- With Microsoft Windows NT, the printer name is also referenced and can be changed by the user, but the printer driver is always WINSPOOL.

If you select a specific printer, Crystal Reports looks for that printer by name. If the printer you selected cannot be found, the default printer is chosen, resulting in the possibility of printing inconsistencies.

When selecting a specific printer, such as a label printer or a printer dedicated to printing invoices, the printer name must be the same as the name of the printer the report was designed on. Be aware that anyone printing the report must use that same printer or they could encounter problems.

If your report is part of an application that you are distributing, you can provide a Select Printer dialog box. Using this dialog box, users of your report can choose the correct printer or rename the printer accordingly.

Setting page orientation and paper size

You can print your reports using either portrait or landscape orientation, and in a variety of paper sizes. You can specify these options using the Printer Setup command from the File menu.

To set page orientation and paper size

1. On the **File** menu, click **Printer Setup**.
   The Print Setup dialog box appears.
2. In the Printer area, in the **Name** list, select the printer you want to use, if it is not already the default printer.
3. In the Paper area, in the **Size** and **Source** lists, select the desired paper.
   The paper size options are directly related to the printer you select. For example, the HP LaserJet driver (PCL) offers a choice of letter, legal, executive, or A4 paper sizes, whereas the PostScript printer driver lets you choose from letter, legal, note, A4, B5, letter small, and A4 small paper sizes.
4. In the Orientation area, click **Portrait** or **Landscape**.
5. Click **OK** to save your changes.

Printer drivers

Updating printer drivers

In order to maintain performance, Crystal Reports queries the printer driver for each of the font elements (font metrics), such as average character height, character width, height of the ascenders and descenders, internal leading, and so on. A problem may develop if using an older printer driver that does not return the font metrics accurately. If you are experiencing problems when printing (missing fields, incorrect formatting, and so on), it is recommended that you obtain and install the most recently updated drivers for your printer. In many cases, the newer printer drivers provide accurate font metrics and any printing issues are quickly resolved.

Inconsistencies due to printer drivers

When printing, inconsistencies may occur if different printer drivers are used to create and print your reports. These inconsistencies are a result of the various methods that individual printer drivers use to measure text metrics such as font size. When printed, text-based objects may be misaligned, truncated, or overprint each other. Examples of text-based objects include string or character fields, text objects, memo fields, numeric fields, and formula fields.

Problems such as these may arise when you have:

- Two identical printers, but each one is using a different printer driver.
- Two different printers using the same printer driver.

- Two different printers using different printer drivers.
- One printer driver that uses the TrueType font and a second printer driver that maps TrueType fonts to PostScript fonts.
- Two identical printers using the same printer driver, but each one is printing from a different version of Microsoft Windows.
- Two identical printers using the same printer driver, but the printer drivers are different versions.
- Two identical printers, two identical printer drivers, and two identical operating systems, but the resolution of the video drivers is different.

Therefore, while a document using one printer driver may require six full lines to display a block of text:

- Using a second printer driver that measures fonts narrower could result in the same block of text requiring less than six full lines.
- Using a third printer driver that measures fonts wider could require more than six full lines.

For the most part, this situation cannot be avoided. The goal of the report distributor is to design reports that accommodate printer driver dependency and still print consistently using different printer drivers. To do this, Crystal Reports provides several design solutions. If taken into account when creating your report, these solutions can ensure proper printing and distribution for your report in almost any environment.

Formatting properties

You can set formatting properties using the Format Editor for objects and the Section Expert for report sections. In most cases, you can set one of two types of properties:

- Absolute (always apply the property).
- Conditional (apply the property only if certain criteria are met).

Working with absolute formatting

Absolute formatting is formatting that applies under any condition. This type of formatting property always follows a *select, then apply* procedure. For example, you *select* what it is that you want to format (object or section), *then* you *apply* the formatting to the selection using property settings.

You can use the following dialog boxes to format your reports:

- Format Editor to format field values.
- Section Expert to format entire sections.
- Highlighting Expert to conditionally format all types of fields.

Each of these dialog boxes contains a number of different formatting properties, as well as the tools for turning the properties on or off and specifying attributes.

Adding borders, color, and shading to a field

Crystal Reports allows you to add borders, color, and shading to fields on your report in order to emphasize important data and create professional-looking reports.

To add borders, color, and shading to a field

1. Right-click the field you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Editor**.
   The Format Editor dialog box appears.
3. Click the **Border** tab.
4. Select the line style, color, and background color of the field.
5. Click **OK** to save your changes.

Making a report or object read-only

You can make a report or object read-only so it can't be formatted. When you set this option, choices in the Format Editor become inactive. The formatting options that are usually available on the toolbars or shortcut menus are also suppressed for the report or object.

Note: This feature is for your convenience in protecting report formatting; it is not intended to be used as report security.

To make a report read-only

1. On the **File** menu, select **Report Options**.
2. Select **Read-only**.

To make an object read-only

1. Right-click the object you want to make read-only.
2. On the shortcut menu, click **Format Editor**.
   The Format Editor dialog box appears.
3. On the **Common** tab, select **Read-only**.

Note: You can also do this by clicking the Lock Format button on the Formatting toolbar.

Locking an object's size and position

You can lock the position of the selected report object so it can't be moved. When you set this option, you cannot drag the object in the Report Designer and the Size and Position command becomes inactive.

To lock an object's size and position

1 Select the object whose size and position you want to lock.

2 On the Formatting toolbar, click **Lock Size/Position**.

Changing your default field formats

Crystal Reports allows you to display database fields in almost any format on your report. This section describes how to use the Options command to control the default format settings that Crystal Reports uses when you add a field to any report. In the Options dialog box, you can set the default formats for database fields of the following type: String, Number, Currency, Date, Time, Date and Time, and Boolean.

Note: When you change default field formats, your new settings affect only the objects that you subsequently add to a report. To format fields that you've already added to a report, you must right-click the field in the report and select Format Field from the shortcut menu.

To specify default formats for fields

1 On the **File** menu, click **Options**.

2 In the Options dialog box, click the **Fields** tab.

3 Click the button appropriate to the type of field you want to format (String, Number, Currency, Date, Time, Date and Time, or Boolean).

The Format Editor appears.

4 Use the Format Editor's tabs to specify the formats you want.

5 Click **OK**.

Setting default formats for Date, Time, and Date and Time fields

The following procedures first describe how to specify standard formats for Date, Time, and Date and Time fields, and then describe how to customize the formats for such fields.

Note: These default settings will affect only the objects that you subsequently add to a report. To format fields that you've already added to a report, you must right-click the field in the report and select Format Field from the shortcut menu.

To set standard default formats for Date, Time, and Date and Time fields

1 On the **File** menu, click **Options**.

2 In the Options dialog box, click the **Fields** tab.

3 To open the Format Editor, click the button appropriate to the field you want to format (**Date**, **Time**, or **Date and Time**).

Note: If you click the Date and Time button in the Format Editor, then any subsequent changes will affect "Date and Time" fields only. You must click Date or Time to format independent date fields or time fields.

4 In the Format Editor dialog box, click the **Date and Time** tab.

5 Select a predefined format from the list (or click **Customize** to create your preferred format). When you click a new format, you can preview the results in the Sample area of the Format Editor.

Note: From the list of predefined formats, you can choose the System Default settings to ensure that Crystal Reports uses the formats dictated by Windows. You can alter your system's settings in the Regional Settings Properties dialog box, located in the Control Panel.

6 Once you've selected a format, click **OK** in the Format Editor dialog box.

7 Click **OK** in the Options dialog box.

Now, when you add Date, Time, or Date and Time fields to a report, Crystal Reports should use the format you specified.

To customize formats for Date, Time, and Date and Time fields

1 On the **File** menu, click **Options**.

2 In the Options dialog box, click the **Fields** tab.

3 Open the Format Editor by clicking the button appropriate to the field you want to format (**Date**, **Time**, or **Date and Time**).

4 In the Format Editor dialog box, click the **Date and Time** tab.

5 Click **Customize** to open the Custom Style dialog box.

Note: If you chose to format "Date and Time" fields at Step 3, then you will see three tabs in the Custom Style dialog box (Date and Time, Date, and Time). The formats specified in these tabs apply only to the two elements of "Date and Time" fields, and will not affect the formats specified for independent date fields or time fields.

6 Create your preferred format by adjusting the various options in the Custom Style dialog box.

7 Once you've finished designing your format, click **OK** in the Custom Style dialog box.

8 Click **OK** in the Format Editor dialog box.

9 To format another type of field, click the appropriate button in the Options dialog box. Otherwise, click **OK** to return to Crystal Reports.

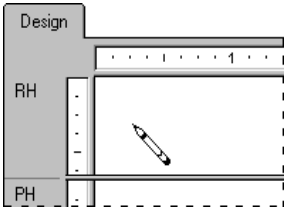
Now, when you add Date, Time, or Date and Time fields to your reports, Crystal Reports should use the customized format that you created.

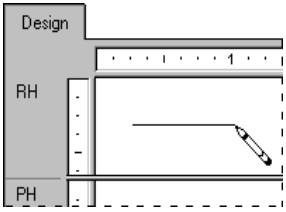
Adding and editing lines

Crystal Reports allows you to add lines to a report to emphasize important data and create professional-looking reports.

To add lines to a report

1 On the Insert Tools toolbar, click **Insert Line**.





2 Use the pencil cursor to draw the line where desired.

Note: You cannot draw diagonal lines.

To edit lines on a report

1 Right-click the line you want to format to bring up the shortcut menu.

2 On the shortcut menu, click **Format Line**.

The Format Editor dialog box appears.

3 On the **Line** tab, make the desired changes to the line.

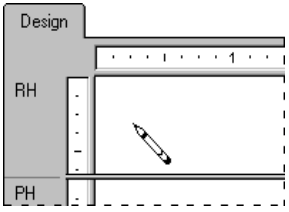
4 Click **OK** to save your changes.

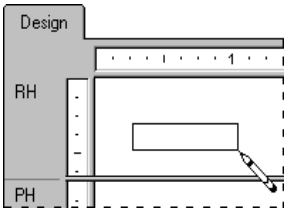
Adding and editing boxes

Crystal Reports allows you to add boxes to a report to emphasize important data and create professional-looking reports.

To add boxes to a report

1 On the Insert Tools toolbar, click **Insert Box**.





2 Use the pencil cursor to draw the box where desired.

To edit boxes on a report

1. Right-click the box you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Box**.

   The Format Editor dialog box appears.
3. On the **Box** tab, make the desired changes to the box.
4. Click **OK** to save your changes.

Adding shapes to a report

When designing report formats in Crystal Reports, you can insert a variety of shapes such as circles, eclipses, and boxes with rounded corners, as part of your report. This is especially useful for formatting reports in languages that require these shapes to effectively communicate.

To add shapes to a report

1. Add a box to your report.

   See "Adding and editing boxes" on page 203.
2. Right-click the box to bring up the shortcut menu.
3. On the shortcut menu, click **Format Box**.
4. In the Format Editor that appears, click the **Rounding** tab.
5. Select a number between 0 and 100 in the Rounding box, or move the slider to the right to increase the curvature of the box corners.

   The box that you started with gradually changes to an ellipse or circle, depending on how far you move the slider to the right.

   Note: If you have specified rounding for a box, you cannot use the Drop Shadow option that is usually available on the Box tab of the Format Editor.
6. Once the appropriate shape is created, click **OK** to save your changes.

Using conventional accounting formats

As a way of supporting the conventions used in the accounting profession, Crystal Reports lets you decide on how to display the currency symbol, negative values, and zero values on your financial reports. You can also set up your report to reverse the signs for credit and debit amounts.

To use accounting conventions in a report

1. Right-click the currency field or number field you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Field**.

   The Format Editor dialog box appears with the Number tab open.
3. In the Style area, select how you want the system number format to appear for either positive or negative values.
4. In the Currency Symbol (system default) area, specify how you want the currency symbol to appear with the values on your report.
5. Click **OK** to save your changes.

To customize the accounting conventions for a report

1. Right-click the currency field or number field you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Field**.

   The Format Editor dialog box appears with the Number tab open.
3. Click the **Customize** button.

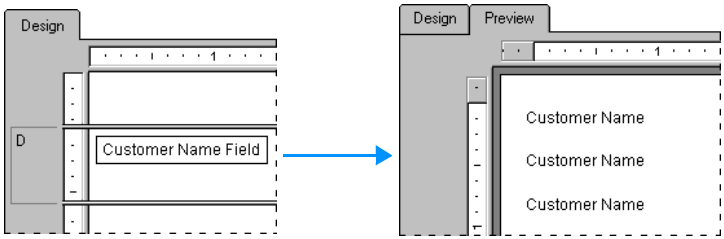
   The Custom Style dialog box appears with the Number tab open.
4. Select the **Use Accounting Format** check box.

   Once you select this option, the following conditions occur:
   - In the Negatives list, how the negative values appear on your report are determined by the Windows locale settings. The negative values are represented by either a minus sign or brackets.
   - In the Show Zero Values as list, the dash symbol is automatically selected to represent zero values on your report.
   - On the Currency Symbol tab of the Custom Style dialog box, the currency symbol is positioned on the left-side of the currency and numeric values.

   Note: Changes made to the Windows locale settings are implemented only after you exit and restart Crystal Reports.
5. Select the **Reverse Sign for Display** check box to reverse the signs for debit and credit amounts in your financial reports.
6. Click **OK** to save your changes.
7. Click **OK** again to return to your report.

Using white space between rows

The height of a section in relation to the objects within it affects the amount of white space that appears between rows on the report.



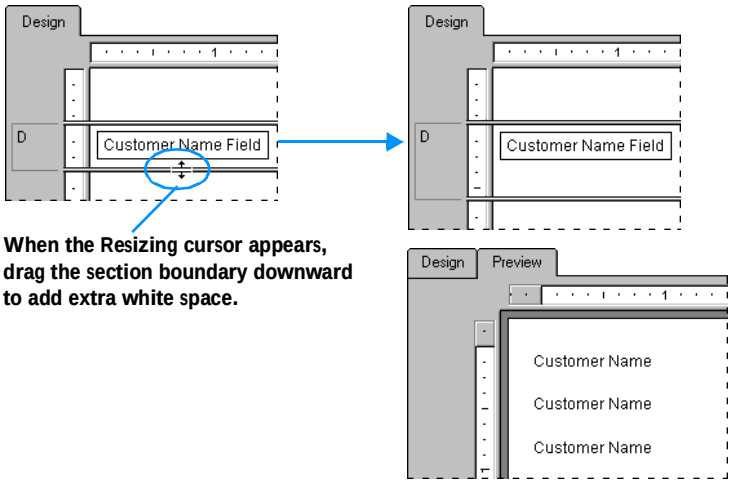
The free-form Design tab lets you add and delete white space in two ways:

- Using the Resizing cursor to resize the area on the Design tab.
- Changing the option in the Section Expert.

Note: You can also right-click the shaded area to the left of the section and select Fit Section from the shortcut menu. The program automatically resizes the section so that the bottom boundary is even with the baseline of the bottom object in the section.

Adding white space by resizing

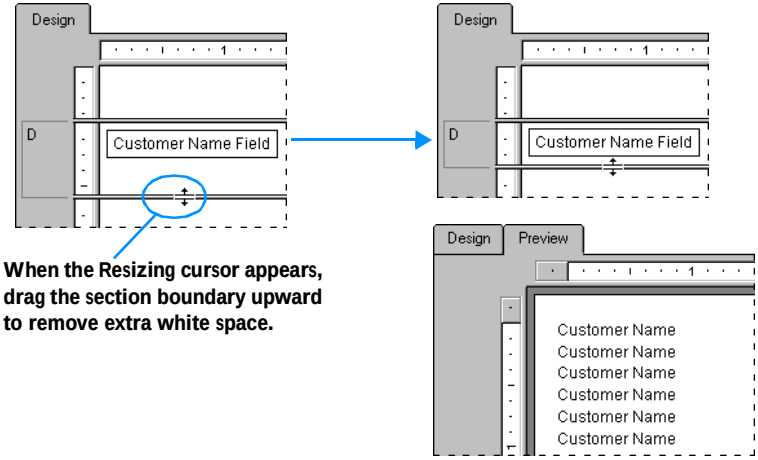
To add extra white space between rows in the report, move the pointer over the lower section boundary line. The pointer changes to a Resizing cursor.



Note: White space can also be added to a section by right-clicking the shaded area to the left of the section and selecting Insert Line from the shortcut menu. The program resizes the section automatically, adding the amount of space necessary to hold a line of typical database fields.

Deleting white space by resizing

To delete unnecessary white space within a section, move the pointer over the lower section boundary line. The pointer changes to a Resizing cursor.



When the Resizing cursor appears, drag the section boundary upward to remove extra white space.

Deleting white space by suppressing a section

If an entire section is blank (for example, if you are not putting anything into the Page Footer section of the report), you can eliminate the unnecessary white space that the Page Footer would occupy by suppressing the section in the Section Expert.

To delete white space by suppressing a section

1. On the **Report** menu, click **Section Expert**.

   The Section Expert dialog box appears.

   Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.

2. In the Sections area, click the section you want to suppress.
3. On the **Common** tab, select the **Suppress** (**No Drill-Down**) check box.
4. Click **OK** to return to your report.

   The blank section will no longer be printed.

Working with conditional formatting

Conditional formatting is formatting that applies only under certain conditions. For example, in a report you may only want:

- Customer balances printed in red if they are past due.
- The dates to appear in Day, Month, Year format if the customer is Canadian.
- Background color to appear on every other line.

Crystal Reports makes it easy to apply conditional formatting in these and hundreds of other situations.

With absolute formatting, you follow the *select, then apply* procedure. For conditional formatting, you follow the same general procedure, but you go a step further and set up conditions that determine whether or not the formatting will be applied. You specify these conditions using simple formulas. For more information on creating formulas using Crystal or Basic syntax, see the online help.

When a conditional formatting formula is set up, the formula overrides any fixed settings you have made in the Format Editor. For example, if you select the *Suppress* option, then set up a conditional formula for the *Suppress* option, the property will still apply only if the condition in the formula is met.

Crystal Reports enables you to set both on and off properties and set attribute properties conditionally. However, each of these requires a different kind of formula.

Conditional on or off properties

A conditional on or off property tests to see if a condition has been met. It is *on* if the condition is met, *off* if the condition is not met. There is no middle ground. Use Boolean formulas for this kind of formatting.

Crystal syntax example

```
condition
```

Basic syntax example

```
formula = condition
```

The program tests each value to see if it meets the condition and it returns a “yes” or “no” answer. It then applies the property to every value that returns a “yes” answer.

Conditional attribute properties

A conditional attribute property tests to see *which* of two or more conditions is met. The program then applies the formatting appropriate to the condition. For example, assume that you want values under quota printed in red and all other values printed in black. The program tests to see whether the value is under quota or not. If it is under quota, then it applies the red attribute; if it is not, then it applies the black attribute.

Use an If-Then-Else formula for this kind of conditional formatting.

Crystal syntax example

```
If Condition A Then
     crRed
Else
     crBlack
```

Basic syntax example

```
If Condition A Then
     formula = crRed
Else
     formula = crBlack
End If
```

When conditional attribute properties are set up, Crystal Reports loads a selection of attributes into the Functions list in the Formula Workshop. Double-click any of these attributes to add them to a formula. For example, if you are setting horizontal alignment conditionally, the Functions list contains attributes such as DefaultHorAligned, LeftAligned, and Justified. If you are setting borders conditionally, the Functions list contains attributes such as NoLine, SingleLine, and DashedLine.

Note: Always include the Else keyword in conditional formulas; otherwise, values that don't meet the If condition may not retain their original format. To retain the original format of values that don't meet your If condition, use the DefaultAttribute function.

Crystal syntax example

```
If Condition A Then
     crRed
Else
     DefaultAttribute
```

Basic syntax example

```
If Condition A Then
     formula = crRed
Else
     formula = DefaultAttribute
End If
```

You can take this kind of property one step further. You can specify a list of conditions and a property for each; you are not limited to two conditions. For example, if you have a number field on your report that contains sales figures from countries around the world, you can specify the number attribute(s) that you want to apply to each country. In this case, your conditions would specify that if it is from Country A, the program should apply the Country A attribute; if it is from Country B, the Country B attribute; if it is from Country C, the Country C attribute, and so on.

With more than two alternatives, use this kind of formula:

Crystal syntax example

```
If Condition A Then
     crRed
Else If Condition B Then
     crBlack
Else If Condition C Then
     crGreen
Else
     crBlue
```

Basic syntax example

```
If Condition A Then
     formula = crRed
ElseIf Condition B Then
     formula = crBlack
ElseIf Condition C Then
     formula = crGreen
Else
     formula = crBlue
End If
```

Use a multi-condition If-Then-Else formula for this kind of conditional formatting.

Changing conditional fonts

For memo or string fields that are based on conditions such as a parameter value, you can change the font, font style, size, and color for these fields using the Format Editor.

To change conditional fonts

1. Right-click the field you want to format to bring up the shortcut menu.
2. On the shortcut menu, click **Format Field**.

   The Format Editor dialog box appears.
3. Click the **Font** tab.
4. To change any of the font options, click the appropriate **Formula** button, located on the right side of the dialog box.
5. In the Formula Workshop, you can specify that conditional fonts will change only when certain conditions are met.
6. Click **Save**.

   Note:

   - If there is an error in the formula, a message box appears, asking whether to exit without examining the error. If you click No, a second message box will be displayed, detailing the error.

- If there is no error in the formula, you are returned to the Format Editor. Note that the Formula button has changed. This indicates that a formula has been entered for that property.

7 Click **OK** to return to your report.

Creating footers after the first page

You may choose to print a page footer on all pages except the first page. You can do this by formatting the Page Footer section conditionally, using an on or off property.

To create footers after the first page

1 Place the field you want displayed as a page footer in the Page Footer section of the report.

2 On the **Report** menu, click **Section Expert**.

Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.

The Section Expert dialog box appears.

3 In the Sections area, click **Page Footer**.

4 To open the Formula Workshop, click the **Formula** button, located to the right of the **Suppress** (**No Drill-Down**) check box.

5 Enter the following formula in the Format Formula Editor:

Crystal syntax example:

```
PageNumber = 1
```

Basic syntax example:

```
formula = PageNumber = 1
```

This formula suppresses the page footer on the first page, but not on any of the other pages.

6 Click **Save and close**.

Note:

- If there is an error in the formula, a message box appears, asking whether to exit without examining the error. If you click No, a second message box will be displayed, detailing the error.
- If there is no error in the formula, you are returned to the Section Expert. Note that the Formula button has changed. This indicates that a formula has been entered for that property.

7 On the Standard toolbar, click **Preview** to preview the report and ensure that the page footer appears on all pages but the first.

Note:

- If you have a multi-line page footer and have inserted the lines into separate Page Footer sections, you will need to suppress each section conditionally, using the formula above.
- To create a page header that appears on all pages but the first, place the header information in the Page Header section and then suppress that section conditionally, using the same formula that was used for suppressing the Page Footer section.

Using the Highlighting Expert

The Highlighting Expert enables you to apply conditional formatting to all types of report fields (Number, Currency, String, Boolean, Date, Time, and Date and Time fields). With the expert, you format the selected field either by specifying a condition based on that field's value, or by specifying a condition based on the value of a different report field. In other words, the expert enables you create the following formula: `If the value of field X meets condition A, then apply the specified formatting to the field selected on the report`.

When used for conditional formatting, the Highlighting Expert allows you to:

- Modify several attributes at once, without writing a formula.
- Highlight all field types used in the report.
- Format font style, background color, font color, and border style.
- Format a field based on its own values or the values of another field.
- Highlight a cross-tab or OLAP cell based on row and column heading values.
- Enter values using your locale-specific number format (such as 1,224.23 for North American users).
- Enter dates numerically or textually (January 12, 2001, or Jan 12, 2001).
- Undo highlighting quickly.

When you need to conditionally format report fields, the Highlighting Expert is quicker and easier to use than the Formula Workshop. The Highlighting Expert is most commonly used to highlight field values that are in some way distinguished from other values in the report. You might, for example, highlight your key customers by printing the {Customer.Last Year's Sales} field with a red background whenever the sales value exceeds $50,000. Alternatively, to draw attention to outstanding orders, you might bold the {Product.Product Name} field whenever the {Orders.Shipped} value is False.

The Highlighting Expert, however, is not as flexible as the Formula Workshop. To use the formatting capabilities of Crystal Reports to their full potential, create your own conditional formatting formulas with the Formula Workshop (accessible through the Format Editor dialog box). For complete details, see "Using Formulas" on page 329.

Conditionally formatting fields using the Highlighting Expert

The Highlighting Expert is a quick alternative to the Formula Workshop; it allows you to conditionally format any of your report fields. The Highlighting Expert essentially enables you create the following formula: `If the value of field X meets condition A, then apply the specified formatting to the field selected on the report.`

The dialog box is divided into two areas: the Item list area displays the formula; the Item editor area allows you to specify the formula. The Item editor area includes a Sample field that displays the formatting specifications being applied.

To conditionally format fields using the Highlighting Expert

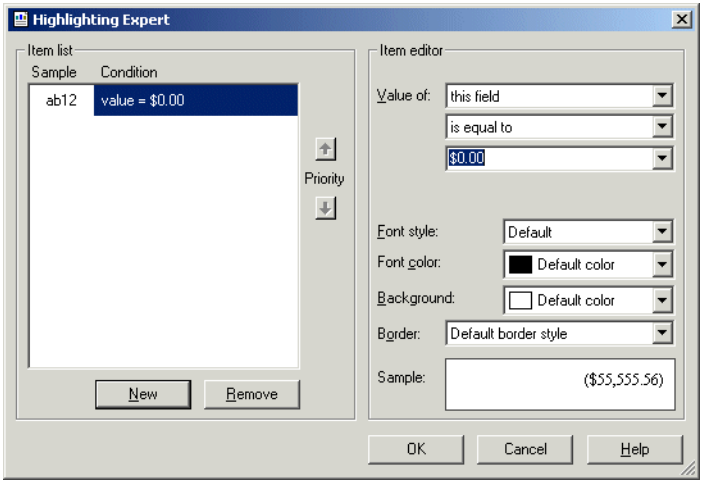
1. To open the Highlighting Expert, right-click the field you want to format and select **Highlighting Expert** from the shortcut menu.

   Tip: You can also start the expert by clicking the Highlighting button on the Expert Tools toolbar, or by clicking Highlighting Expert on the Format menu.

   When opened, the expert is set to format the field that is currently selected on the report.

2. In the Highlighting Expert, click **New** to create a new conditional formula with default settings.



3. In the Item editor area, click the **Value of** list and select the field that you want to base your condition on.

   The field chosen here is the field upon which your condition is based; this field need not be the field that is being formatted. To create a condition based on the values of the field that is being formatted, select “this field” from the list. To base your condition on a different report field, select it from the list of available fields.

Note: The "Value of" list displays only those fields that you have added to the report.

4 Select a comparison from the second list (is equal to, is less than, and so on).

This comparative statement works as the operator in the conditional formula created by the expert.

5 Complete the condition by entering the desired value in the box.

Note: If the field selected in the "Value of" list is not numeric, the text box turns into a list of available values, from which you must select one.

6 In the **Font style**, **Font color**, **Background**, and **Border** lists, specify the formatting changes that you want to apply to the selected field when your condition is met.

7 Repeat steps 3 and 4 if you want to apply multiple highlighting conditions to the selected field.

Note: You can use the expert's Remove button to delete highlighting formulas from the list.

8 Use the **Priority** arrows to specify the order in which you want Crystal Reports to apply your conditions. For details, see "Setting highlighting priorities" on page 214.

9 Click **OK** to return to your report.

Setting highlighting priorities

The Priority buttons in the Item list area of the Highlighting Expert allow you to set priorities for your formulas. This is useful when you have two or more formulas that could offer conflicting results in some situations.

For example, suppose that you highlight the Unit Price field on the report. You assign to this field a highlighting formula that shows a yellow background when a unit price is greater than $100. Then, on this same report, you create another highlighting formula that shows a red background when a unit price is greater than $200. Considering that 100 is a subset of 200, you could have Unit Price fields with yellow backgrounds when, in fact, those fields should have red backgrounds. In other words, a unit price of $300 could receive either a red or a yellow background, depending on which formula has been assigned priority.

To set priorities for highlighting formulas

1 On the **Format** menu, click **Highlighting Expert**.

2 In the Item list area, select one of the conditional highlighting formulas that you have created.

3 Click the **Priority** arrows to move the selected formula to a position above or below the other formula(s).

Note: A formula has priority over another formula when it is higher in the Items list area.

4 Click **OK**.

5 Click the **Preview** tab or refresh the report to see the highlighting changes.

Undo/Redo activities

Crystal Reports includes multiple levels of undo. With multiple levels of undo, you can undo any number of changes to an object, in reverse order, until you have your report in the condition you want it.

The program also has a redo feature that reverses an undo. If you move an object, for example, and do not like its new position, you can click Undo to move it back to its original position. If you then change your mind, you can click Redo to restore the latest change.

The Undo and Redo buttons have lists that allow you to undo or redo a number of changes at one time.

- To undo an action, click **Undo** on the Standard toolbar.

  The first time the button is clicked, it reverses the most recent change made to the report. Each additional time the button is clicked, it reverses the next most recent change.

  To undo several actions at once, click the arrow button to display the list of actions. Select the series of actions you wish to undo.

- To redo a change after you have undone it, click **Redo** on the Standard toolbar.

  The program disables the Undo button and the Undo/Redo commands whenever there is nothing to undo/redo or when you have made a change that cannot be reversed.

  To redo several actions at once, click the arrow button to display the list of actions. Select the series of actions you wish to redo.

Note: You can only undo or redo actions in order from the most recent backward. You cannot undo an action without undoing more recent actions.

Charting 13

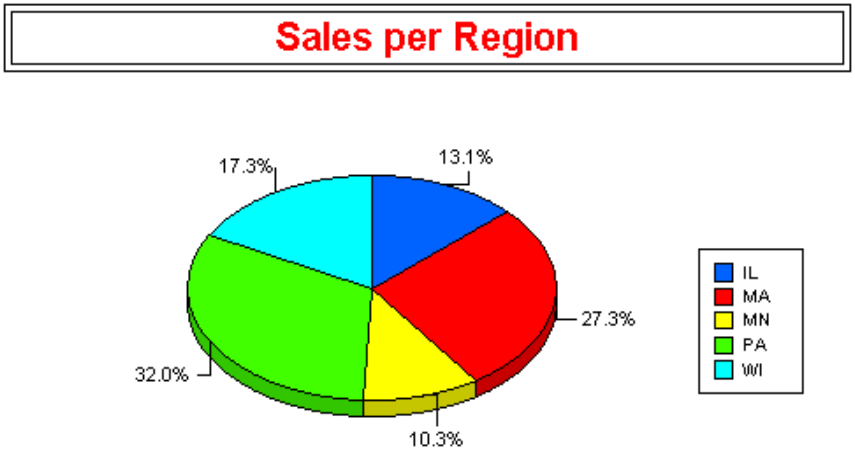
Crystal Reports enables you to present summarized data in colorful, easy-to-read charts. This chapter demonstrates how to create charts and how to use them in reports to make report data more meaningful and easier to understand. You can choose from a number of chart layouts and types, as well as drill down to see the details behind the graphical summaries and format chart objects.

Charting concepts

Charting overview

Crystal Reports enables you to include sophisticated, colorful charts in your reports. You can use charts any time you want to improve the usefulness of a report.

For example, if you have a sales report grouped by Region with a subtotal of Last Year's Sales for each region, you can quickly create a chart that will display Sales per Region.



You can chart on the following:

- Summary and subtotal fields.
- Detail, formula, and Running Total fields.
- Cross-Tab summaries.
- OLAP data.

You will typically chart on summary and subtotal information at the group level. However, depending on the type of data you are working with, you can create an Advanced, Cross-Tab, or OLAP grid chart for your report.

Chart layouts

The Chart Expert provides four layouts that correspond to certain sets of data.

You can create charts with any of the following layouts, and depending on the data you are using, you can change the chart from one layout to another.

Advanced

Use the Advanced layout when you have multiple chart values or when you do not have any group or summary fields in the report.

The Advanced chart layout supports one or two condition fields: with these condition fields, you can create a 2-D, 3-D, or pie chart. Other specific functions with the Advanced layout include:

- Values can be grouped in ascending, descending, or specified order, as well as by Top N or Sort totals.
- Values can be plotted for each record.
- Values can be plotted as a grand total for all records.
- Charts can be based on formula and Running Total fields.

Group

The Group layout is a simplified layout in which you show a summary on change of field for topics such as Country.

Note: In order to create a chart using the Group layout, you must have at least one group and at least one summary field for that group.

Cross-Tab

Use the Cross-Tab layout to chart on a Cross-Tab object. A Cross-Tab chart uses the fields in the cross-tab for its condition and summary fields.

OLAP

Use the OLAP layout to chart on an OLAP grid. An OLAP chart uses the fields in the OLAP grid for its condition and summary fields.

Chart types

Different sets of data are particularly suited to a certain chart type. The following is an overview of the main chart types and their most common uses.

Bar

Most bar charts (also known as a column chart) display or compare several sets of data. Two useful bar charts are the Side-by-Side bar chart and the Stacked bar chart.

- Side-by-Side bar chart
  A Side-by-Side bar chart displays data as a series of vertical bars. This type of chart is best suited for showing data for several sets over a period of time (for example, last year's sales figures for AZ, CA, OR, and WA).
- Stacked bar chart
  A Stacked bar chart displays data as a series of vertical bars. This type of chart is best suited for representing three series of data, each series represented by a color stacked in a single bar (for example, sales for 1997, 1998, and 1999).

Line

A line chart displays data as a series of points connected by a line. This type of chart is best suited for showing data for a large number of groups (for example, total sales over the past several years).

Area

An area chart displays data as areas filled with color or patterns. This type of chart is best suited for showing data for a limited number of groups (for example, percentage of total sales for AZ, CA, OR, and WA).

Pie

A pie chart displays data as a pie, split and filled with color or patterns. Pie charts are typically used for one group of data (for example, the percentage of sales for the entire inventory); however, you have the option to choose multiple pie charts for multiple groups of data.

Doughnut

A doughnut chart is similar to a pie chart, displaying data as sections of a circle or doughnut. If, for example, you charted sales by region on a particular report, you would see the total number of sales (the figure) in the center of the doughnut and the regions as colored sections of the doughnut. As with the pie chart, you have the option to choose multiple doughnut charts for multiple groups of data.

3-D Riser

A 3-D Riser chart displays data in a series of 3-dimensional objects, lined up side-by-side, in a 3-dimensional plane. The 3-D Riser chart shows the extremes in your report data. For example, the differences between sales by customer by country are visually dynamic when presented in this chart.

3-D Surface

3-D Surface charts present a topographic view of multiple sets of data. If, for example, you need a chart to show the number of sales by customer by country, in a visually dynamic and relational format, you might consider using the 3-D Surface chart.

XY Scatter

An XY Scatter chart is a collective of plotted points that represent specific data in a pool of information. The XY Scatter chart allows the user to consider a larger scope of data for the purpose of determining trends. For example, if you input customer information, including sales, products, countries, months, and years, you would have a collective of plotted points that represents the pool of customer information. Viewing all of this data on an XY Scatter chart would allow you to speculate as to why certain products were selling better than others or why certain regions were purchasing more than others.

Radar

A radar chart positions group data, such as countries or customers, at the perimeter of the radar. The radar chart then places numeric values, increasing in value, from the center of the radar to the perimeter. In this way, the user can determine, at a glance, how specific group data relates to the whole of the group data.

Bubble

A bubble chart (an extension of the XY Scatter chart type) displays data as a series of bubbles, where the size of the bubble is proportional to the amount of data. A bubble chart would be very effective with the number of products sold in a certain region; the larger the bubble, the greater the number of products sold in that region.

Stock

A stock chart presents high and low values for data. It is useful for monitoring financial or sales activities.

Numeric Axis

A numeric axis chart is a bar, line, or area chart that uses a numeric field or a date/time field as its "On change of" field. Numeric axis charts provide a way of scaling your X-axis values, thus creating a true numeric X-axis or a true date/time X-axis.

Gauge

A gauge chart presents values graphically as points on a gauge. Gauge charts, like pie charts, are typically used for one group of data (for example, the percentage of sales for the entire inventory).

Gantt

A Gantt chart is a horizontal bar chart often used to provide a graphical illustration of a schedule. The horizontal axis shows a time span, while the vertical axis shows a series of tasks or events. Horizontal bars on the chart represent event sequences and time spans for each item on the vertical axis. You should use only date fields when creating a Gantt chart. The field you choose for the data axis should be set to "For Each Record," and the start and end-date fields should be added to the "Show value(s)" area of the Chart Expert's Data tab.

Where to place a chart

The placement of a chart determines which data is displayed and where it is printed. For example, if you place a chart in the Report Header section, the chart includes data for the entire report. If you place it in a Group Header or Group Footer section, it displays group specific data.

Note: If your report contains subreports, you can place charts in those subreports as well. See "Subreports" on page 365.

Drill down with charts

Not only is charting a means of presenting data—it is also an analysis tool. Move your cursor over a section of the group chart on the Preview tab, so that the pointer becomes a Drill-down cursor, then double-click to view the underlying details for that section of the chart.

Drill-down with legends

If the chart consists of one or more group fields, you can use the chart legend to drill down on individual groups. Double-click the drill-down cursor on the markers and text in the legend to view the details about that section of the chart.

Creating charts

Charting on details or formula fields (Advanced layout)

The Advanced layout allows you to create a chart based on specific values. Since charts are a good way to display summarized information, they are often based on a summary field in your report. With an Advanced layout, you can create a chart without the need for a summary field by using values that appear in the Details section of your report.

To create a chart based on the Advanced layout, you must specify two things:

- Conditions (there can be two).
- Values (there can be multiple values).

Condition

The condition is used to indicate when to plot the point. For example, a chart showing last year's sales for your customers uses the Customer Name field as the condition. Each time the condition changes (the customer name changes), a point is plotted.

You also have the option of plotting a point for each record, or plotting one point for all records.

Value

The Advanced chart uses the value to indicate what information is plotted as the points on your chart. For example, to create a chart showing last year's sales for your customers, the Last Year's Sales field would be the value.

To chart on a details or formula field

1 On the **Insert** menu, click **Chart**.

Tip: Another way to do this is to click the Insert Chart button on the Insert Tools toolbar.

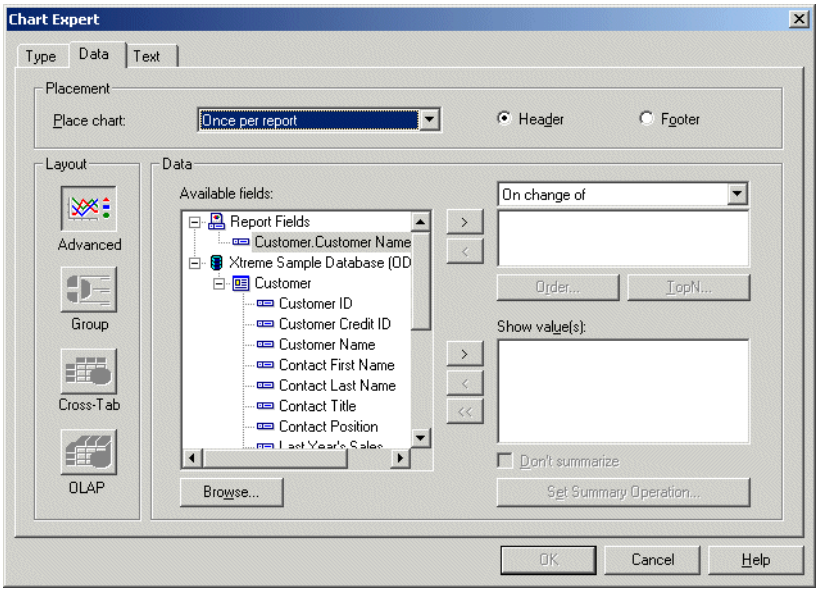
The Chart Expert dialog box appears.

2 On the **Type** tab, in the **Chart type** list, select a chart type. Then click the chart subtype that best illustrates your data. See "Chart types" on page 219.

3 Select **Automatically set chart options** if you want the Chart Expert to use the default options on the Axes and Options tabs.

Note: If you want to select options on the Axes and Options tabs manually, ensure that "Automatically set chart options" is not selected.

4 Click the **Data** tab.



5 In the Layout area, click **Advanced**, if it is not already selected.

6 In the Placement area, in the **Place chart** list, specify how often your chart appears on the report, then click **Header** or **Footer** to specify where to place your chart.

7 In the Data area, specify the database fields you want to use as conditions.

You can select **On change of** from the list, then add up to two database fields in the box underneath the list.

The arrow buttons on the Chart Expert dialog box allow you to move fields from one list to the other. Single arrows move only the selected field; double arrows move all fields at the same time.

8 Add the database fields you want to use as values to the **Show value(s)** list.

9 If you do not want Crystal Reports to automatically summarize the chart values for a formula field, select the **Don't summarize** check box.

10 If the **Axes** and **Options** tabs appear, you can customize some of the chart's properties, such as the scaling of the axes, the legend, and the data points.

11 Click the **Text** tab.

12 Accept the default title information or add new titles to your chart.

13 Click **OK**.

Your chart is placed in the Header or Footer section of the report, depending on your selection in Step 7. You may have to refresh your report to preview the finished chart.

Note: When your chart is inserted, it may cover a portion of the report. Move and resize the chart so that it fits properly within the report.

Charting on summary or subtotal fields (Group layout)

Many of the charts you create are based on summary or subtotals within your report. In order to create these charts, you must have a summary or subtotal already inserted into your report in a group header or footer. For more information on inserting summaries and subtotals, see "Summarizing grouped data" on page 135 and "Subtotaling" on page 139.

To chart on a summary or subtotal field

1 On the **Insert** menu, click **Chart**.

The Chart Expert dialog box appears.

Tip: Another way to do this is to click the Insert Chart button on the Insert Tools toolbar.

2 On the **Type** tab, in the **Chart type** list, select a chart type. Then click the chart subtype that best illustrates your data. See "Chart types" on page 219.

3 Select **Automatically set chart options** if you want the Chart Expert to use the default options on the Axes and Options tabs.

Note: If you want to select options on the Axes and Options tabs manually, ensure that "Automatically set chart options" is not selected.

4 Click the **Data** tab.

5 In the Layout area, click **Group**, if it is not already selected.

6 In the Placement area, in the **Place chart** list, specify how often your chart appears on the report, then click **Header** or **Footer** to specify where to place your chart.

Note: The "Place chart" list includes all groups in your report that have summary fields except for the inner-most group.

7 In the Data area, in the **On change of** list, click the group field you want to base your chart on; then, in the **Show** list, click the summary field you want to display on your chart.

8 If the **Axes** and **Options** tabs appear, you can customize some of the chart's properties, such as the scaling of the axes, the legend, and the data points.

9 Click the **Text** tab.

10 Accept the default title information or add new titles to your chart.

11 Click **OK**.

Your chart is placed in the Header or Footer section of the report, depending on your selection in Step 6. You may have to refresh your report to preview the finished chart.

Note: When your chart is inserted, it may cover a portion of the report. Move and resize the chart so that it fits properly within the report.

Charting on Cross-Tab summaries (Cross-Tab layout)

Crystal Reports allows you to include a chart based on summary values in your Cross-Tab report. For example, with a Cross-Tab that shows the amount of a certain product sold in each region of the United States, you may want to include a chart showing the percentage of total sales provided by each region for that product.

To create a Cross-Tab chart, you must first have a Cross-Tab in your report. For more information, see "Cross-Tab Objects" on page 261.

To chart on a Cross-Tab summary

1 Select the Cross-Tab on which you want to chart.

2 On the **Insert** menu, click **Chart**. The Chart Expert dialog box appears.

Tip: Another way to do this is to click the Insert Chart button on the Insert Tools toolbar.

3 On the **Type** tab, in the **Chart type** list, select a chart type. Then click the chart subtype that best illustrates your data. See "Chart types" on page 219.

4 Select **Automatically set chart options** if you want the Chart Expert to use the default options on the Axes and Options tabs.

Note: If you want to select options on the Axes and Options tabs manually, ensure that "Automatically set chart options" is not selected.

5 Click the **Data** tab.

6 In the Layout area, click **Cross-Tab**, if it is not already selected.

7 In the Placement area, click **Header** or **Footer** to specify where to place your chart.

Note: How often your chart appears on the report depends on where the Cross-Tab summary field has been placed.

8 In the Data area, in the **On change of** list, click the group field you want to base your chart on.

9 If necessary, in the **Subdivided by** list, click a secondary row or column you want to base your chart on.

10 In the **Show** list, click the summary field you want to display on your chart.

11 If the **Axes** and **Options** tabs appear, you can customize some of the chart's properties, such as the scaling of the axes, the legend, and the data points.

12 Click the **Text** tab.

13 Accept the default title information or add new titles to your chart.

14 Click **OK**.

Your chart is placed in the Header or Footer section of the report, depending on your selection in Step 7. You may have to refresh your report to preview the finished chart.

Note: When your chart is inserted, it may cover a portion of the report. Move and resize the chart so that it fits properly within the report.

Charting on an OLAP cube (OLAP layout)

The OLAP layout lets you chart on an OLAP grid. In order to create an OLAP chart, you must first have an OLAP grid in your report. For more information, see "Creating an OLAP report" on page 279.

To chart on an OLAP cube

1 Select the OLAP grid on which you want to chart.

2 On the **Insert** menu, click **Chart.** The Chart Expert dialog box appears.

Tip: Another way to do this is to click the Insert Chart button on the Insert Tools toolbar.

3 On the **Type** tab, in the **Chart type** list, select a chart type. Then click the chart subtype that best illustrates your data. See "Chart types" on page 219.

4 Select **Automatically set chart options** if you want the Chart Expert to use the default options on the Axes and Options tabs.

Note: If you want to select options on the Axes and Options tabs manually, ensure that "Automatically set chart options" is not selected.

5 Click the **Data** tab.

6 In the Layout area, click the **OLAP** button, if it is not already selected.

7 In the Placement area, click **Header** or **Footer** to specify where to place your chart.

Note: The frequency with which your chart appears in the report depends on the report section that the OLAP grid has been placed in.

8 In the Data area, in the **On change of** list, click the field you want to base your chart on.

9 If necessary, in the **Subdivided by** list, click a secondary row or column you want to base your chart on.

Note: Be sure that the chart type selected in Step 3 supports a secondary charting field.

10 If the **Axes** and **Options** tabs appear, you can customize some of the chart's properties, such as the scaling of the axes, the legend, and the data points.

11 Click the **Text** tab.

Accept the default title information or add new titles to your chart.

12 Click **OK**.

Your chart is placed in the Header or Footer section of the report, depending on your selection in Step 7. You may have to refresh your report to preview the finished chart.

Note: When your chart is inserted, it may cover a portion of the report. Move and resize the chart so that it fits properly within the report.

Working with charts

Once you have created a chart, you may want to add a new title, headings, or a legend, change fonts, or even change the type of chart. Crystal Reports provides many options for working with your existing charts; learn more by choosing a topic from the following list:

- "Editing charts using the Chart Expert" on page 227
- "Editing charts using the Chart Options menu items" on page 228
- "Using the zooming features with bar and line charts" on page 228
- "Auto-arranging charts" on page 229
- "Formatting charts" on page 229
- "Using the underlay feature with charts" on page 230

Editing charts using the Chart Expert

Editing charts with the Chart Expert allows you to return to the expert in which you designed your chart. You can modify many of your original choices, such as the type of chart to display, the data on which the chart is based, and so on.

To edit a chart using the Chart Expert

1 Right-click the chart to bring up the shortcut menu.

2 On the shortcut menu, click **Chart Expert**.

3 In the Chart Expert dialog box, make the desired changes.

4 Click **OK**.

Editing charts using the Chart Options menu items

Some of the editing options available in the Chart Expert are also available directly from the Chart Options menu that's available when you right-click a chart. You can perform the following operations by using the commands on the shortcut menu:

- Apply a new template to the chart.
- Change the template specifications of the chart.
- Change the chart titles.
- Change the numeric axis grids and scales of the chart.
- Format the selected chart.

For more information about how to use the commands available on the Chart Options menu, see the Chart Help online help (Sscdlg.hlp)—installed by default in the `\Program Files\Common Files\Crystal Decisions\2.0\bin` directory.

Using the zooming features with bar and line charts

On the Preview tab, you can find commands for zooming bar charts and line charts within your report. You have the ability to zoom in and out on these chart types at any time, with each time being referred to as *instance-specific*. If you decide to save the instance of the chart that has been zoomed in or out, you must save the data with the report.

To zoom in and out on a bar or line chart

1. On the **Preview** tab, right-click the bar or line chart to bring up the shortcut menu.
2. On the shortcut menu, click **Zoom In**.
3. Drag the Zoom In cursor around a section of the chart to enclose it within the tracking rectangle.

   The chart zooms in to the section you selected.
4. Repeat the previous step to zoom in further.

   Note: To see adjacent areas on a zoomed-in chart view (neighboring bars in a bar chart, for example), use the Pan option on the shortcut menu. Pull the Pan cursor to the left or right to move in that direction.
5. To zoom out on a chart, right-click the chart to bring up the shortcut menu.
6. On the shortcut menu, click **Zoom Out**.
7. With the Zoom Out cursor, click the chart.

   The chart zooms out one level of magnification.
8. Click the chart again to zoom out further.

Auto-arranging charts

If you move or resize chart objects on the Preview tab, select the Auto-Arrange Chart feature to reset the chart.

To auto-arrange a chart

1. On the **Preview** tab, right-click the chart to bring up the shortcut menu.
2. On the shortcut menu, click **Auto-Arrange Chart**.

   Crystal Reports resets the chart to its original size and position.

Formatting charts

Changing the border of a chart

1. On the **Design** or **Preview** tab, right-click the chart to bring up the shortcut menu.
2. On the shortcut menu, click **Format Chart**.
3. On the Format Editor dialog box, click the **Border** tab to see its options.
4. Change the line style, color, background color, and add or remove a drop shadow from the chart border.
5. Click **OK** to save your changes.

   Crystal Reports returns you to the report and implements your changes.

Conditionally formatting a chart

1. On the **Design** or **Preview** tab, right-click the chart to bring up the shortcut menu.
2. On the shortcut menu, click **Chart Expert**.
3. Clear **Automatically set chart options** if it is selected, and click the **Options** tab.
4. In the Chart color area, select **Color**.

   To set conditional color formatting, click the Format button and use the Chart Color Format Expert to set your conditions. For information about the expert, search the online help for Chart Color Format Expert.

   Note:

   - The option to apply conditional formatting is not available for every chart type.
   - An area chart must have two “On change of” values for conditional formatting to appear.
   - If your chart type is line, the chart must have data markers before you can see conditional formatting.

- After you apply conditional formatting, you must select “Color by Group” on the Look tab of the Chart Options dialog box before you will be able to see your formatting. To set this option, right-click your chart, point to Chart Options on the shortcut menu, and select General from the submenu. This note applies to line, 3-D Riser, and 3-D Surface charts with one “On change of” field, as well as to bar, numeric axis, 3-D Surface, radar, stock, and charts with two summaries.

5 Click **OK**.

Changing the chart’s legend text

1 On the **Preview** tab, click the text in your chart’s legend to select it.

Tip: Be sure to select the text and not the entire legend.

2 Right-click the legend text and select **Edit Axis Label** from the shortcut menu.

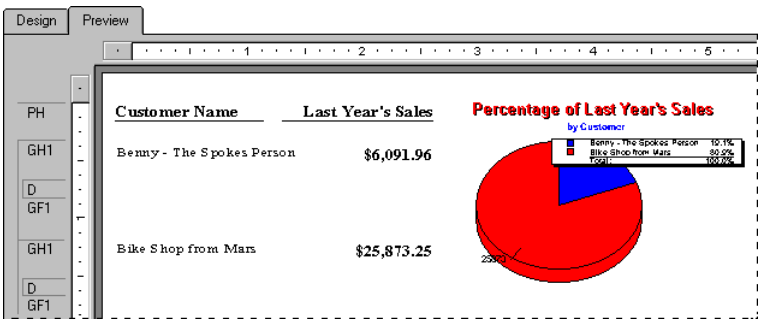
3 In the Label Aliasing dialog box, add the text you want to see in the **Displayed Label** field.

4 Click **OK**.

Using the underlay feature with charts

Since charts can print only in certain sections of your report, the underlay feature gives you more control in the overall look of your report. Instead of having a chart print ahead of the data it represents, it can appear next to the data for a more comprehensible report.

This is how your report looks when you underlay a chart with report data.



To underlay a chart

1 Create your chart and place it in the Report Header section. See “Creating charts” on page 222.

2 On the **Report** menu, click **Section Expert**.

Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.

The Section Expert dialog box appears.

3 In the Sections area, click **Report Header**, then select the **Underlay Following Sections** check box.

4 Click **OK**.

Crystal Reports returns you to the report. Your chart will now underlay the sections below it.

5 If necessary, move or resize the chart.

Mapping 14

Crystal Reports enables you to include maps with reports made up of geographic data. This chapter explains how to use maps in reports to make report data more meaningful and easier to interpret. You can customize and rearrange the appearance of a map and activate the drill-down mode to view the details behind the graphical summaries.

Mapping concepts

Mapping overview

With Crystal Reports, you can include geographic maps in reports. Maps help you analyze report data and identify trends more efficiently. For example, you could create a map that shows sales by region. You would then be able to:

- Use one of the five map types to analyze the data.
- Adjust the appearance and organization of the map (allowing you to better identify trends).
- Drill down on the map regions to view underlying data.

Map layouts

The Map Expert provides four layouts that correspond to certain sets of data.

You can create maps with any of the following layouts, and depending on the data you are using, you can change the map from one layout to another.

Advanced

Use the Advanced layout when using multiple map values or when you do not have any groups or summaries in the report.

Group

The Group layout is a simplified layout in which you show a summary on change of a geographic field (such as Region).

Note: In order to create a map using the Group layout, you must have at least one group and at least one summary field for that group.

Cross-Tab

Use the Cross-Tab layout when mapping on a Cross-Tab object. A Cross-Tab map does not require groups or summary fields.

OLAP

Use the OLAP layout when mapping on an OLAP grid. An OLAP map does not require groups or summary fields.

Note: If there is no map associated with the data you specify, then an empty block will appear unless the section that the map is placed in has been formatted to suppress if blank.

Map types

The Map Expert also provides five basic map types, each suitable for a different strategy of data analysis. When deciding which map type best fits your report, you should consider the type of data you want to analyze. For example, if you want the map to display one data item for each geographic division (city, state, country, and so on), then you might use a Ranged, Dot Density, or Graduated map. On the other hand, if you want the map to display more than one value for each new geographic division, then you could use a Pie Chart map or a Bar Chart map. The following is an overview of the main map types and their most common uses.

Ranged

A Ranged map breaks the data into ranges, assigns a specific color to each range, then color codes each geographic area on the map to display the range. For example, you could create a map that displays Last Year's Sales by Region. If you have sales ranging from zero to 100,000, you might give the map five ranges, with equal intervals of 20,000 each. You could use shades of red (going from dark to light red) to color code each region according to those sales figures. Then you could use this map to see where sales are the highest.

With equal intervals, you might end up with all your regions ranging between zero and 20,000, except perhaps one region (for example, California) that might have exceptionally high sales (such as 98,000). This map would be a very distorted representation of the data. A more useful map would have ranges like 0-5000, 5000-10000, 10000-15000, 15000-20000, and over 20000. It is important to carefully define your ranges.

Note: The end value for each division is repeated as the start value for the next division; the actual value is included in the group it starts. That is, the ranges in the previous example are actually: 0-4999, 5000-9999, and so on.

There are four different distribution methods for ranged maps.

- **Equal count**
  This option assigns intervals in such a way that the same number of regions (or as close to the same number of regions as possible) appear in each interval. In other words, this option would assign intervals so that each color on the map is assigned to the same number of regions. The numeric quantity of the summary values in each interval may or may not be equal, depending on the individual regions and their summary values.
- **Equal ranges**
  This option assigns intervals in such a way that the summary values in each interval are numerically equal. The number of regions in each interval may or may not be equal, depending on the individual regions and their summary values.

- **Natural break**
  This option assigns intervals using an algorithm that attempts to minimize the difference between the summary values and the average of the summary values for each interval.
- **Standard deviation**
  This option assigns intervals in such a way that the middle interval breaks at the mean (or average) of the data values, and the intervals above and below the middle range are one standard deviation above or below the mean. Standard deviation is the square root of the variance. It projects how various values in a set of values deviate from the mean for that set.

Dot Density

A Dot Density map displays a dot for each occurrence of a specified item. For example, you might create a United States map that shows one dot for each shipbuilder in the nation. In states like Tennessee, there would be no dots. However, in some coastal states, such as South Carolina, you might be able to count the dots on the map, since their dispersal would be fairly wide.

The purpose of a Dot Density map is to provide an overall impression of the distribution of the specified item. A Dot Density map is much like a nighttime satellite photo of the United States, where you can see the lights of all the cities. Such a map is not a very accurate means of communicating information (particularly if you have a large number of items), but it is a good way to give an overview of the distribution.

Graduated

A Graduated map is much like a Ranged map; it shows one symbol per instance of a specified item. This symbol is a circle by default, but you can choose a different symbol if you prefer. Each symbol is proportional in size to the value of the item it represents (within a range of three sizes).

A Graduated map communicates the same information as the Ranged map, but you would usually create a Ranged map for a case in which the geographic areas have distinct geographic boundaries (as in the case of Regions), while you would use a Graduated map for displaying data that is linked to points rather than precise areas (as in the case of Cities).

For example, a map of an individual region could use graduated circles to represent the sales for each office. The size of each circle would be proportional to the sales (or to the sales range) of the office it represents. On this map, an office with a sales figure of $70,000 might have a large circle, and an office with a sales figure of $20,000 might have a small circle. So, a Graduated map provides a more efficient representation of point data (e.g. Cities) than a Ranged map does, and it uses sized symbols rather than colors to distinguish variations in the values of the items it represents.

Pie Chart

A Pie Chart map displays a pie chart over each geographic area. The pie charts represent data items that make up a whole. Each slice of the pie represents an individual data item and shows that item's percentage in the whole. For example, you could create a Pie Chart map showing heating fuel types by region. You might have four types of heating fuel (four slices in each pie): electricity, gas, wood, and solar. Each region would then have a pie chart showing the breakdown of heating fuel types within that region. Washington state would probably use a high percentage (a large slice of the pie) of electricity because of the hydropower in that region, while Idaho would probably use a high percentage (a large slice of the pie) of natural gas.

You can use this map type to compare the distribution of several items within a particular region. You can also specify that the pie charts be sized proportionately so that, as with the symbols in a Graduated map, the pie charts will appear in various sizes, depending on the underlying data values. This will allow you to compare the totals between regions.

Bar Chart

A Bar Chart map works like a Pie Chart map, but may be more useful for certain sets of data. Typically, you would use a Bar Chart map for items that do not total 100%; that is, for data items that do not make a whole, or for data items that are unrelated. For example, you could create a Bar Chart map that displays use of heating fuel by region. You might choose to analyze use of three types of fuel: electricity, gas, and solar. Each bar chart on the map could contain individual bars for each of these types. In this example, the data items (electricity, gas, and solar) do not comprise a whole. There may be other types of fuel used in these regions, such as wood, but this map only focuses on three of them. Also, the purpose of the map is to compare each region's use of each fuel type with that of every other region. In a Pie Chart map, you could show these three fuel types as percentages of the entire fuel use within each region, and though you could compare the percentages for each region, you would probably not be able to compare the actual values for each region because each region would have the same total value (100%).

Where to place a map

When you choose where to place the map, you determine the amount of data that will be included in the map. For example, if you place the map in the Report Header section, the map includes data for the entire report. If you place it in a Group Header or Group Footer section, it displays group-specific data. This choice will also determine whether the map prints once for the entire report, or many times (once for each instance of a given group).

Note: If your report contains subreports, you can place maps in those subreports as well.

Drill down with maps

Not only is mapping a means of presenting data—it is also an analysis tool. Move your cursor over a section of the map on the Preview tab, so that the pointer becomes a Drill-down cursor, then double-click to view the underlying details for that section of the map.

Note: If you drill down on a region that has no data associated with it, you will get a message saying, "There are no detail records for that {Region Name}."

Creating maps

The process for creating a map varies depending on the data you are mapping on. The following sections detail the mapping process for each map layout.

Mapping on details fields (Advanced layout)

The Advanced layout allows you to create a map based on specific values. With an Advanced layout, you can create a map without the need for a summary field by using values that appear in the Details section of your report.

To create a map based on the Advanced layout, you must specify two things:

- Conditions (there can be two).
- Value (there must be only one value).

Condition

The condition is used to indicate where to plot the data on a map. This condition must be a string field. In order for a map to be generated, the field should contain geographic information. For example, a map showing last year's sales for each country uses the Country field as the condition. Each time the condition changes (the country changes), that area on the map is highlighted.

Value

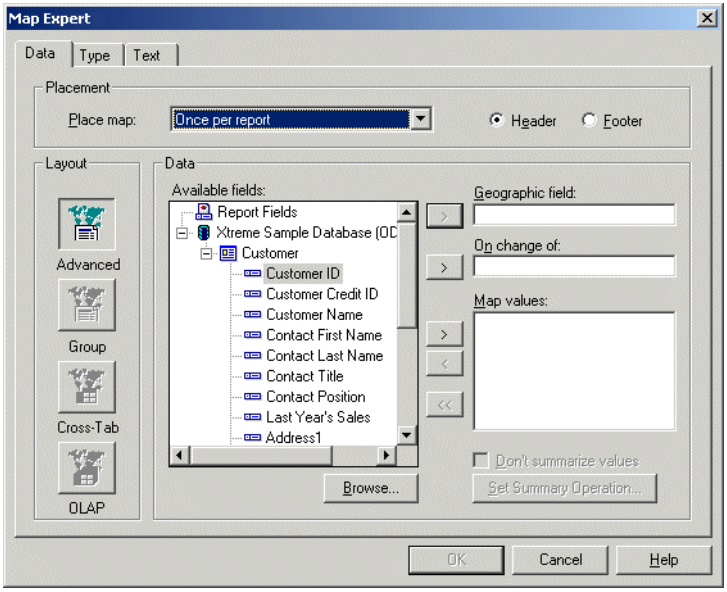
The Advanced layout uses the value to indicate what information is mapped when the area on the map is highlighted. For example, to create a map showing last year's sales for the countries, the Last Year's Sales field would be the value.

To map on a details field

1. On the **Insert** menu, click **Map**.

Tip: Another way to do this is to click the Insert Map button on the Insert Tools toolbar.

The Map Expert dialog box appears.



2 On the **Data** tab, in the Layout area, click **Advanced**, if it is not already selected.

3 In the Placement area, in the **Place map** list, specify how often your map appears on the report, then click **Header** or **Footer** to specify where to place your map.

4 In the Data area, add the database field you want to base your map on to the **Geographic field**.

The arrow buttons on the Map Expert dialog box allow you to move fields from one list to the other. Single arrows move only the selected field; double arrows move all fields at the same time.

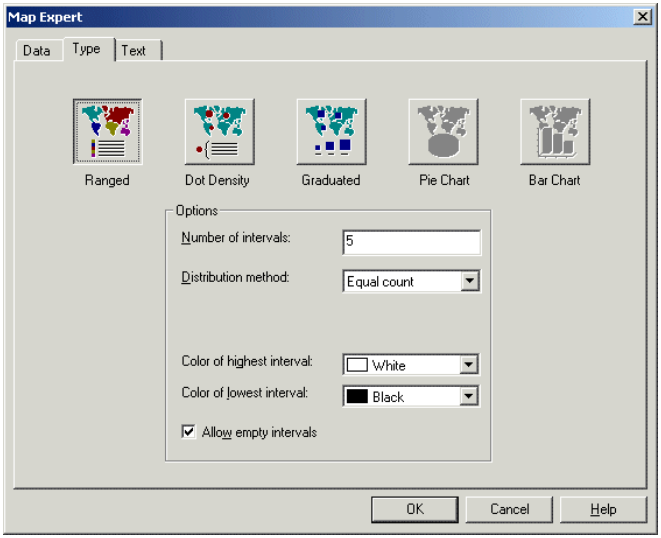
Note: You must map on a geographic field, such as City, Region, or Country; otherwise, Crystal Reports cannot generate your map.

5 Add the database field you want to appear with each change to the **On change of** field.

6 Add the database fields you want to use as values to the **Map values** list.

7 If you do not want Crystal Reports to automatically summarize the map values for a formula field, select the **Don't summarize values** check box.

8 Click the **Type** tab.



9 Click the map type that best illustrates your data (Ranged, Dot Density, Graduated, Pie Chart, or Bar Chart). See "Map types" on page 235.

10 In the Options area, apply formatting options to your map.

11 Click the **Text** tab.

12 In the **Map title** field, enter a title for your map.

13 In the Legend area, you can click one of the following options:

- **Full legend** to show a detailed legend on your map.
- **Compact legend** to show a condensed legend on your map.
- **No legend** to exclude the legend from your map.

14 If you click **Full legend**, click **Made by map** to have Crystal Reports create a legend title based on your map, or click **Specify** to enter your own legend title and subtitle.

15 Click **OK**.

Your map is placed in the Header or Footer section of the report, depending on your selection in Step 3.

Mapping on group fields (Group layout)

To map on a group, you can use the Group layout, in which you show a summary (such as Last Year's Sales) on change of a geographic field (such as Region). In order to create a map using the Group layout, you must have at least one group and at least one summary field for that group.

To map on a group

1 On the **Insert** menu, click **Map**.

Tip: Another way to do this is to click the Insert Map button on the Insert Tools toolbar.

The Map Expert dialog box appears.

2 On the **Data** tab, in the Layout area, click **Group**, if it is not already selected.

3 In the Placement area, in the **Place map** list, specify how often your map appears on the report, then click **Header** or **Footer** to specify where to place your map.

4 In the Data area, in the **On change of** list, click the group field you want to base your map on, then in the **Show** list, click the summary field you want to display on your map.

5 Click the **Type** tab.

6 Click the map type that best illustrates your data (Ranged, Dot Density, or Graduated). See "Map types" on page 235.

7 In the Options area, apply formatting options to your map.

8 Click the **Text** tab.

9 In the **Map title** field, enter a title for your map.

10 In the Legend area, you can click one of the following options:

- **Full legend** to show a detailed legend on your map.
- **Compact legend** to show a condensed legend on your map.
- **No legend** to exclude the legend from your map.

11 If you click **Full legend**, click **Made by map** to have Crystal Reports create a legend title based on your map, or click **Specify** to enter your own legend title and subtitle.

12 Click **OK**.

Your map is placed in the Header or Footer section of the report, depending on your selection in Step 3.

Mapping on Cross-Tab summaries (Cross-Tab layout)

With the Cross-Tab layout, you can map on a Cross-Tab summary field. For example, with a Cross-Tab that shows the total number of items of a product sold in each region of the United States, you may want to include a map that shows the percentage of the total sales of the product provided by each region.

In order to create a Cross-Tab map you must first have a Cross-Tab in your report. For more information about Cross-Tabs, see "Cross-Tab Objects" on page 261.

To map on a Cross-Tab summary

1 On the **Insert** menu, click **Map**.

Tip: Another way to do this is to click the Insert Map button on the Insert Tools toolbar.

The Map Expert dialog box appears.

2 On the **Data** tab, in the Layout area, click **Cross-Tab**, if it is not already selected.

3 In the Placement area, click **Header** or **Footer** to specify where to place your map.

4 In the Data area, in the **Geographic field** list, click a geographic field to supply the geographic areas for your map.

5 If necessary, in the **Subdivided by** list, click another field to map on.

Crystal Reports uses this field to subdivide the pie or bar charts appearing on your map.

6 In the **Map on** list, click a summary field to supply the numeric data for your map.

7 Click the **Type** tab.

8 Click the map type that best illustrates your data. See "Map types" on page 235.

- If you plan to map on only one field, be sure to select None from the "Subdivided by" list, then choose either the Ranged, Dot Density, or Graduated map type.
- If you plan to map on two fields, be sure to select an additional field from the "Subdivided by" list, then choose either the Pie Chart or Bar Chart map type.

9 In the Options area, apply formatting options to your map.

10 Click the **Text** tab.

11 In the **Map title** field, enter a title for your map.

12 In the Legend area, you can click one of the following options:

- **Full legend** to show a detailed legend on your map.
- **Compact legend** to show a condensed legend on your map.
- **No legend** to exclude the legend from your map.

13 If you click **Full legend**, click **Made by map** to have Crystal Reports create a legend title based on your map, or click **Specify** to enter your own legend title and subtitle.

14 Click **OK**.

Your map is placed in the Header or Footer section of the report, depending on your selection in Step 4.

Mapping on an OLAP cube (OLAP layout)

Using the OLAP layout, you can map on an OLAP grid. In order to create an OLAP map, you must first have an OLAP grid in your report. For more information about OLAP grids, see "Creating an OLAP report" on page 279.

To map on an OLAP cube

1 On the **Insert** menu, click **Map**.

Tip: Another way to do this is to click the Insert Map button on the Insert Tools toolbar.

The Map Expert dialog box appears.

2 On the **Data** tab, in the Layout area, click **OLAP**, if it is not already selected.

3 In the Placement area, click **Header** or **Footer** to specify where to place your map.

4 In the Data area, click the **On change of** list and select the field on which you want to base your map.

Note: This field should contain geographic information, such as Country.

5 If necessary, click the **Subdivided by** list and select a secondary row or column to incorporate into your map.

Note: Choosing to subdivide by a secondary mapping field, either row or column, will affect the map types available in Step 8.

6 Click the **Type** tab.

7 Click the map type that best illustrates your data. If there is no secondary mapping field, you can choose **Ranged**, **Dot Density**, or **Graduated**; if there *is* a secondary mapping field, you can choose **Pie Chart** or **Bar Chart**. See "Map types" on page 235.

8 In the Options area, apply formatting options to your map.

9 Click the **Text** tab.

10 In the **Map title** field, enter a title for your map.

11 In the Legend area, you can click one of the following options:

- **Full legend** to show a detailed legend on your map.
- **Compact legend** to show a condensed legend on your map.
- **No legend** to exclude the legend from your map.

Note: If you click Full legend, click "Made by map" to have Crystal Reports create a legend title based on your map, or click Specify to enter your own legend title and subtitle.

12 Click **OK**.

Your map is placed in the Header or Footer section of the report, depending on your selection in Step 4.

Working with maps

Once you have created a map, you may want to add a new title, headings, and a legend, and you may want to change fonts, or even the map type. Crystal Reports provides many options for working with your existing maps; learn more by choosing a topic from the following list:

Editing maps using the Map Expert

Editing maps with the Map Expert allows you to return to the expert in which you designed your map. You can modify many of your original choices, such as the type of map to display, the data on which the map is based, and so on. From either the Design or Preview tab, you can open the Map Expert and make your modifications.

To edit a map using the Map Expert

1. Right-click the map to bring up the shortcut menu.
2. On the shortcut menu, click **Map Expert**.
3. In the Map Expert dialog box, make the desired changes.
4. Click **OK**.

Changing the map title

1. On the **Preview** tab, right-click the map to bring up the shortcut menu.
2. On the shortcut menu, choose **Title**.
3. In the Change Map Title dialog box, enter a new title.
4. Click **OK** to save your changes.

   Crystal Reports returns you to the Preview tab and implements your changes.

Changing the map type

You can change the map type and set the properties for that map directly from the menu that appears when you right-click a map on the Preview tab. For example, if you want to see how the data of a Ranged map would look if it were presented in a Dot Density style, you can rearrange the map without having to return to the Map Expert and rework the format.

To change the map type

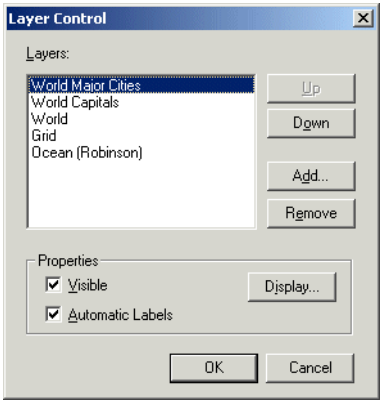
1. On the **Preview** tab, right-click the map to bring up the shortcut menu.
2. On the shortcut menu, choose **Type**.
   The Customize Map dialog box appears.
3. In the **Map type** list, click the new map type.
4. In the Options area, apply formatting options to your new map.
   The properties made available to you vary depending on the map you have selected.
5. Click **OK** to save your changes.
   Crystal Reports returns you to the Preview tab and implements your changes.

Changing map layers

It takes only a moment to rearrange map layers. You can also set properties, such as visibility for those layers, and you can add and remove layers as needed.

To change map layers

1. On the **Preview** tab, right-click the map to bring up the shortcut menu.
2. On the shortcut menu, and choose **Layers**.
   The Layer Control dialog box appears.



3 In the **Layers** list, click **Up** or **Down** to arrange the map layers.

4 In the Properties area, set the properties for each map layer, specifying whether the layer is visible and whether it is automatically labeled.

5 If necessary, click **Display** to open the Display Properties dialog box.

Use this dialog box to set the default display mode and zoom range (the minimum and maximum possible magnification) for the layer in question, then click OK to return to the Layer Control dialog box.

6 Click **OK** to save your changes.

Crystal Reports returns you to the Preview tab and implements your changes.

Resolving data mismatches

Sometimes the map you are working with uses a different geographic name from the one used in the database. For example, a map of Europe may use "United Kingdom" while the database uses the abbreviation "U.K." Until you resolve the data mismatch, the map will not be able to display any information for the selected geographic region.

To resolve data mismatches

1 On the **Preview** tab, right-click the map to bring up the shortcut menu.

2 On the shortcut menu, and choose **Resolve Mismatch**.

The Resolve Map Mismatch dialog box appears.

3 Click the Resolve Mismatch tab to see a listing the geographic names that do not correspond between the database and map.

4 In the **Assign this Field Name** list, click the name you want to resolve.

5 To assign the heading, click a keyword from the **To this Map Name** list.

6 If there is a match, click **Match**.

The new assignment is displayed in the Matched results part of the Resolve Mismatch tab.

7 Once you are finished resolving the data mismatches, click **OK**.

The map can now display data for the selected geographic region.

Changing the geographic map

If you prefer to have your data values presented with a different geographic map, specify your changes using the Change Map dialog box.

To change the geographic map

1. On the **Preview** tab, right-click the map to bring up the shortcut menu.
2. On the shortcut menu, click **Resolve Mismatch**.

   The Resolve Map Mismatch dialog box appears, displaying the name of the current map you are using, and a list of replacement maps that you can choose from.
3. In the **Available maps** list, click the new map you want to use.
4. Click **OK** to save your changes.

   Crystal Reports returns you to the Preview tab and implements your changes.

Zooming in and out on a map

1. On the **Preview** tab, right-click a map to bring up the shortcut menu.
2. On the shortcut menu, click **Zoom In**.
3. Drag the selection cursor around the map section you want to see.

   The map zooms in one level of magnification.
4. Drag another selection to zoom in further.
5. To zoom out on a map, right-click the map to bring up the shortcut menu.
6. On the shortcut menu, click **Zoom Out**.
7. Click the map.

   The map zooms out a level of magnification.
8. Click the map again to zoom out further.

Note:

- If a map occurs once for each instance of a group, any panning or zooming settings you specify are instance-specific. In other words, if you zoom in on the map in one group header, this setting is only for that group header. None of the headers have been changed. This way you can set each map to different settings.
- If the Save Data with Report option is selected from the File menu, your panning and zooming settings will be saved with the report. If the option is not selected, then only the default map settings will be saved.

Panning a map

1. On the **Preview** tab, right-click a map to bring up the shortcut menu.
2. On the shortcut menu, click **Pan**.
3. Drag the panning cursor to the desired map section.

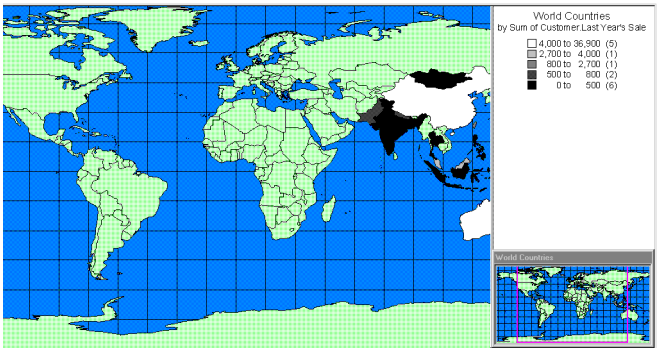
Centering a map

1 On the **Preview** tab, right-click a map to bring up the shortcut menu.

2 On the shortcut menu, click **Center Map**.

Hiding and showing the Map Navigator

The Map Navigator provides a small-scale version of the currently displayed map, so that you can select areas you'd like to pan. By panning, you can quickly drag to change the focus of your map as you require. You can then return your map to the center of the display area by using the Center Map command.



You have the option of hiding or showing the Map Navigator.

To hide the Map Navigator

1 On the **Preview** tab, right-click the map to bring up the shortcut menu.

2 On the shortcut menu, click **Map Navigator**.

The Map Navigator is removed from the Preview tab.

To show the Map Navigator

1 On the **Preview** tab, right-click the map to bring up the shortcut menu.

2 On the shortcut menu, click **Map Navigator**.

The Map Navigator is displayed on the Preview tab.

Formatting Maps

Changing the border of a map

1. On the **Design** or **Preview** tab, right-click the map to bring up the shortcut menu.
2. On the shortcut menu, click **Format Map**.
3. On the Format Editor dialog box, click the **Border** tab to see its options.
4. Change the line style, color, background color, and add or remove a drop shadow from the map border.
5. Click **OK** to save your changes.

   Crystal Reports returns you to the report and implements your changes.

Using the underlay feature with maps

Since maps can only print in certain areas and sections of your report, the underlay feature makes mapping even more useful. Instead of having a map print before the data it represents, you can have the map appear alongside the data, thus making the report easier to read and easier to understand.

To underlay a map

1. Create your map and place it in the Report Header section.

   For more information, see "Creating maps" on page 238.
2. On the **Report** menu, click **Section Expert**.

   Tip: Another way to do this is to click the Section Expert button on the Expert Tools toolbar.

   The Section Expert dialog box appears.
3. In the Sections area, click **Report Header**, then select the **Underlay Following Sections** check box.
4. Click **OK**.

   Crystal Reports returns you to the report. Your map will now underlay the sections below it.
5. If necessary, move or resize the map.

OLE 15

This chapter explains how Object Linking and Embedding (OLE) can be used to edit graphics or other objects from within your report instead of opening an additional application.

OLE overview

Object Linking and Embedding (OLE) enables you to insert objects (OLE objects) into a report from other applications (OLE server applications) and then use those applications from within Crystal Reports to edit the objects if necessary.

If you were not using OLE, you would have to exit Crystal Reports, open the original application, change the object, return to Crystal Reports, delete the object originally inserted, and then insert the newly revised object.

OLE terminology

Before you work with OLE objects in Crystal Reports, it might be helpful to review some terminology:

- **OLE**
  OLE is an acronym for Object Linking and Embedding. It refers to the ability to create compound documents; that is, documents that contain elements from other applications that can be edited using the original application.
- **OLE Object**
  An OLE object is, broadly speaking, a presentation of data that was created in another application and that maintains a relationship with the application that was used to create it. A bitmap created in Paint, an Excel spreadsheet, or a graph from MS Graph may all be OLE objects if they are inserted in the receiving document as OLE objects. If they are not inserted as OLE objects, they retain no relationship with the original application.
- **OLE Container Application**
  An OLE container application is one that can contain and process OLE objects created in other applications (such as Paint or Excel). Crystal Reports is a container application.
- **Container Document**
  A container document is a document that was created using the container application and that contains one or more OLE objects.
- **OLE Server Application**
  An OLE server application is an application that allows its documents to be inserted into an OLE container document as OLE objects. Microsoft Word and Excel are examples of applications that are both OLE servers and OLE containers. That is, they can both create new OLE objects and they can contain OLE objects created elsewhere.
- **Server Document**
  The server document is a file created in the server application that stores the original OLE object.

Types of OLE objects

- **Static object**
  A static OLE object is a picture of an object that can be displayed and printed, but not edited in place. It does not have a connection to a server application. There are two kinds of static objects: static bitmaps and static metafiles.
- **Embedded object**
  An embedded object contains a representation of the object, as well as information that defines the content. It does not have a connection to the original object in the server document. As a general rule, use embedded objects when you need to edit the object independently of the original server document.
- **Linked object**
  A linked object contains a representation of the data and a link to the file where it originated. As a general rule, use linked objects when the data in the server document is changing and you want the object in your report to be updated when you open the report.

General OLE considerations

There are several points to keep in mind when utilizing OLE functionality.

- When you double-click an embedded OLE object, the object is activated for editing and the Report Designer merges its menus and toolbars with those of the object's server application. If the OLE server application does not support this behavior, the object appears in a separate window. When you are finished editing, click outside the object and the Report Designer toolbars reappear.
- When you double-click a linked OLE object, the program opens the object's server application with the object displayed and ready for editing. You cannot edit a linked object in place in Crystal Reports because you are working on the original object. Since the object could be linked to multiple documents, displaying the original in the server application limits access to one editor at a time.

Inserting OLE objects into reports

There are several ways to insert OLE objects into an application.

- The OLE Object choice on the Insert menu can be used to import an existing object or create a new one. You can place either embedded or linked objects using this method.
- Paste Special on the Edit menu can be used to copy or cut the object from an OLE server application and paste it in a report. If the object can be pasted in multiple formats, you need to decide which format to use. For example, when inserting text from a Microsoft Word document, you can paste the text as Microsoft Word document text (which can be edited in Word), or as a metafile, which is simply a non-editable picture of the text. Use Paste Special to place either embedded or linked objects.

- Finally, you can drag and drop an object from an OLE server application. You do this by opening Crystal Reports in one window and the OLE server application in another and then dragging the object between the two. However, when an object is dragged into a report, the object is embedded, not linked.

Note: When inserting selected information (portions of larger files), it is best to use either Copy and Paste Special, or drag and drop editing. When inserting entire files, click OLE Object from the Insert menu.

To drag and drop OLE objects

This procedure assumes that you have Microsoft Excel or another spreadsheet program on your computer that is an OLE server application.

1. Open both Crystal Reports and Microsoft Excel and size the program windows so that they are both visible.

   You will need to be able to see both the object's origin and its destination.

2. Open an existing report in Crystal Reports.
3. Create a spreadsheet in Microsoft Excel by typing the numbers one through ten (1-10) in cells A1 through A10.
4. Select cells A1 through A10.
5. Move the cursor to the left edge of the highlighted cells until an arrow pointer appears.
6. Press the left mouse button and hold it down.

   This activates the cells to be moved. Since you're only going to copy the cells, press the Ctrl key and hold it down too.

7. Drag the cells into the Details section of the report in Crystal Reports, and when you have the cells positioned, release the mouse button then the Ctrl key.
8. Right-click the spreadsheet object in the report and observe that it is identified as an OLE object and that the second command from the last identifies it as a Worksheet Object.
9. Double-click the spreadsheet object.

   The menus and toolbars change to a combination of those from the OLE server application and Crystal Reports. You can now edit the object in-place. The object is an embedded object. Any changes you make to the object will not affect the original.

10. Click outside the object when you are finished.

    The menus and toolbars change back to those of Crystal Reports. Any changes you made while editing are saved.

How OLE objects are represented in a report

An OLE object is displayed in a report in any of several different formats depending on how you chose to insert the object.

- If you have chosen to display the object as an icon, the icon will appear in the report. Users can choose to edit the object later by double-clicking the icon.
- If the object was created from an existing file, the data from that file (or an icon) is displayed in the report. This data can be edited by double-clicking the object or its icon.
- If you are creating a new object, the application for the object type that you chose will open, and you can begin designing the object. When you are finished, close or exit the application. The object (or its icon) will be displayed in the report.

Editing OLE objects in reports

In-place editing

In-place editing is the ability to change an OLE object's properties from within an OLE container application (such as Crystal Reports). The container application's menu items change to provide the editing tools from the server application so that you can make changes easily.

When an OLE object is placed in a report, the object becomes part of the report. To edit the object, double-click it and then modify it using the editing tools found in the object's original application or from a similar application that allows in-place editing. You can edit an OLE object only if you have the corresponding server application installed on your computer. You can view and print a report containing an OLE object even if you don't have the server application installed.

Dynamic OLE menu commands

When you have an OLE object selected, there is a submenu on the Edit menu that provides commands for the object. The name of this submenu reflects the type of the OLE object. The commands on the submenu also vary according to the type of object.

The object may be described as a Document object, Bitmap Image object, Picture object, Worksheet object, or something similarly descriptive:

- If the object is embedded, the Edit menu displays those commands that are available to that type of embedded object.
- If the object is linked, the Edit menu displays commands for that type of linked object.

Commands on the context menus change in a similar fashion.

These dynamic commands are provided to give you more control when working with OLE objects.

OLE and the Insert Picture command

When the Object command on the Insert menu is used to place pictures in a report, the picture will be converted to an embedded or linked object associated with Microsoft Paint (or another image editing application).

When the Picture command on the Insert menu is used to place pictures in a report, the Report Designer converts them into static objects.

Static objects cannot be edited. To edit a static object, it has to be converted into an editable type of object using the Convert command on the Edit menu.

Working with static OLE objects

To insert a static OLE object

1. Create a report using the sample data from **Xtreme.mdb**. Place any field in the **Details** section of the report.
2. On the **Insert** menu, click **Picture**.

   Tip: Another way to do this is to click the picture button on the Insert Tools toolbar.

   The Open dialog box appears.

   Note: Crystal Reports supports Windows Bitmap, TIFF, JPEG, and PNG files, as well as Windows Metafiles.
3. Select **Xtreme.bmp** from the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory.

   Note: The location of the bitmap may vary depending on the destination specified during installation.
4. Click **Open**.
5. When the object frame appears, place the picture on the far left side of the **Report Header** section.
6. Right-click the picture, and notice that the picture is identified as an OLE object on the shortcut menu.
7. When you double-click the object, nothing happens. A static OLE object cannot be edited.

To convert a static OLE object to a bitmap image object

1. Right-click the static OLE object you placed, and choose **Convert Picture Object** from the shortcut menu.

   The Convert dialog box appears.
2. Select **Paintbrush Picture** in the **Object Type** list, and click **OK**.

   You have just converted a static OLE object to a modifiable OLE object.

Note: This option does not appear if the original file was a metafile.

3 Right-click the object you have just converted.

The second command from the bottom of the shortcut menu is now Bitmap Image Object.

4 Double-click the object.

Microsoft Paint, or another graphics application installed on your machine opens.

Working with embedded vs. linked objects

Since embedded and linked objects each have different properties, it is important for you to consider the capabilities of each when deciding which OLE format to use.

Embedded objects

An embedded object can be created from within Crystal Reports or by using a file that already exists. If you use an existing file, the object is copied to the report.

When changes are made to an embedded object it does not affect the original file and vice versa. The changes are made only to the object within your report.

To insert an embedded OLE object

1 Click the **Design** tab.

2 On the **Insert** menu, click **OLE Object**.

The Insert Object dialog box appears.

3 Select **Create From File**.

The dialog box changes, allowing you to either type in an object name or browse.

4 Click **Browse** and choose **Xtreme.bmp** from the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory.

Note: The location of the bitmap may vary depending on the destination specified during installation.

5 Click **Open** to return to the Insert Object dialog box.

6 Click **OK** to return to the report.

An object frame appears.

7 Place the object in the Report Header section, just to the right of the first picture.

8 Right-click this picture and observe that it is also identified as an OLE Object on the shortcut menu.

The second command from the bottom of the shortcut menu identifies it as a Bitmap Image Object. It is an embedded OLE object.

9 Double-click the object.

Note: You have remained in Crystal Reports but the menus and tools have merged with Microsoft Paint, or those of another graphics package installed on your machine. These tools can be used to edit the bitmap in place. Remember that even though you are working with the tools of the OLE server application, you are not working on the original file; you are working with a copy and nothing you do here will affect the original.

Linked objects

When a linked object is inserted into a report, Windows copies a snapshot of the data (not the data itself) from a file that already exists. The image of the object is added to your report along with a reference to the data used to create it. The actual data remains with the original file.

When the object is activated from within a report, the original file is opened inside the application that was used to create it. Any changes you make directly affect the original file also.

If you want the data in an object to remain available to other applications, and always reflect the most current changes to the data, link the object to the report. An automatic link is refreshed from the source file every time the report is loaded; a manual link is refreshed only when you request it. You can force a refresh, break a link, or redefine the link in the Edit Links dialog box.

When you have a linked object and you break the link using the Links dialog box, all connections to the original data in the server document are broken. A linked object in a container application is merely a representation of that object and a link between the object and the server document. When you break the link you're left with only the representation, with no relationship to the original data or to the editing capabilities of the server application. In this situation, Crystal Reports turns the object into a static object, a stand-alone object that can neither be edited using OLE capabilities nor converted into an editable OLE object.

To link a bitmap image object

1 On the **Insert** menu, click **OLE Object**.

The Insert Object dialog box appears.

2 Select **Create From File**.

The dialog box changes, enabling you to enter an object name or browse.

3 Click **Browse** and choose **Xtreme.bmp** from the `\Program Files\Crystal Decisions\Crystal Reports 9\Samples\En\Databases` directory.

Note: The location of the bitmap may vary depending on the destination specified during installation.

4 Click **Open** to return to the Insert Object dialog box.

5 Select the **Link** check box.

6 Click **OK** to return to the report.

An object frame appears.

7 Place this object just to the right of the embedded object.

8 Right-click this object and observe that this object is identified as an OLE object.

The second command from the bottom of the shortcut menu identifies this as a Linked Bitmap Image Object.

9 Double-click the object.

Microsoft Paint or the graphics application you are using opens, displaying the original file. Any changes you make to the original will be reflected in the object that appears in your report.

10 Close Microsoft Paint or the graphics application that you are using and return to your report.

Cross-Tab Objects 16

A Cross-Tab object is a grid that displays values that are grouped and summarized in two directions. This chapter provides you with information about how you can use Cross-Tab objects in your report.

What is a Cross-Tab object?

A Cross-Tab object is a grid that returns values based on the criteria you specify. Data is presented in compact rows and columns. This format makes it easy to compare data and identify trends. It is made up of three elements:

- Rows
- Columns
- Summary fields

Column

Row

	USA	MEXICO	CANADA	UK	TOTAL
Gloves	4	0	4	0	8
Belts	0	1	1	1	3
Shoes	0	0	0	1	1
TOTAL	4	1	5	2	12

Sum of gloves in USA (intersection)

- The rows in a Cross-Tab run horizontally (from side to side). In the example above, “Gloves” is a row.
- The columns in a Cross-Tab run vertically (up and down). In the example above, “USA” is a column.
- The summary fields are found at the intersection of rows and columns. The value found at each intersection represents a summary (sum, count, and so on) of those records that meet the row and the column criteria. In the example above, the value at the intersection of “Gloves” and “USA” is four, the number of gloves sold in the USA.

A Cross-Tab also includes several totals:

	USA	MEXICO	CANADA	UK	TOTAL
Gloves	4	0	4	0	8
Belts	0	1	1	1	3
Shoes	0	0	0	1	1
TOTAL	4	1	5	2	12

Total of products in row (gloves)

Grand total all products (row) all countries (column)

Total of products in column (USA)

- At the end of each row is a total for that row. In the example above, this total represents a single product sold in all countries. At the end of the “Gloves” row is the value 8, the total number of gloves sold in all countries.

  Note: The total column can appear at the beginning of each row.
- At the bottom of each column is a total for that column. In the example above, this total represents all products sold in a single country. The value at the bottom of the “USA” column is four, the total number of products (gloves, belts and shoes) sold in the USA.

  Note: The total column can appear at the top of each column.

- At the intersection of the Totals column (totals for the products) and the Totals row (totals for the countries) is a grand total. In the example above, the value at the intersection of the Total Column and Total Row is 12, the total number of all products sold in all countries.

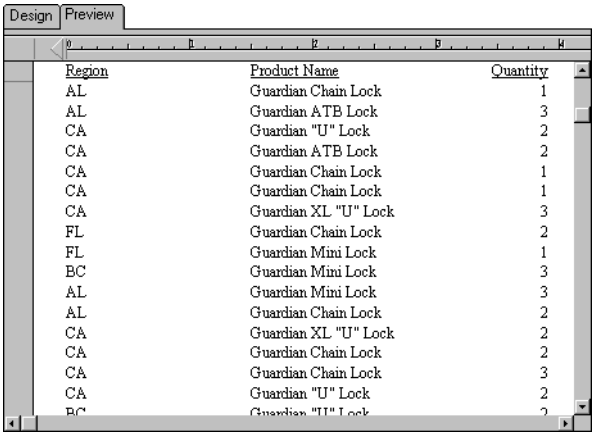
Cross-Tab example

This example demonstrates how you can use a Cross-Tab object to make it easier to understand complex data. The goal of this report is to analyze the unit sales of five different bicycle locks in four different regions (a unit sales of locks *by* region report). For greater clarity, only the most essential information in these reports has been included:

- The region from which the order came.
- The name of the lock.
- The quantity ordered.

The first way of looking at the data is in the most basic of all reports, a columnar report with no grouping or sorting.

Report of order data—no sorting/grouping



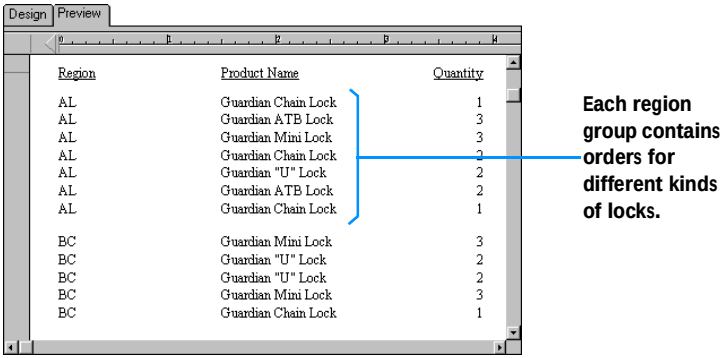
Region	Product Name	Quantity
AL	Guardian Chain Lock	1
AL	Guardian ATB Lock	3
CA	Guardian "U" Lock	2
CA	Guardian ATB Lock	2
CA	Guardian Chain Lock	1
CA	Guardian Chain Lock	1
CA	Guardian XL "U" Lock	3
FL	Guardian Chain Lock	2
FL	Guardian Mini Lock	1
BC	Guardian Mini Lock	3
AL	Guardian Mini Lock	3
AL	Guardian Chain Lock	2
CA	Guardian XL "U" Lock	2
CA	Guardian Chain Lock	2
CA	Guardian Chain Lock	3
CA	Guardian "U" Lock	2
BC	Guardian "U" Lock	2

This report presents details. Each row represents an individual order. There are many orders from each of the regions for different locks. But because there is no summary information, it is nearly impossible to get any useful information out of a report like this.

The next logical step is to group the data in some way. You can group it by region, or by product line. The following section shows a look at both of these options.

Report of order data—grouped by region

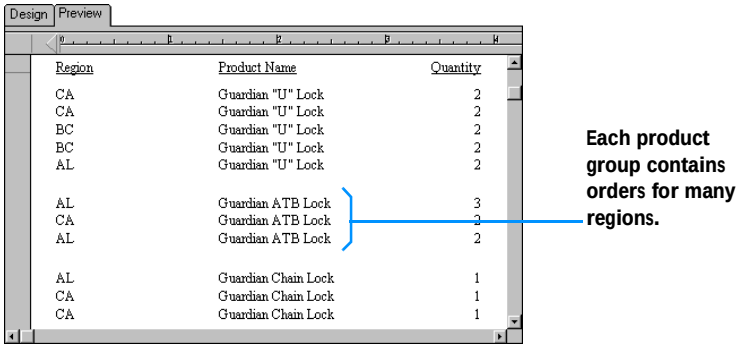
This report uses the data seen in the first report, but here the data is grouped by region. All the orders in each region are grouped together, but each regional group contains orders for different types of locks. Because the groups contain different kinds of data, summarizing the Quantity field will determine the total number of locks sold per Region, but not the total of each type.



Report of order data—grouped by product

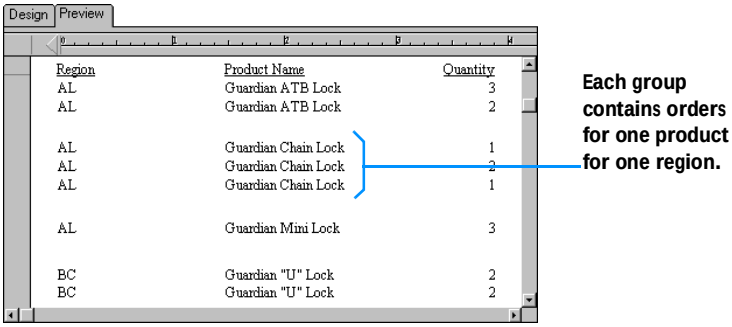
This report groups the data by product. Each group displays all the orders for a specific product. At first it appears that this might be useful, but then it becomes clear that each product group includes orders from several different regions.

The information is helpful, and it brings you closer to your goal, but you are still a long way from having the information you need.



Report of order data—grouped by region and product

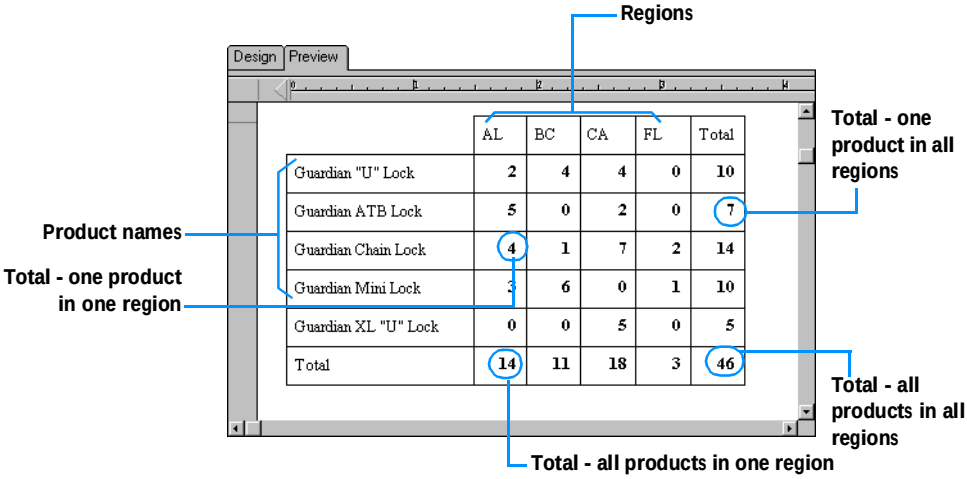
This report is the logical next step. If the *By Region* report contains multiple products in each region group, and the *By Product* report contains multiple regions in each product group, then it seems to make sense to combine the two. Doing that, you group first by Region and then by Product.



But the data is all spread out and remains difficult to analyze. This information is useful, and with a little work you can use a report like this to get the comparison information you need. However, a Cross-Tab offers a better solution.

Order data in a Cross-Tab object

With Cross-Tabs, all the information you need is provided in a compact format. The report shows the products sold in each region and what the unit sales were. It is easy to see, for example, that Guardian Mini Locks are not popular at all in California but they are the biggest seller in BC or that Florida is being outsold by Alabama in every lock category.



	AL	BC	CA	FL	Total
Guardian "U" Lock	2	4	4	0	10
Guardian ATB Lock	5	0	2	0	7
Guardian Chain Lock	4	1	7	2	14
Guardian Mini Lock	3	6	0	1	10
Guardian XL "U" Lock	0	0	5	0	5
Total	14	11	18	3	46

In this Cross-Tab:

- Product names make up the row headings.
- Regions make up the column headings.
- The value at each row / column intersection is the sum of all the orders for a particular product for a particular region; for example, the total number of Guardian Mini Locks purchased in British Columbia.
- The total at the end of each row is the total of all of the purchases for one product in all regions; for example, the total number of Guardian ATB Locks purchased in Alabama, British Columbia, California, and Florida combined.
- The total at the bottom of each column is the total number of all kinds of locks ordered in one region; for example, the number of locks of all kinds purchased in California.
- The total in the bottom right corner is the grand total showing the total unit sales of all five locks in all four regions.

The report is compact, and enables you to compare your customers' purchasing habits quickly.

Creating a Cross-Tab report

This section provides you with the steps to create a Cross-Tab object in a new report and how to add a Cross-Tab object to an existing report.

Things to keep in mind when using Cross-Tab objects:

- You can have multiple rows, columns, and summarized fields.
- The report fields you can use as rows, columns, or summarized fields include print-time formulas and running total fields.
- You can include a group sort (top or bottom N) on the rows in your Cross-Tab.
  Note: Group sorting cannot be applied to columns.
- You can insert as many Cross-Tab objects in a report as you need.
- You can insert the Cross-Tab into either the Report Header or Footer, or the Group Header or Footer.
- You can place Cross-Tab objects in subreports. This is useful when you want to refer to the results from another report.

To create a new cross-tab report

1. On the **File** menu, click **New**.

   Tip: Another way to do this is to click the New button on the Standard toolbar.

   The Crystal Reports Gallery appears.
2. In the **Choose a Wizard** area click **Cross-Tab**.
3. Click **OK**.

   The Cross-Tab Report Creation Wizard appears.

Specifying the data source

1 On the Data screen, locate the data source you want to use.

This example uses the Xtreme Sample Database.

For information on how to select a data source, see "Selecting a database to use" on page 21.

2 Select the tables that contain the fields to include in the report.

Tip: Use the Ctrl-click combination to pick a non-continuous range of fields and the Shift-click combination to pick a continuous list of fields.

This example uses Customer, Orders, Orders Detail and Product.

3 Click the > arrow to add the selected tables.

4 Click **Next**.

The Link screen appears.

Modifying the links

1 In this example, check to make sure that the Customer table is linked to the Product table via the Orders and Orders Detail tables.

If you do not like the linking that Crystal Reports has automatically suggested, change it by clicking Clear Links, and then select the fields that you would like to link together.

2 Click **Next**.

The Cross-Tab screen appears.

Defining the structure of the Cross-Tab

Add fields from the Available Fields area to the Rows, Columns, and Summary Fields areas.

1 From the **Customer** table, select **Region**; then click the > arrow next to the Rows area.

The Region field is added to the Rows area.

Tip: You can also add a field to the different Cross-Tab areas by selecting it and dragging it to the Columns, Rows, or Summary Fields areas.

2 From the **Product** table, select **Product Class**; then click the > arrow next to the Columns area.

The Product Class field is added to the Columns area.

3 From the **Orders** table, select **Order Amount**; then click the > arrow next to the Summary Fields area.

The Orders Amount is added to the Summary Fields area.

4 Select the summary operation you want to perform on **Order Amount** from the list beneath the Summary Fields area.

5 Click **Next**.

The Chart screen appears.

Adding a chart

1 Select the kind of chart you want to see in your report:

- Bar Chart
- Line Chart
- Pie Chart

Note: If you don't want to add a chart, leave No Chart selected and skip to "Selecting fields" on page 268.

2 Change the default charting information to suit your report.

For more information about creating a group chart, see "Charting on summary or subtotal fields (Group layout)" on page 224.

3 Click **Next**.

The Record Selection screen appears.

Selecting fields

Use this screen to apply selection conditions. This limits the results that are returned in your report to only those records that you are interested in.

1 Select the field to apply selection to.

For this example choose Region.

2 Click the > arrow.

The field is added to the Filter Fields area and the filter types list appears below it.

3 From the filter types list choose a filter method.

For this example, choose "is one of" as the filter method.

4 In the filter values list that appears, select the value(s) you want to filter on and click **Add**.

For this example, choose AZ, CA, and OR from the drop-down list.

The report will provide you with a breakdown and summary of accessory and bicycle sales for Arizona, California, and Oregon.

5 Click **Next**.

The Grid Style screen appears.

Applying a predefined style and finishing the report

1 Select a style.

For this example choose Original.

2 Click **Finish**.

The Cross-Tab report appears in the Preview tab.

3 If you do not want the Cross-Tab to appear in the **Report Header**, click the **Design** tab, and move the Cross-Tab to the **Report Footer**, or **Group Header** or **Footer**.

Note: If you place the Cross-Tab in the Report Footer, unsuppress the footer by right-clicking in the Report Footer's grey area and selecting Don't Suppress.

4 Click the **Preview** tab to view the report.

To add a Cross-Tab to an existing report

In this example, a Cross-Tab is added to an existing report to show the sales in North America compared to the rest of the world.

1 Open your report.

This example uses the Group By Intervals.rpt included in the Feature Examples directory.

2 Click the **Design** tab.

3 On the **Insert** menu, click **Cross-Tab**.

Tip: Another way to do this is to click the Insert Cross-Tab button on the Insert Tools toolbar.

The Cross-Tab Expert appears. This dialog box is made up of three tabs: Cross-Tab, Style, and Customize Style.

Defining the structure of the Cross-Tab

Add fields to the Rows, Columns, and Summarized Fields areas.

1 From the **Available Fields** list, select **Country**; then click the > arrow next to the Rows area.

The Country field is added to the Rows area.

Tip: You can also add a field to the different Cross-Tab areas by selecting it and dragging it to the Columns, Rows, or Summarized Fields areas.

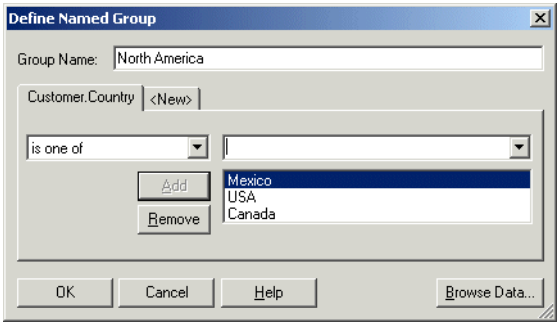
2 From the **Available Fields** list, select **Last Year's Sales**; then click the > arrow next to the Summarized Fields area.

The Last Year's Sales field is added to the Summarized Fields area.

3 Apply additional structure:

- Click **Country** in the Rows area.
- Click **Group Options**.
- Select **in specified order** from the drop-down list.
- Type **North America** for the name of the Named Group.
- Click **New**.
- Choose **is one of** from the drop-down list.

- Select **Canada**, **Mexico**, and **USA** from the drop-down list.



- Click **OK** to close the Define Named Group dialog box.
- Click **OK** to close the Cross-Tab Group Options dialog box.

Choosing a predefined style

1. Click the **Style** tab.
2. Select a style.

Applying a custom style

1. Click the **Customize Style** tab.
2. Customize the style as desired.

Finishing the Cross-Tab

1. Click **OK**.
2. Place the Cross-Tab in either the Report Header or Report Footer.
3. On the **Report** menu, click **Refresh Report Data**.
   The updated report appears.

Working with Cross-Tabs

This section describes ways you can work with a Cross-Tab once you've added it to your report:

- Showing values as percentages
- Abbreviating large summarized fields
- Customizing row/column labels
- Using running totals in Cross-Tabs
- Printing Cross-Tabs that span multiple pages

Showing values as percentages

1. Right-click the blank top-left area of a Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.

2. On the **Cross-Tab** tab, select a summarized field; then click **Change Summary**.
3. In the Edit Summary dialog box, select the **Show as percentage of** option, and choose the summary field you want to base the percentage on.
4. Choose **Row** or **Column** depending on whether you want your percentage values to be calculated horizontally (row) or vertically (column).
5. Click **OK** to close the Edit Summary dialog box.
6. Click **OK** to return to your Cross-Tab.

   Summarized field values are now displayed as percentages on your Cross-Tab.

Abbreviating large summarized fields

Because the values in a Cross-Tab's summarized fields are often very large, Crystal Reports lets you abbreviate such values.

Note: To complete this procedure, the report you're working with must include the custom function called `cdFormatCurrencyUsingScaling`. To learn how to add this custom function, search the online help for the topic "Adding custom functions from a repository."

1. Right-click the summarized field you want to abbreviate and choose **Format Field** from the shortcut menu.

   The Format Editor appears.

2. On the Common tab, click the conditional formatting button adjacent to **Display String**.
3. In the Functions tree of the Format Formula Editor, select `cdFormatCurrencyUsingScaling` from the Custom Functions folder.
4. Complete the custom function's arguments as follows:

   `cdFormatCurrencyUsingScaling (CurrentFieldValue,1,"K","M")`

   - As the first argument, choose `CurrentFieldValue` from the Formatting Functions folder of the Functions tree. This argument defines the actual value to be summarized.
   - As the second argument, enter the number of decimal places you want to display on your Cross-Tab.
   - As the third and fourth arguments, enter the strings (surrounded by quotation marks) that you want to display as thousands and millions symbols.

5. Click **Check** to identify any errors in the formula.

6 Fix any syntax errors the Formula Checker identifies.

7 When the formula has the correct syntax, click **Save and close** on the Formula Workshop toolbar.

   You return to the Format Editor dialog box.

8 Click **OK** to return to your Cross-Tab.

   Your summarized field values appear abbreviated as defined.

Note: You can create your own custom function, or you can copy and modify the provided sample to abbreviate a summarized field. In general, using a custom function is not a requirement of working with the Display String feature. You can write any Display String formula you need as long as its return value is a string.

Customizing row/column labels

By default, row and column labels are derived from the data on which you base your Cross-Tab. You can, however, customize row and column labels.

1 Right-click the row or column whose label you want to customize and choose **Format Field** from the shortcut menu.

   The Format Editor appears.

2 On the Common tab, click the conditional formatting button adjacent to **Display String**.

3 In the Format Formula Editor, create formula text that describes the custom names you want to use as well as the conditions under which these names should be applied.

   For more information about how to do this, see "Working with conditional formatting" on page 208.

4 Click **Check** to identify any errors in the formula.

5 Fix any syntax errors the Formula Checker identifies.

6 When the formula has the correct syntax, click **Save and close** on the Formula Workshop toolbar.

   You return to the Format Editor dialog box.

7 Click **OK** to return to your Cross-Tab.

   Your customized row and column names appear when the conditions you set are matched.

Using running totals in Cross-Tabs

Running Total fields can be useful as summary fields in Cross-Tab objects. Create your Cross-Tab as shown in "Creating a Cross-Tab report" on page 266, but choose a running total field as your Summary Field.

If you want a running total that totals down the Cross-Tab's columns, you must sort the records in the Cross-Tab by Column and then Row.

To total down columns

1. Right-click the blank top-left area of a Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.
2. On the **Cross-Tab** tab of the Cross-Tab Expert, make sure you have added a Running Total field to the **Summarized Fields** area.
3. Click **OK** to return to the Cross-Tab in your report.
4. Click **Record Sort Expert** and sort your Cross-Tab by the field you designated as a Column in the Cross-Tab Expert.
5. Create a second sort on the field you designated as a Row in the Cross-Tab Expert.

If you want a running total that totals across the Cross-Tab's rows, you must sort the records in the Cross-Tab by Row and then Column.

To total across rows

1. Repeat steps 1 to 3 of the previous procedure.
2. Click **Record Sort Expert** and sort your Cross-Tab by the field you designated as a Row in the Cross-Tab Expert.
3. Create a second sort on the field you designated as a Column in the Cross-Tab Expert.

If you add a Group Sort (top or bottom N sort) based on a Running Total field, the sort is carried out on the running total values as shown in the Cross-Tab's Row / Column Total areas—it is not based on the summary field's totals. Also, in this case, the running total results appear correctly only in the total data and not in the cell data.

Printing Cross-Tabs that span multiple pages

When you create a Cross-Tab that is wider or longer than the specified page size, the program automatically spans the printing across enough pages to accommodate the size of the Cross-Tab. In the Preview tab, a line will appear at each page break. For ease in reading, column headings will be repeated on subsequent pages. Row headings can also be repeated using the Keep Columns Together option.

To repeat row labels

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.
2. Click the **Customize Style** tab.
3. Select the **Repeat Row Labels** check box, if desired.

Formatting Cross-Tabs

Crystal Reports has powerful formatting capabilities that can be applied to Cross-Tabs. The following sections describe these key procedures:

- Changing width, height, and alignment of Cross-Tab cells
- Formatting background color of entire rows/columns
- Formatting fields individually
- Formatting several fields at one time
- Suppressing Cross-Tab data
- Displaying summarized fields horizontally

By applying such formatting as background color, borders, and fonts, you can emphasize important data and create professional-looking, easy-to-understand Cross-Tabs. For more information, see "Formatting properties" on page 199.

You can also use the Highlighting Expert to apply conditional formatting to Cross-Tab cells. To access the Highlighting Expert, right-click the cell you want to format and, on the shortcut menu, click Highlighting Expert. For more details, see "Using the Highlighting Expert" on page 212.

Changing width, height, and alignment of Cross-Tab cells

1. Click a cell within the Cross-Tab to activate the sizing handles.
2. Click and drag one of the sizing handles to expand the width or height of the cell.
3. Click an alignment option on the Formatting toolbar to change the alignment of the data in the cell.

   You can choose between left, right, center, and justified alignment.

Note: Changing one cell affects all like cells. For example, changing the width of one summary field cell changes all other summary field cells at the same time.

Formatting background color of entire rows/columns

Use background colors to emphasize rows or columns within you Cross-Tab.

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.
2. Click the **Customize Style** tab.
3. Click the row (in the Rows area) or column (in the Columns area), and select a color from the **Background Color** drop-down list.
4. Click **OK** to return to the Cross-Tab.

   The row/column is formatted as specified.

Formatting fields individually

1. Right-click the field you want to format and choose **Format Field** from the shortcut menu.

   The Format Editor appears.

2. In the Format Editor, make font, background, borders, numbering, currency symbols, and printing characteristics choices.

3. Click **OK** to return to the Cross-Tab.

   The field is formatted as specified.

Formatting several fields at one time

1. Use the Shift-click method to highlight the desired fields.

2. Right-click any one of them and choose **Format Objects** from the shortcut menu.

   The Format Editor appears.

3. In the Format Editor, make font, background, borders, numbering, currency symbols, and printing characteristics choices.

4. Click **OK** to return to the Cross-Tab.

   The fields are formatted as specified.

Suppressing Cross-Tab data

This section provides you with the steps required to suppress data in your report. You can suppress:

- Empty rows and columns.
- Row and column grand totals.
- Subtotals and their labels.

To suppress empty rows and columns

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.

2. Click the **Customize Style** tab.

3. Select either the **Suppress Empty Rows** or **Suppress Empty Columns** check box.

4. Click **OK**.

   Now, when you print the report, empty rows and/or columns will not appear.

To suppress row and column grand totals

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.

2. Click the **Customize Style** tab.
3. Select either the **Suppress Row Grand Totals** or **Suppress Column Grand Totals** check boxes.
4. Click **OK**.

To suppress subtotals and their labels

If you have more than two groups in your Cross-Tab you can suppress the subtotal and label for one of them.

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.

2. Click the **Customize Style** tab.
3. Click the field whose subtotal you want to suppress.

   The Suppress Subtotal and the Suppress Label check boxes become active.

4. In the Group Options area, select the **Suppress Subtotal** check box.
5. Click the **Suppress Label** check box to suppress the label associated with subtotal.
6. Click **OK**.

Displaying summarized fields horizontally

If you have two or more summarized fields in your Cross-Tab, you can display their values horizontally instead of vertically (the default).

1. Right-click the blank top-left area of the Cross-Tab and select **Cross-Tab Expert** from the shortcut menu.

   The Cross-Tab Expert appears.

2. Click the **Customize Style** tab.
3. In the Summarized Fields area, select **Horizontal**.
4. Click the **Show Labels** check box if you want to display labels for the summarized fields.

   Labels are displayed in the direction you chose for the summarized fields.

5. Click **OK**.

Creating and Updating OLAP Reports 17

This chapter describes how to create OLAP reports, how to update them when the location of your data changes, and how to work with the data displayed in an OLAP grid.

OLAP reporting with Crystal Reports

Although relational databases such as SQL servers and PC databases are the most common sources of data, Online Analytical Processing (OLAP) and Multi-Dimensional Data are rapidly becoming the popular data-storage and analysis formats. Crystal Reports provides the same access and reporting features for OLAP data sources that it provides for relational data.

Note: The term OLAP is used in this topic to refer to all common forms of OLAP and Multi-Dimensional Data storage and access systems.

Crystal Reports supports access to OLAP data through direct connection and Open OLAP gateways. A direct connection requires an installed OLAP client and uses DLLs located on your local machine, whereas the Open OLAP gateway doesn't require either. With an Open OLAP connection, data access is achieved through a Name Server host that communicates with Crystal Reports and your OLAP data source. Set up an Open OLAP connection on the Advanced Settings tab of the New Server dialog box.

For direct connections, Crystal Reports supports many OLAP server types. These types are displayed in the Server Type list of the New Server dialog box. This list is dynamic and will show those server types for which you have an installed client.

OLAP Grid Objects

When you design a report with OLAP data, Crystal Reports creates a primary report containing one or more OLAP grid objects. OLAP grid objects look and act much like Cross-Tab objects, but they are designed specifically for OLAP data.

OLAP grid objects provide true multi-dimensional reporting. Add dimensions to either axis to analyze three, four, or more dimensions in a single OLAP grid. Or instead of viewing multiple dimensions within one OLAP grid, create multiple OLAP grids within the same report.

Note: When Crystal Reports displays an OLAP grid, it can do so quickly if the grid is long (down many pages) instead of wide (across many pages). When the grid is long, the program processes it on a page-by-page basis. When the grid is wide, the program must gather all the data before it can display a page. This may require significantly more processing time.

Creating an OLAP report

OLAP reports are created using the OLAP Report Creation Wizard or the OLAP Expert. Before you start creating your report, ensure you have the following information:

- OLAP Type
- Server/Database name
- User ID
- Password
- Parameters (applies only to OLE DB for OLAP)

Tip: You can return to the OLAP Expert by selecting the OLAP Report Setting option on the Report menu. Do this if you want to change your page dimensions, or if you want to change parameter values.

To create an OLAP report

1. From the **File** menu, click **New**.

   Tip: Another way to do this is to click the New button on the Standard toolbar.

   The Crystal Reports Gallery appears.
2. Select **OLAP**.
3. Click **OK**.

   The OLAP Report Creation Wizard appears with the Data screen active.

Specifying the data source

Use the Data screen to define the OLAP data source.

1. Click **Select Cube**.

   The Crystal OLAP Connection Browser appears.
2. Browse your OLAP server for the cube you want to connect to.

   If your server isn't in the list, click Add Server. In the New Server dialog box, provide the server information; then click OK.
3. Select the desired cube and click **Open**.

   The Data screen reappears with the supplied data source information.
4. Click **Next**.

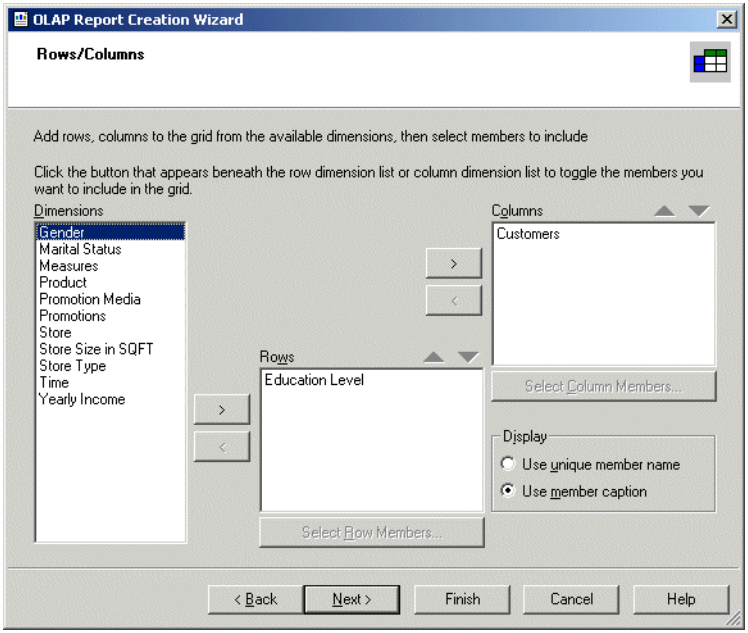
Note: You can also click "Select CAR File" on the Data screen. If you choose this option, the Open dialog box appears. Search for a file created in Crystal Analysis Professional.

Defining the structure of the grid

Use the Rows / Columns screen to define how your data is structured within the grid. You can place the dimensions into either the Columns or Rows areas.

Tip: It is also possible to drag and drop the dimensions into either the Rows or Columns area.

1. Select a dimension to appear in the report.
2. Click the > arrow adjacent to either the Rows or the Columns areas, depending on where you would like the dimension to be located.
3. Continue adding dimensions to the Rows and Columns areas.



4. Click the **Up** and **Down** arrow buttons associated with the Rows and Columns areas to arrange the order of the dimensions.

   Note:

   - If you accidentally add a dimension to either the Rows or Columns area, click the < arrow to return it to the Dimensions list.
   - Click “Use unique member name” to display short names for members in your dimension (for example, months might appear as M01, M02, and so on). Click “Use member caption” to display long names (for example, January, February, and so on).

5 Select a dimension in either the Rows or the Columns areas and click **Select Row Members** to specify the members to be included in your report.

The Member Selector dialog box appears.

6 Select the members you would like to include.

7 Click **OK**.

8 Click **Next**.

The Slice / Page screen appears.

Setting sliced dimensions and specifying the number of grids

Use this screen to set sliced dimensions and to specify the number of grids that are required.

The Slice area determines the boundaries that will be placed on your report. For example, if you have a time dimension that is made up of fiscal quarter members, you could specify the report return the results for a specific quarter.

Note: If all of your members were used in the Rows / Column tab, nothing will appear here.

The Page area enables you to determine the number of grids and the subject of each. For example, if you have a product dimension, you could put this in the page area and specify two different products. This would result in two grids with the same rows, columns, and formatting, but each one would be based on a different product.

Specify a slice

1 Double-click a dimension in the **Slice** list to determine the member that is going to be used as the slice.

The Member Selector dialog box appears.

2 Select the appropriate member, expanding the structure if necessary.

3 Click **OK**.

4 Click **Next** if you do not want to add a page.

The Style screen appears.

Add a page

1 Use the > arrow to add dimension(s) to the **Page** list.

The Member Selector dialog box appears.

2 Expand the dimensions structure and select the appropriate fields.

3 Click **OK**.

4 Click **Next**.

The Style screen appears.

Add a parameter

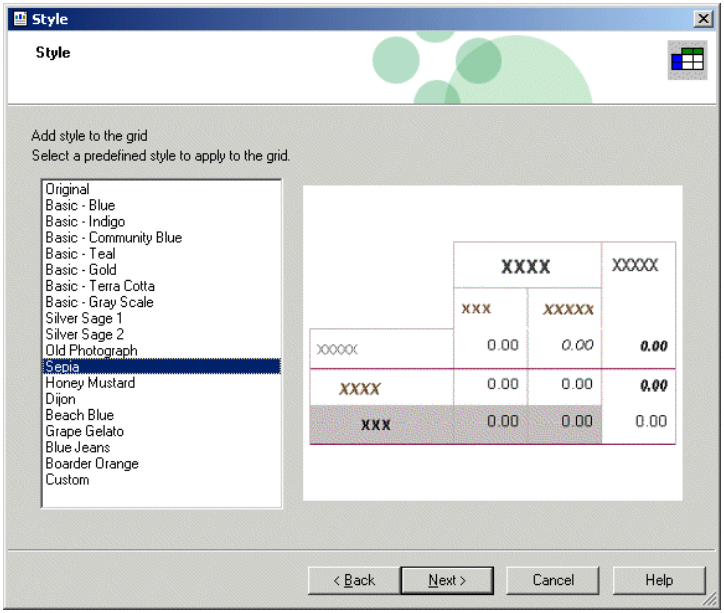
You can create a parameter field to link to any dimension by clicking the Create/Edit button. When you add a parameter to a dimension in your OLAP grid, users are prompted to select a value when refreshing report data. For more information about parameter fields, see "Parameter Fields" on page 349.

Applying a predefined style

1 Select a predefined style for the grid from the list.

Tip: If you do not want to use a predefined style, click Next.

A preview of the color scheme you select appears on the right.



Note: The color of the style you select may not appear correctly if your screen resolution is set to 256 colors. Increase the resolution to correct this.

2 Click **Next**.

The Chart screen appears.

Inserting a chart

1 Select the kind of chart you want to add to your report from the options shown on the Chart screen.

Tip: If you do not want to insert a chart, click Finish.

2 Add a title for your chart.

3 In the **On change of** list, select the dimension you want to base your chart on.

4 If necessary, in the **Subdivided by** list, click a secondary row or column you want to base your chart on.

Note: Be sure that the chart type selected in Step 3 supports a secondary charting field.

5 Click **Other Dimension** to define field values for any dimensions you've used in your OLAP grid but not in your chart.

For more information refer to "Charting on an OLAP cube (OLAP layout)" on page 226 and "Creating charts" on page 222.

6 To finalize your report, click Finish.

Updating an OLAP report

The location of the OLAP data accessed by your report may change. Some possible scenarios include:

- The location of the OLAP server or database may change due to restructuring of Information System resources.
- Another cube instance may have been successfully processed and reflect more current information.
- New reports may be developed on a smaller database that represents a subset or production version of the primary OLAP database, then ported over to the actual working database to report on real data.

Before updating your report, it is important to ensure that it is compatible with the data. Some things to look for include:

- Removing dimensions contained in the report that are not in the cube.
- Removing fields referred to explicitly in the design of the report, such as a formula.

Note: If a report contains subreports or grids, and the database used for them have changed name or location, you must update each subreport or grid.

To update a cube location in an OLAP grid object

1 Select the grid by clicking on the border.

2 From the **Database** menu, click **Set OLAP Cube Location**.

Tip: You can also right-click the grid and select Set OLAP Cube Location.

The Confirm Command dialog box appears.

3 Click **Yes**.

The Set OLAP Cube Location dialog box appears.

4 Click **Select**.

The Crystal OLAP Connection Browser appears.

5 Browse your OLAP server for the cube you want to connect to.

If your server isn't in the list, click Add Server. In the New Server dialog box, provide the server information; then click OK.

6 Select the desired cube and click **Open**.

The Set OLAP Cube Location dialog box reappears.

7 Click **OK**.

The Propagate Set Location dialog box appears.

8 If you have more than one OLAP grid in your report, click **Yes** to update the cube location of the remaining grids; otherwise go to the next step.

9 If you also want to change the relational database location to the new data source, click **Yes**; otherwise click **No** to return to the report.

If you click Yes, the Set Datasource Location dialog box appears.

Updating the database location

1 In the Set Datasource Location dialog box, expand the **More Data Sources** folder in the "Replace with" area.

2 Search the **OLAP** folder for the new cube location.

Tip: You can use the Make a New Connection option to search for the cube in the Crystal OLAP Connection Browser.

3 Select the data source name, or an individual table, and click **Update**.

The data source name is changed in the Current Data Source area.

4 Click **Close**.

Formatting data in an OLAP grid

There are several ways to add formatting to rows and columns in an OLAP grid:

- Use the Highlighting Expert to apply conditional formatting to all types of report fields: Number, Currency, String, Boolean, Date, Time, and DateTime fields.

  To access the Highlighting Expert, right-click the field you want to format and, on the shortcut menu, click Highlighting Expert. For more details, see "Using the Highlighting Expert" on page 212.

- Use the Format Editor to apply absolute formatting to fields in the grid. Absolute formatting is applied under all conditions, irrespective of the data values in the field.

  To access the Format Editor, right-click the field you want to format; then, on the shortcut menu, click Format Field. For more details, see "Working with absolute formatting" on page 199.

- Use the Formula Workshop to write your own conditional formatting formulas with Crystal or Basic syntax. The Formula Workshop gives you maximum control over the formatting of the OLAP grid.

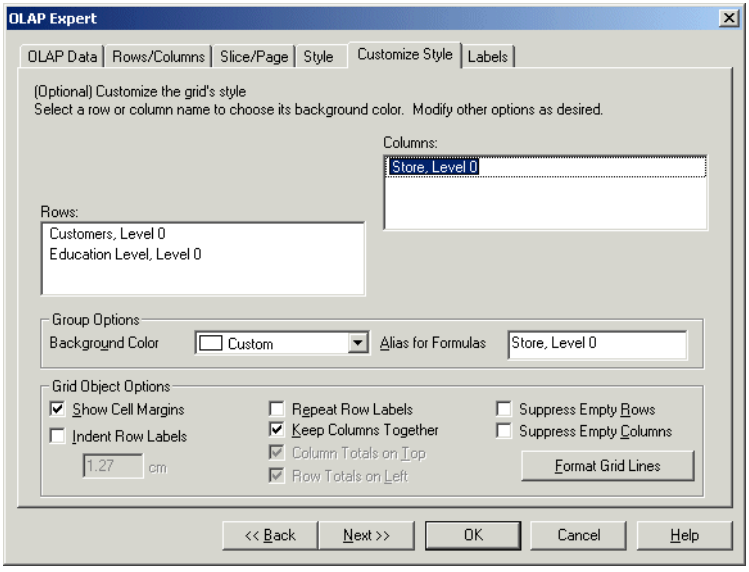
  To access the Formula Workshop, open the Format Editor and click the appropriate Formula button. For more details, see "Working with conditional formatting" on page 208.
- Use the OLAP Expert to re-format the entire OLAP grid. This expert provides all of the formatting options that are available in the OLAP Report Creation Wizard.

  To access the OLAP Expert, select the entire grid object by clicking one of its borders. Right-click the selected grid and, on the shortcut menu, click OLAP Expert.

  The OLAP Expert has two tabs not found in the OLAP Report Creation Wizard. Use the Customize Style and Labels tabs to format your OLAP grid.

Changing the background color of a dimension

1. On the Customize Style tab of the OLAP Expert, select a dimension.



2. In the Group Options area, select the color from the **Background Color** list.
3. Click **OK** if you have finished customizing your grid.

Creating an Alias for a Dimension

You can create aliases to shorten long dimension names. This is useful when you plan on referring to a dimension in a conditional formatting formula (by using the GridRowColumnValue function).

1. On the Customize Style tab of the OLAP Expert, select the dimension.
2. In the Group Options area, enter an alias name in the **Alias for Formulas** field.
3. Click **OK** if you have finished customizing your grid.

Formatting grid lines

In the Format Grid Lines dialog box, you can define whether the lines show or not, as well as the color, style, and width.

1. On the Customize Style tab of the OLAP Expert, click **Format Grid Lines**.

   The Format Grid Lines dialog box appears.
2. From the list, choose the description of the area where you would like the lines to appear, or click the appropriate area in the Format Grid Line diagram.
3. Select the color, style, and width.
4. Select the **Draw** check box to have your lines appear in the report.
5. Click **OK**.
6. Click **OK** if you have finished customizing your grid.

Labelling dimensions

In the Labels tab, you are able to define which dimensions will be labeled and how they will be displayed in the grid. The dimensions available to you are the ones that have been designated as a page or slice.

1. Using the arrow buttons, move the dimensions to the Unlabeled and Labeled areas as required.
2. Make changes to the appearance of the label by using the options available in the Label Position and Label Spacing areas.
3. Click **OK** if you have finished customizing your grid.

Changing the view of OLAP data

When analyzing OLAP data in your report, there are several basic ways to change how the data is displayed in the grid.

Note: The methods described here let you manipulate your OLAP grid directly from the Preview tab of Crystal Reports. You can also carry out the same and additional, more advanced, functionality on the Cube View tab. Search the online help for "Cube View tab" to learn more about this feature.

To show or hide dimension members

1. Right-click the dimension whose members you want to show or hide.
2. On the shortcut menu, click **Expand Member** or **Collapse Member**.

   The dimension expands to show its members, or collapses to hide them.

   Tip: You can also double-click parent dimensions to drill down into the hierarchical data displayed in the OLAP grid.

To add totals to an OLAP grid

1. Right-click the dimension you want to see totals for.
2. On the shortcut menu, select **Automatic Totals**.

   A submenu appears with options for the kind of totals you can add.
3. Select the option you want.

A "Total" row or column (or both) is added to your OLAP grid. To remove totals later, select No Totals.

To alter the data displayed in the OLAP grid

- Drag and drop rows and columns to rearrange the data in the OLAP grid.
- Drag and drop dimensions to and from the OLAP Labels area to add or remove data from the grid.

To pivot an OLAP grid

- Right-click the grid border and select **Pivot OLAP Grid** from the shortcut menu.

  This changes the orientation of your OLAP grid by switching the row and column dimensions.

To define the order of fields in the OLAP grid

1. Expand the row or column dimension whose fields you want to reorder.
2. Right-click the dimension and, on the shortcut menu, click **Reorder Displayed Members**.
3. In the Reorder Displayed Members dialog box, select fields and click the **Up** and **Down** arrows to rearrange the members that are displayed in the grid.
4. Click **OK** to effect your changes and return to the OLAP grid.

Sorting and filtering OLAP grid data

The sorting capabilities of the OLAP grid enable you to order data by row and column values. If you want to limit the data based on particular field values, add one or more filters to the grid. Filters also allow you to perform top or bottom N analysis (by actual values and by percentage).

- Sorting data in an OLAP grid
- Filtering data in an OLAP grid

Sorting data in an OLAP grid

Use the OLAP grid's sorting capabilities to quickly arrange grid data in a useful manner. You can add, edit, and remove sorts by right-clicking the appropriate row or column member and selecting from the shortcut menu. You can sort data in ascending or descending order, and you can specify whether or not you want to break grid hierarchies.

In the following grid, for instance, an ascending sort has been added to the Budget column.

			Budget
All Products			**2,449,052.00**
	Frozen Goods		**106,392.00**
		Pastry	**395.93**
		Frozen Produce	**97,490.98**
	Bakery		**173,531.50**
		Cakes and Pies	**72,723.45**
		Loaves and Buns	**86,326.31**

In this case, the OLAP grid respects the parent/child relationships between grid members and sorts the data values accordingly. (Frozen Goods precedes Bakery, but Pastry follows Frozen Goods.)

In the next example, the Budget column remains sorted in ascending order; in this case, however, the Break Hierarchies option is selected.

			Budget
All Products	Frozen Goods	Pastry	**395.93**
	Bakery	Cakes and Pies	**72,723.45**
		Loaves and Buns	**86,326.31**
	Frozen Goods	Frozen Produce	**97,490.98**
	Frozen Goods		**106,392.00**
	Bakery		**173,531.50**
All Products			**2,449,052.00**

Now the Report Designer disregards parent/child relationships between grid members and sorts on the basis of data value alone. (Frozen Goods still precedes Bakery, but Pastry precedes all others.)

You can add up to three sorts to grid rows and up to three sorts to grid columns. In each case, the first sort takes precedence, and each subsidiary sort serves to further differentiate between grid data. If you add a fourth sort to a row or to a column, your first three are removed from the grid, and the new sort becomes the single, primary sort.

Tip: To locate a sorted row or column, move the mouse pointer over the OLAP grid. When you reach a sorted member, the pointer turns into a double-arrow.

To sort data in the OLAP grid

1. Right-click the row or column member that you want to sort by.
2. On the shortcut menu, point to **Add First Sort.**
3. From the submenu, select from the available sort options:
   - Ascending
   - Descending
   - Ascending, Break Hierarchies
   - Descending, Break Hierarchies

Note: An ascending sort on a grid row orders data values from lowest to highest, left to right. An ascending sort on a column orders data values from lowest to highest, top to bottom.

To reverse the sort direction

- Right-click the sorted grid member and, on the shortcut menu, click **Change Direction of Sort**.

To add a subsidiary sort

1. If you added the first sort to a row, right-click another row member; if you added the first sort to a column, right click another column member.
2. On the shortcut menu, point to **Add Next Sort**.
3. From the submenu, select from the available sort options:
   - Ascending, Break Hierarchies
   - Descending, Break Hierarchies

The subsidiary sort enables you to differentiate between members that possess identical data values in the first sorted row or column. Similarly, you may add a third sort in order to further differentiate between grid members.

To remove a sort

- Right-click the sorted grid member and, on the shortcut menu, click **Remove Sort**.

Filtering data in an OLAP grid

Use filters to exclude grid data that are not important, or to display only the data that you want to see. You can filter grid data by actual values, or you can choose to exclude or display the top or bottom N, or the top or bottom N%.

Tip:

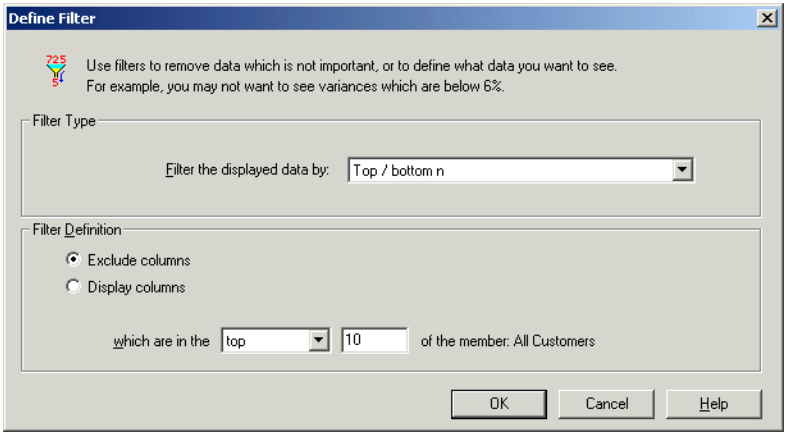
- To locate a filtered row or column, move the mouse pointer over the OLAP grid. When you reach a filtered row or column, the pointer turns into an X.
- If you filter all the cells in your OLAP grid, right-click the empty grid and select Remove All Filters from the shortcut menu.

To add a filter

1. Right-click the field whose values you want to filter.
2. On the shortcut menu, click **Add Filter**.
3. In the Define Filter dialog box, use the **Filter Type** list to specify how you want to filter the data.

   You can filter data by “Actual values,” or by selecting “Top / bottom n” or “Top / bottom n%.”
4. Use the **Filter Definition** options to specify which rows or columns you want to exclude or display.

For instance, this filter displays only the columns that are in the top ten when ranked by “All Customers.”



5 Click **OK** to add the filter and return to the OLAP grid.

Adding calculations to OLAP grids

Calculated members enable you to perform specific calculations on OLAP data and to display the results in rows or columns that are added to the OLAP grid. You can create a quick calculation, such as a Variance member calculated by subtracting Budget from Sales. Or you can use functions to perform complex Statistical Analysis or Time Series calculations.

Define your own calculation with an MDX (multidimensional expressions) query, or use the Calculation Expert to add one of these predefined calculations to your OLAP grid:

- **Contribution (%)**
  This shows the percentage contribution by each member in a hierarchical dimension. See “Calculating contribution (%) with the Calculation Expert” on page 292.
- **Growth**
  This shows how a value has grown from one point in time to another, expressed either in absolute or in percentage terms. See “Calculating growth with the Calculation Expert” on page 292.
- **Variance**
  This compares the value of one dimension member with a target value; the resultant variance can be expressed as an absolute value or as a percentage variance. See “Calculating variance with the Calculation Expert” on page 293.

For details on defining your own calculation, see "Defining a calculated member as an MDX query" on page 293.

Note: You can add several calculated members to an OLAP grid. The order in which they are resolved is the order in which they are added to the grid. This is important if one calculation uses the results of another.

Calculating contribution (%) with the Calculation Expert

1. Right-click the row or column member where you want the calculated member to appear.
2. On the shortcut menu, click **Add Calculated Member**.
   The Calculated Members dialog box appears.
3. On the **Calculation Expert** tab, type a **Calculation Name**.
4. In the **Calculation Type** list, click **Contribution (%)**.
5. In the Contribution Definition area, use the tree view to locate and select the member whose contribution you wish to calculate.
6. Right-click the selected member and, on the shortcut menu, click **Add to 'Contribution of' field**.
7. In the **To Dimension** List, click the dimension to which the selected member has contributed.
8. Select **Calculate contribution to the overall total** or **Calculate contribution to each level**.
   Tip: Click the Calculation tab to see the MDX query that the expert has formulated.
9. Click **OK** to add the calculated field to the OLAP grid.

Calculating growth with the Calculation Expert

1. Right-click the row or column member where you want the calculated member to appear.
2. On the shortcut menu, click **Add Calculated Member**.
   The Calculated Members dialog box appears.
3. On the **Calculation Expert** tab, type a **Calculation Name**.
4. In the **Calculation Type** list, click **Growth**.
5. In the Growth Calculation area, use the tree view to locate and select the member whose growth you wish to calculate.
6. Right-click the selected member and, on the shortcut menu, click **Add to 'Calculate Growth of' field**.

7 Select the **Time Dimension** against which you wish to calculate growth.

8 Select **Calculate growth as a percentage** if this is what you want.

Tip: Click the Calculation tab to see the MDX query that the Expert has formulated.

9 Click **OK** to add the calculated member to the OLAP grid.

Calculating variance with the Calculation Expert

1 Right-click the row or column member where you want the calculated member to appear.

2 On the shortcut menu, click **Add Calculated Member**.

The Calculated Members dialog box appears.

3 On the **Calculation Expert** tab, type a **Calculation Name**.

4 In the **Calculation Type** list**,** click **Variance**.

5 In the Variance Calculation area, use the tree view to locate and select the member you wish to compare against a target member.

6 Right-click the selected member and, on the shortcut menu, click **Add to 'Compare' field**.

7 Locate and select the member you want to be the *target* for calculating the variance.

8 Right-click the selected member and, on the shortcut menu, click **Add to 'To target of' field**.

9 Choose whether **Values less than the target are good** or **Values less than the target are bad**.

10 Select **Calculate variance as a percentage** if this is what you want.

Tip: Click the Calculation tab to see the MDX query that the Expert has formulated.

11 Click **OK** to add the calculated member to the OLAP grid.

Defining a calculated member as an MDX query

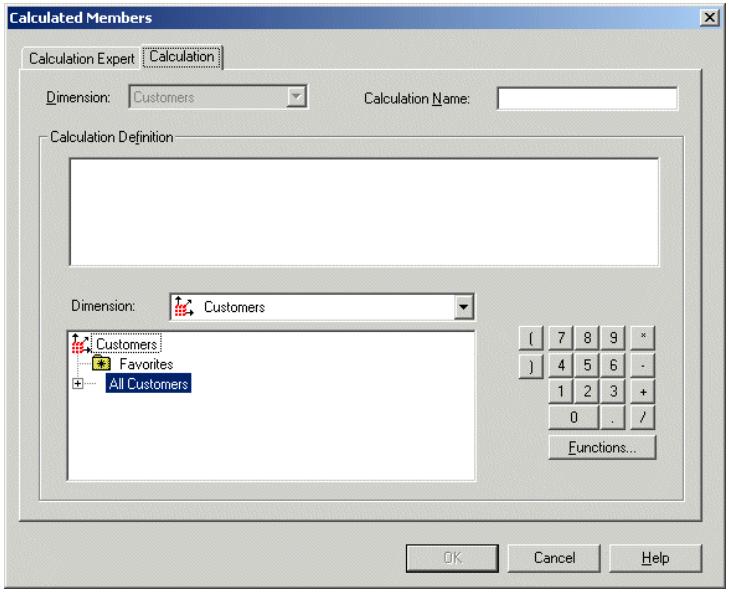
If you are familiar with multidimensional expressions (MDX), you can define a calculated member as an MDX query.

1 On your OLAP grid, right-click the row or column member where you want the calculated member to appear.

2 On the shortcut menu, click **Add Calculated Member** in order to access the Calculated Members dialog box.

3 In the **Calculation Type** list, click **Custom**.

4 Click the **Calculation** tab.



The Calculation tab consists of:

- A Calculation Definition box where you enter your MDX query.
- A Calculation Name box for the name of the new calculated member.
- Tools to help you define the calculation (Dimension list, tree view, keypad, and Functions button).

5 Type a **Calculation Name**.

6 In the Calculation Definition area, enter your calculation as an MDX query.

Use the Dimension list and the tree view to locate and select members for the calculation. Double-click any member to add it to the calculation.

You can also use functions in your definition. For information on adding functions from the library, see "Using MDX functions in a calculated member" on page 294. As well, see "Multidimensional expressions (MDX) functions reference" in the online help.

7 Click **OK** to add the calculated member to the OLAP grid.

Using MDX functions in a calculated member

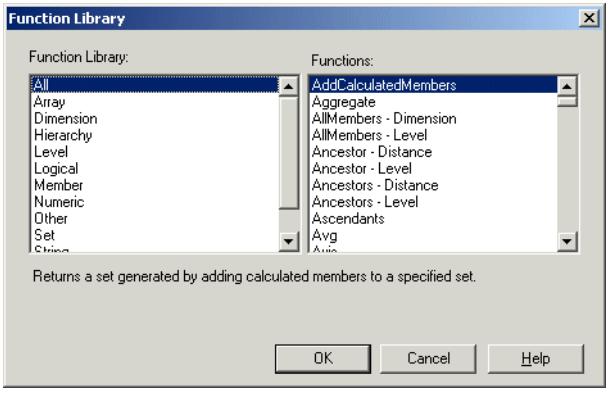
When defining your own calculated member with an MDX (multidimensional expressions) query, you can utilize the standard range of MDX functions. Type the required function directly, or select from the Function Library, which displays each of the available functions along with a brief description.

The available functions and their corresponding syntax blocks vary slightly, depending on the interface you use to connect to your OLAP data. For more information, see "Multidimensional expressions (MDX) functions reference" in the online help.

For more details on MDX functions, consult your OLAP provider's documentation (such as the *Microsoft® SQL Server™ OLAP Services* documentation).

To add an MDX function to a calculated member

- When defining your own calculated member, type the required function into the Calculation Definition area, or click the Functions button and select from the Function Library dialog box.



Tip: Select a function to view its description. Double-click a function to add it to your calculation.

For more information, see "Defining a calculated member as an MDX query" on page 293.

Editing calculated members

1. Select the calculated member and right-click its heading.
2. On the shortcut menu, click **Edit Calculated Member**.
   The Calculated Members dialog box appears, allowing you to edit the parameters of the member.

Removing calculated members

1. Select the calculated member and right-click its heading.
2. On the shortcut menu, click **Remove Calculated Member**.
   The member is removed from the OLAP grid.

Distributing and Viewing Reports 18

This chapter provides you with information about how to distribute finished reports using a variety of methods (printing, faxing, exporting). It also includes information about Report Parts and how to set up hyperlinks so you can see Report Parts in various Report Viewers. Finally, it ends with information about using smart tags with report objects in Office XP.

Distributing reports

Crystal Reports enables you to distribute your report using a variety of methods. This section covers:

- Printing a report
- Faxing a report
- Exporting a report
- Working with Enterprise folders

Printing a report

To print a report

1 On the **File** menu, select **Print**, then click **Printer**.

Tip: Another way to do this is to click the Print button on the Standard toolbar.

The Print dialog box appears.

2 Choose the following:

- Print range: all pages or a specific range.
- Number of copies.
- Collate copies: if this option is selected, the report will print each page in order. For example, if you are printing two copies of a report with four pages your report will print page one, two, three, and four of the first copy, then print the second copy.

3 Click **OK**.

The Printing Records dialog box appears showing the progress of your print job.

Faxing a report

Many fax applications, such as Microsoft® Fax and Delrina™ WinFax, allow you to set up a printer driver that will fax documents over a modem. When using one of these applications, you can fax a report from Crystal Reports.

To fax a report

1 On the **File** menu, click **Printer Setup**.

The Print Setup dialog box appears.

2 Select the fax driver from the **Name** drop-down list.

3 Click **OK**.

4 On the **File** menu, select **Print**, then click **Printer**.

The Print dialog box appears.

5 Click **OK**.

Your fax application appears, prompting you to select a cover page and to fill in the appropriate fax information.

Exporting a report

Finished reports can be exported to a number of popular spreadsheet and word processor formats, as well as to HTML, ODBC, and common data interchange formats. This makes the distribution of information easier. For example, you may want to use the report data to project trends in a spreadsheet package or to enhance the presentation of data in a desktop publishing package.

Note: Crystal Reports lets you insert objects anywhere on the report page. When you export to formats such as MS Word, MS Excel, and HTML, however, objects that you placed between lines are moved to the closest line in the output. To alleviate the formatting problems this behavior could cause, it is recommended that you use Guidelines when designing your reports. For more information, see "Designing with guidelines" on page 187.

The exporting process requires you to specify a format and a destination. The format determines the file type, and the destination determines where the file is located.

Format Types

Crystal Reports provides you with many different export format types. They include:

- Adobe Format (PDF)
- Crystal Reports (RPT)
- HTML 3.2
- HTML 4.0
- MS® Excel 97–2000
- MS® Excel 97–2000 (Data only)
- MS® Word
- ODBC
- Record style (columns no spaces)
- Record style (columns with spaces)
- Report Definition
- Rich Text Format
- Separated Values (CSV)
- Tab-separated text
- Text
- XML

Note: When you export to Text format, the export is based on characters per inch.

In addition to the standard export format types installed on your machine you may find additional export format types are available to you. These are determined by the DLL files on your local machine.

Note: When you export a report to a file format other than Crystal Reports format (.RPT), you may lose some or all of the formatting that appears in your report.

However, the program attempts to preserve as much formatting as the export format allows.

Destination

The destination determines the export location of your report. Crystal Report enables you to choose one of six destinations. They include:

- Application
- Disk file
- Exchange® Folder
- Lotus® Domino™
- Lotus® Domino Mail
- MAPI (Microsoft Mail™)

Exporting to an application

If you export to an application, the program exports the report to a "temp" file in the specified format and then opens the file in the appropriate application.

Note:

- When exporting in ODBC format, "Application" and "Disk file" are the same.
- You will need to specify a file path if you are exporting in HTML format.
- The file name of the report and the file name of the "temp" file cannot be the same.

The sections below provide instructions on how to export a report to the MS Excel 97–2000 format for each of the different destination types.

To export to an application

1. Open the report you would like to export.
2. On the **File** menu, click **Export**.

   Tip: Another way to do this is to click the Export button on the Standard toolbar.

   The Export dialog box appears.
3. Select the export format type from the **Format** drop-down list.

   In this case, choose MS Excel 97–2000.
4. Select **Application** from the **Destination** drop-down list.
5. Click **OK**.

   The Excel Format Options dialog box appears. Most of the options in this dialog box are self explanatory; for example, you can choose to export pages headers and footers, or you can choose to export a page range.

   The options in the "Column width" area let you define the width of the Excel cells in your output based on a point width (a constant width) or a width determined by objects in various sections of your Crystal report.

Note: Selecting "Whole Report" does not necessarily emulate the layout of your Crystal report. Rather, the Excel cell width is based on objects found in any section of your report. Likewise, for example, selecting "Details" creates cell widths based on the objects found only in the report's Details section.

6 Change the formatting options as needed.

7 Click **OK**.

The Exporting Records dialog box appears.

Note: Click Cancel Exporting to cancel the export process.

The program exports the report and opens it in the appropriate application. In this example, Microsoft Excel opens with the exported data.

Exporting to a disk file

If you export to a disk file, the program saves the report to the disk or diskette you have specified.

To export to a disk file

1 Open the report you would like to export.

Tip: Another way to do this is to click the Export button on the Standard toolbar.

2 On the **File** menu, click **Export**.

The Export dialog box appears.

3 Select the export format type from the **Format** drop-down list.

In this case, choose MS Excel 97–2000.

4 Select **Disk file** from the **Destination** drop-down list.

5 Click **OK**.

The Excel Format Options dialog box appears. For more information about this dialog box, see step 5 in "Exporting to an application" on page 300.

6 Change the formatting options as needed.

7 Click **OK**.

The Select Export File dialog box appears.

8 Select the appropriate directory.

9 Enter the File name.

10 Click **Save**.

Note: Click Cancel Exporting to cancel the export process.

The program exports the report and saves it in the designated directory. In this example the file is saved in the "temp" directory in an .xls format.

Exporting to a Microsoft Exchange folder

Crystal Reports enables you to export a report file to a Microsoft Exchange folder. You select the folder, and the report is stored there in the format that you specify. A Microsoft Exchange folder can contain standard notes (mail), files, and instances of Microsoft Exchange forms.

To export to a Microsoft Exchange folder

1. Open the report you would like to export.
2. On the **File** menu, click **Export**.

   Tip: Another way to do this is to click the Export button on the Standard toolbar.

   The Export dialog box appears.
3. Select the export format type from the **Format** drop-down list.

   In this case, choose MS Excel 97–2000.
4. Select **Exchange Folder** from the **Destination** drop-down list.
5. Click **OK**.

   The Format Options dialog box appears. For more information about this dialog box, see step 5 in "Exporting to an application" on page 300.
6. Change the formatting options as needed.
7. Click **OK**.

   The Choose Profile dialog box appears.

   Note: You will be prompted by the Microsoft Outlook™ Setup Expert if Microsoft Exchange, Microsoft Mail, or Internet E-mail is not set up on your machine.
8. Select the desired profile from the **Profile Name** drop-down list.

   If the profile is not listed, click New to create it.
9. Click **OK** when finished.
10. When the Select a Folder dialog box appears, select the folder in the profile in which you want the report to appear, and click **OK**.

The report is exported to the Microsoft Exchange folder you selected. The exported report can now be accessed through the Microsoft Exchange client.

Exporting to Lotus Domino

You must have Version 3.0 or later of the Lotus Domino client. You will also require, at a minimum, depositor access. Crystal Reports will not export to a Lotus Domino OS/2 client.

Note: Before you export to Lotus Domino, you must create a Form called "Report Form" that includes a Comments field that you can edit, and a View that has two columns: one called "#" (created by default) and another that points to the Comments field.

To export to Lotus Domino

1 Open the report you would like to export.

2 On the **File** menu, click **Export**.

Tip: Another way to do this is to click the Export button on the Standard toolbar.

The Export dialog box appears.

3 Select the export format type from the **Format** drop-down list.

In this case, choose MS Excel 97–2000.

4 Select **Lotus Domino Database** from the **Destination** drop-down list.

5 Click **OK**.

The Format Options dialog box appears. For more information about this dialog box, see step 5 in "Exporting to an application" on page 300.

6 Change the format settings as required.

7 Click **OK**.

The Select Database dialog box appears.

8 Double-click the Lotus Domino server you would like to export your report to.

The file name defaults.

9 Select the database you would like to export the report to.

10 Click **OK**.

The Comments dialog box appears.

11 Type in any comments that are to appear when another user selects your report from the Lotus Domino Desktop.

12 Click **OK**.

The export process begins.

The next time a user logs onto the Lotus Domino database you specified, they will see the report in their desktop. The user can double-click the report file name to display the comments you wrote and then double-click the report icon to view the report.

Exporting to MAPI (Microsoft Mail)

Note: This option works only if you have a mail client installed (MS Outlook, MS Mail, or Exchange). Because the exported report is attached to an email message, you must also have an email account configured properly.

1 Open the report you would like to export.

2 On the **File** menu, click **Export**.

Tip: Another way to do this is to click the Export button on the Standard toolbar.

The Export dialog box appears.

3 Select the export format type from the **Format** drop-down list.
In this case, choose MS Excel 97–2000.

4 Select **Microsoft Mail (MAPI)** from the **Destination** drop-down list.

5 Click **OK**.
The Format Options dialog box appears. For more information about this dialog box, see step 5 in "Exporting to an application" on page 300.

6 Change the formatting options as needed.

7 Click **OK**.
The Send Mail dialog box appears.

8 Enter the address details, then click **Send**.
The Exporting Records dialog box appears.

Note: Click Cancel Exporting to cancel the export process.

Working with Web Folders

You have the ability to open and save reports using Web Folders if you:

- Are running Windows 2000 or have Office 2000 installed.
- Have access to a web server that is configured to support Web Folders.
- Add a Web Folder from this server into your Web Folders folder.

To open your report

1 On the **File** menu, click **Open**.
The Open dialog box appears.

2 Click **Web Folders**.

3 Open the folder that contains the report.

4 Double-click to open the report.

To save your report

1 On the **File** menu, click **Save As**.
The Save As dialog box appears.

2 Click **Web Folders**.

3 Locate the folder you would like to save the report to.

4 Enter the file name.

5 Click **Save**.

Note: After making changes to a report in a Web Folder, you must save the changes to the same file in the same Web Folder.

Working with Enterprise folders

You can open reports from Crystal Enterprise folders in Crystal Reports. If Crystal Reports detects Crystal Enterprise on your machine, it displays the appropriate folders in the Open/Save As dialog box.

To open a report in an Enterprise folder

1. On the **File** menu, click **Open**.
2. In the Open dialog box, click **Enterprise Folders**.

   Note: The Enterprise Folders button is not visible if Crystal Reports doesn't detect Crystal Enterprise on your machine.

   If you have not already logged on to an Automated Process Scheduler (APS), the Connect to APS dialog box appears.

   Tip: You can also log on to an APS from the File menu before you open a report.

3. Click the **Logon using** list to select Enterprise Authentication or Windows NT Authentication.

   Enterprise Authentication requires a user name and password that is recognized by Crystal Enterprise.

   Windows NT authentication requires a user name and password that is recognized by Windows NT.
4. Enter your user name and password.
5. In the **APS** field, enter the name of the APS you want to connect to.
6. Click **OK**.

   The Enterprise folders appear and you can select a report to open in Crystal Reports.

Note: After making changes to a report in an Enterprise folder, you must save the changes to the same file in the same Enterprise folder.

Viewing reports

You can view Crystal reports using a number of report viewers available through Crystal Enterprise and the Crystal Reports Software Development Kit (SDK). For information about the Crystal Report Viewers, see the *ePortfolio User's Guide* in the Crystal Enterprise documentation or the *Developer's Guide* in the Crystal Reports documentation.

In general, the Crystal Report Viewers are page viewers that let you see complete pages of your Crystal reports. However, one viewer—the Report Part Viewer—lets you see specific report objects without viewing the entire page. Report objects displayed in such a way are referred to as Report Parts.

What are Report Parts?

Report objects displayed by themselves in a viewer—without the rest of the report page—are referred to as Report Parts.More precisely, however, Report Parts are objects that use hyperlinks to point from a home report object to a destination object.

Report Parts are designed to work with the DHTML viewer subset of the Crystal Report Viewers to expand the navigation possibilities within and between reports. Report Part hyperlinks can link to other objects in the current report or to objects in any other report. This linking lets you create a guided path through your reports so that only specific information is shown at each stop along the path.

Viewing Report Parts instead of the whole page is a powerful feature that allows you to seamlessly integrate reports into portal and wireless applications.

Report Part Viewer

The Report Part Viewer is a new viewer that lets you display Report Parts without the rest of the report page. This viewer can be integrated into web applications so that specific report objects can be shown to users without them having to see the rest of the report.

For the most part, you set up the Report Part hyperlinks in the Report Designer, but you take advantage of their functionality in the report viewers.

What is navigation?

The new navigation functionality in Crystal Reports lets you move to other report object(s) in the same report, or to object(s) in another report—with a specified data context. In this last case, the other report must be managed in Crystal Enterprise (CE). This navigation is available only in the DHTML viewers (zero-client, server-side viewers). Its advantage is that you can link directly from one object to another; the required data context is passed automatically so you go to the object and data that is relevant.

Report Parts use this new navigation functionality when linking between Report Part objects. The key difference between Report Part navigation and regular (page) navigation is that, when navigating with Report Parts (using the Report Part Viewer), you see only the objects identified as Report Parts. In regular navigation, you go to the identified object(s), but you see the entire page.

Setting up navigation

Navigation (page or Report Part) is set up on the Hyperlink tab of the Format Editor dialog box. When you select Report Part Drilldown or Another Report Object, you are prompted to add information that defines the hyperlink.

In the DHTML Viewer Only area of the Hyperlink tab, you can select:

- **the type of hyperlink you want to create**

  The Report Part Drilldown option is available for summary fields, group charts and maps, and fields in your report's group header or group footer. This option is relevant only to Report Parts; regular navigation (using the page view) drills down by default. For more information, see "Report Part-specific navigation" on page 308.

  The Another Report Object option can be used for both Report Part and page navigation. This option allows you to specify the destination object(s) you want to navigate to, and to specify the data context that should be passed.

- **the report to link to**

  The Select From field defaults to the current report (it is blank when the current report is referenced). Specific information can be added manually or pasted in.

- **the report instance**

  You can navigate to the latest instance of a report object that is managed through Crystal Enterprise. For example, you set the home report object to link to the target report object, and both reports run weekly. For every new instance of the home report, you will be able to navigate to the latest instance of the target report, not just the original one.

- **the report object(s) to link to (destination objects)**

  By selecting one or more objects in this field, you are identifying the destination objects for your navigation. In page navigation, this information determines what object you move to in the page. For Report Part navigation, this information determines what object(s) are displayed upon navigation (only the objects that are identified are displayed).

  Any of the following report object types can be selected in the Object Name field (you can select one or more objects from the same report section):

  - Field objects
  - Charts or maps
  - Bitmaps
  - Cross-tabs
  - Text objects

You cannot select the follow report objects types as destinations:

- Objects inside the Page Header or Page Footer.
- Lines or boxes.
- Subreports or any object inside a subreport.
- Entire sections (you must select the objects inside the section individually).

Specific information can be added in one of two ways:

- You can type the object name(s) as they appear in the Report Explorer.
- You can copy report object(s) from the current report or another and paste its information into the object field.

- **data context**

  The Data Context field will normally be completed when you paste information for the destination object(s). You can also customize this option by using the Conditional Formula button. For more information, see "Data context formats" on page 312.

Report Part-specific navigation

Report Parts use the new Crystal Reports navigation functionality. However, there are some special cases that apply only to Report Parts:

- Only the destination object identified will be displayed.
- Drill down must be specified using the Report Part Drilldown option so that the destination object(s) can also be specified.
- Initial Report Part Settings must be specified for each report that is the first stop on the Report Part navigation path. Any report that is viewed in the Report Part Viewer, and is not navigated to through Report Part navigation, requires Initial Report Part Settings so that the Report Part Viewer has a starting place for its display.

The Report Part Drilldown option

The Report Part Drilldown option lets you define a hyperlink so that the Report Part Viewer can emulate the drill-down functionality of Crystal Reports. The Report Part Viewer displays only destination objects; therefore, to make drill down work, you need to define a navigation path from a home object to one or more destination objects. When you have multiple destination objects, they must all reside in the same report section.

The Report Part Drilldown option does not affect the DHTML page viewers since the option emulates the default Crystal Reports behavior for drill down (which the page viewers already support). Page viewers, however, do not limit which objects are displayed—they always show all report objects.

Note: The Report Part Drilldown option can be used only to navigate between objects in the same report.

Because the Report Part Viewer shows only destination objects, you must define Initial Report Part Settings (that is, a default home object) for a report before your Report Part Drilldown hyperlinks can work. A report's Initial Report Part Settings define the object that appears first in the Report Part Viewer. Think of this object as the place you'll begin your path of drill-down hyperlinks from.

To define Initial Report Part Settings

1. Open the report whose default home object you want to define.
2. Right-click the object you want to set as the default home object and select **Copy** from its shortcut menu.

   Tip:
   - You might find it more useful to copy your hyperlink information from the Design tab because the data context will be more general (that is, you won't be copying only a specific record as you might on the Preview tab).
   - For more information, see "Data context formats" on page 312.
3. On the **File** menu, click **Report Options**.
4. In the Initial Report Part Settings area of the Report Options dialog box, right-click the **Object Name** field and select **Paste** from its shortcut menu.

   The name and data context of the report object you selected as your home object are pasted into the appropriate fields.
5. Click **OK**.

To create a Report Part Drilldown hyperlink

Note: Before you begin this procedure, be sure to read "Setting up navigation" on page 307 to acquaint yourself with the limitations of creating this type of hyperlink.

1. Open a report and select the intended destination object; then click the Format button on the Expert Tools toolbar.

   Tip: You can also do this by selecting Format Field from the Format menu.
2. In the Format Editor, click the **Hyperlink** tab.
3. In the DHTML Viewer Only area, select **Report Part Drilldown**.

   The Hyperlink Information area changes to show the appropriate fields for this type of hyperlink.
4. Enter the field name of your destination object, or select it from the list associated with the **Object Name** field.

   Tip: Use the Report Explorer to quickly identify the default names assigned to each of your report objects. To open the Report Explorer, click its button on the Standard toolbar. The Report Explorer can be open while you're working on the Hyperlink tab of the Format Editor.

5 If you want to add another field name to the **Object Name** field, click the field and move the cursor to the end of the existing text. Enter a semi-colon (;) and then the name of another field from the same section of the report.

6 Click **OK**.

You have established a hyperlink from your report's home object to a destination object or objects. In the Report Part Viewer, you will see the home object first, and when you click it for drill down, you will see the destination object(s).

The Another Report Object option

The Another Report Object option lets you define a hyperlink path for linking objects in the same or different reports. If you are defining a hyperlink path to a different report, that report must be managed in CE. The Report Part Viewer displays only the objects you specify. Like the Report Part Drilldown option, all destination objects must be from the same report section.

The Another Report Object option can also be used by the DHTML page viewer for navigation purposes.

Work flow

Because the Another Report Object option allows you to create hyperlinks between objects in different reports managed in CE, it requires more set up on the Hyperlink tab. This is an overview of the steps you have to complete to set up your hyperlink successfully.

- Open the report that contains the object you want to be the destination object and copy it.
- Open the report that contains the home object, select it, and open the Format Editor.
- On the Hyperlink tab, paste the information from the destination object into the appropriate fields.

The following procedure shows you how to do all of these steps.

To create an Another Report Object hyperlink

Note: Before you begin this procedure, be sure to read "Setting up navigation" on page 307 to acquaint yourself with the limitations of creating this type of hyperlink.

1 Open the report you want to link *to* and the report you want to link *from*.

2 In the report you want to link *to*, right-click the intended destination object and select **Copy** from its shortcut menu.

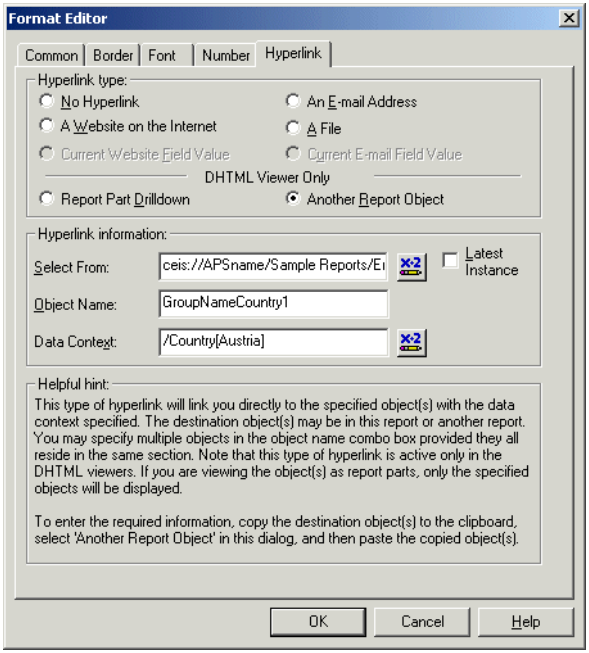
3 In the report you want to link *from*, select the intended home object; then click the **Format** button on the **Expert Tools** toolbar.

Tip: You can also do this by selecting Format Field from the Format menu.

4 In the Format Editor, click the **Hyperlink** tab, and then select the **Another Report Object** option.

5 In the Hyperlink Information area, right-click the **Select From** field and select **Paste** from its shortcut menu.

The identifying information from the destination object you selected in the first report is pasted into the appropriate fields.



6 If you want the information displayed for the hyperlink to be from the latest instance of the report (that is, the saved data from a particular version of the report), select **Latest Instance**.

If you select the check box, the string “,LatestInstance” is appended to your URI; if the check box is then cleared, the string is removed.

Note: If the report named in the Select From field is not connected to an Automated Process Scheduler (APS), this option is not available.

7 If you want to add other fields from the same section of the link *to* report, click the **Object Name** field and move the cursor to the end of the existing text. Enter a semi-colon (;) and then the name of the field(s).

8 You can broaden your **Data Context** by using an asterisk (*****) to represent all records in a group.

For example, if your Data Context is `/USA/CA/Changing Gears` (which will show only that specific detail record), you could change it to `/USA/CA/*` to show all detail records within that group.

Tip:

- You might find it more useful to copy your hyperlink information from the Design tab because the data context will be more general (that is, you won't be copying only a specific record as you might on the Preview tab).
- If your report has a group selection formula, check the data context for your destination object to ensure it contains the correct child index.
- For more information, see "Data context formats" on page 312.

9 Click **OK**.

You have established a hyperlink from your home object to a destination object or objects. In the Report Part Viewer, you would see the home object first, and when you clicked it for drill down, you would see the destination object(s).

Data context formats

- Normally, a data context formula appears as follows:

  ```
  "/" + {Table.Field} + "/" + {Table.Field}
  ```

- You can also use these formats:
  - XPath-like format:

    ```
    /USA/Bicycle
    ```

  - Strongly-Typed format:

    ```
    /Country[USA]/Product Class[Bicycle]
    ```

  Note:

  - You can add detail-level information in 0-based format:

    ```
    /USA/Bicycle/ChildIndex[4]
    ```

  - You can also use a wildcard to identify all instances:

    ```
    /USA/*
    ```

- For Cross-Tab objects, you can use navigation on cells, columns, or rows. To define the data context, use the `GridRowColumnValue` formatting function. For example:

  ```
  "/" + GridRowColumnValue ("Supplier.Country") + "/" + GridRowColumnValue
  ("Product.Product Class").
  ```

Hyperlinks displayed in the viewers

This section summarizes information about how hyperlinks work in the DHTML page viewers and the Report Part Viewer.

Action	DHTML page viewers	Report Part Viewer
Hyperlink to a Report Part or report object	• Allow the selection of one or more report objects (within the same section) to link to. • Drill down is the default behavior.	• Allow the selection of one or more report objects (within the same section) to link to. • Upon navigation, only the selected object(s) will be displayed.
Drill down to a specified destination object	• Allow the selection of one or more report objects (within the same section) to drill down to. • Drill down is the default behavior.	• Allow the selection of one or more report objects (within the same section) to drill down to. • Upon drill down, only the selected report object(s) are displayed.
Default drill down (no specified destination object)	• Drill down is the default behavior.	• Drill down is not allowed unless the destination report object(s) are identified.

Using smart tags

Crystal Reports lets you take advantage of smart tags in Office XP. When you paste a chart, a text object, or a field object into an Office XP application, you can view data from the host report after selecting a smart tag option. This table summarizes the options available to you.

Office XP application	Smart tag option	Crystal Reports object type
Word	View Refresh	Text object Field object Chart
Excel	View Refresh	Text object Field object
Outlook	View Refresh (only when creating a message)	Text object Field object Chart

Before you can use smart tags, web server options must be configured on the Smart Tag tab of the Options dialog box. As well, an .asp or . jsp page must be created for viewing report details. Normally, these tasks should be carried out by your system administrator. For more information, see the *Crystal Enterprise Report Application Server* online help.

Note: The reports you want to use with smart tags must exist on the web server named in the Options dialog box in a directory that mirrors their real location, or the web server must be configured to accept a UNC path.

To use smart tags with a Crystal Reports object

1. Open the Crystal report that contains the object you want to copy to an Office XP application.
2. On the **File** menu, click **Options** and ensure the options on the **Smart Tag** tab have been configured:
   - Your web server must be named.
   - A virtual directory must be named (a default is provided).
   - A viewing page must be named (a default is provided).
3. On the **Preview** tab of Crystal Reports, right-click the text object, field object, or chart you want to copy and select **Copy Smart Tag** from its shortcut menu.
4. Open the appropriate Office XP application, and paste the report object into a document, worksheet, or email message.

   Note: See the table in this section for limitations about which report objects can be pasted into each Office XP application.
5. In your Office XP application, choose the appropriate smart tag from the options for the pasted report object.

   Note: See the table in this section for limitations about which smart tags are available in each Office XP application.

   The .asp or .jsp page named in the Options dialog box appears showing the appropriate information from the host Crystal report.

Report Alerts 19

This chapter provides information about creating and using alerts in your Crystal reports.

About Report Alerts

Report Alerts are custom messages created in Crystal Reports that appear when certain conditions are met by data in a report. Report Alerts may indicate action to be taken by the user or information about report data.

Report Alerts are created from formulas that evaluate conditions you specify. If the condition is true, the alert is triggered and its message is displayed. Messages can be text strings or formulas that combine text and report fields.

Once a Report Alert is triggered, it's not evaluated again until you refresh your report's data.

Because Report Alerts are specific to each report, you decide when to use them and when not to. They can be useful to point out important information, such as sales that fall above or below a limit. And the message is created by you, so it can be specific to your data.

Working with Report Alerts

This section of the guide focuses on the tasks you'll need to undertake in order to use Report Alerts:

- "Creating Report Alerts" on page 316
- "Editing Report Alerts" on page 318
- "Deleting Report Alerts" on page 318
- "Viewing Report Alerts" on page 319
- "Referring to Report Alerts in formulas" on page 319

Creating Report Alerts

You must complete three steps when creating a Report Alert:

- Name the alert.
- Define the condition that triggers the alert.
- Create the message you want to appear when the alert is triggered (this step is optional).

To create a Report Alert

1. On the Report menu, point to **Alerts** and then click **Create or Modify Alerts**.
   The Create Alerts dialog box appears.
2. Click **New**.
   The Create Alert dialog box appears.
3. Enter a name for your new alert in the **Name** box.

4 Enter your alert message in the **Message** box.

The Message box lets you enter a message to be used as a default. If you want the same message to appear every time your alert is triggered, enter it in the Message box.

If, however, you want to use a formula so the message is customized with data elements, see the next step.

5 If you want to use a formula to create an alert message, click the formula button to the right of **Message**.

The Formula Workshop appears. For information on how to use the editor, see "Working with the Formula Editor" on page 337.

6 Enter your alert message formula.

For example, if you want to see the message "Country is a star performer" (where Country is the name of a specific country), you might create the following formula:

```
GroupName ({Customer.Country}) + " is a star performer"
```

Note:

- The result of an alert message formula must be a string.
- The DefaultAttribute function can be used to refer to a message added in the Message box. For example, if the message in your Message box is " is a star performer," your alert message formula might be:

  ```
  GroupName ({Customer.Country}) + DefaultAttribute
  ```

  This formula relies on the text you enter in the Message box; that text becomes the DefaultAttribute.
- Message condition formulas can be created using either Crystal Syntax or Basic Syntax.

7 Click **Condition**.

The Formula Workshop appears.

8 Enter your alert condition formula.

Alert formulas can be based on recurring records or on summary fields, but cannot be based on print-time fields, such as running totals or print time formulas. Alert formulas cannot have shared variables.

If an alert formula is based on a summary field, any recurring fields used must be constant over the summary field. For example, if you are grouping on Country, Region, and City, you might create an alert such as:

```
Sum ({Customer.Last Year's Sales}, {Customer.Region})
```

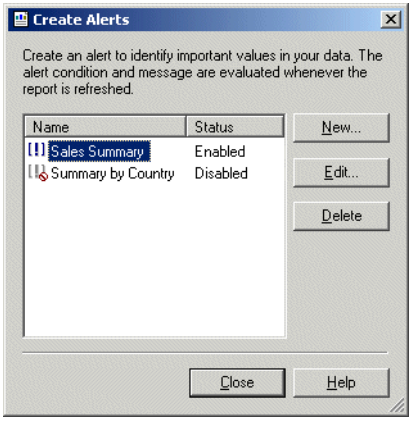
In this case, your formula can refer to either Country or Region, but not City or Customer Name since these are not constant.

Note: Alert condition formulas can be created using either Crystal Syntax or Basic Syntax.

9 Clear the **Enable** check box if you do not want the alert to be evaluated. Otherwise, leave it selected.

10 Click **OK** to save your alert.

You are returned to the Create Alerts dialog box and your new alert is listed. You can see its name and status (Enabled or Disabled).



Only enabled and disabled alerts appear in the Create Alerts dialog box. If an alert is triggered, it is seen in the Report Alerts dialog box.

Editing Report Alerts

1 On the Report menu, point to **Alerts** and then click **Create or Modify Alerts**.

2 In the Create Alerts dialog box, select the alert you want to edit and click **Edit**.

Tip: Double-clicking an alert also lets you edit it.

3 Make the changes you want in the Edit Alert dialog box.

4 Click **OK** to save your changes.

Note: If the alert has already been triggered, editing it removes it from the Report Alerts dialog box.

Deleting Report Alerts

1 On the Report menu, point to **Alerts** and then click **Create or Modify Alerts**.

2 In the Create Alerts dialog box, select the alert you want to delete and click **Delete**.

The selected alert is removed from the Create Alerts dialog box.

Note: If the alert has already been triggered, deleting it removes it from the Report Alerts dialog box as well.

Viewing Report Alerts

You can view triggered Report Alerts by:

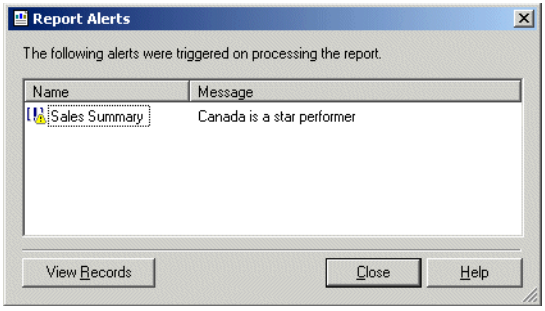
- Refreshing your report's data.
- Selecting Triggered Alerts from the Alerts submenu of the Report menu.

Note: If you want to view alerts when report data is refreshed, you must select "Display Alerts on Refresh" on the Reporting tab of the Options dialog box (this option is also available on the Report Options dialog box).

To view Report Alerts

1. On the Report menu, point to **Alerts** and then click **Triggered Alerts**.

   The Report Alerts dialog box appears.



2. Select the alert whose records you want to see.
3. Click **View Records**.

   A new report tab is opened showing the report record(s) that triggered the alert. If the record is hidden, the record's group is shown but drill down is not performed.

   Note: If you select more than one triggered alert before clicking the View Records button, the results are generated by performing a Boolean AND operation on the selected alerts.
4. To return to the Report Alerts dialog box, click the **Preview** tab.
5. Click **Close** to remove the Report Alerts dialog box.

Referring to Report Alerts in formulas

Alerts can be referred to in formulas. Any formula that references an alert becomes a print time formula.

The following functions are available:

- IsAlertEnabled( )
- IsAlertTriggered( )
- AlertMessage( )

These functions behave the same as alerts created in the Create Alerts dialog box:

- IsAlertTriggered("AlertName") is true only for the records on which the alert is triggered
- AlertMessage("AlertName") displays the message for a record when the alert is true.

Because alerts are not field objects (you cannot drop them into the report), alerts are represented differently from report fields in the Formula Workshop. In the Functions tree, you'll see an Alerts heading where the functions themselves appear. Available alerts are listed by name below that.

Creating Reports from Excel and Access

20

This chapter describes the Add-Ins for Microsoft Excel and Microsoft Access. It explains how you create reports from an Access table or query and an Excel spreadsheet with the Crystal Report Wizard.

Overview of the Crystal Reports Add-Ins

Crystal Reports takes advantage of Microsoft's Add-In technology by offering a report creation Wizard you can use with Microsoft Excel and Microsoft Access. The Crystal Report Wizard offers familiar report design capabilities directly within your Microsoft Office applications—you can create a Crystal report from your data without leaving Excel or Access.

The method for working with the Add-Ins is similar in both Excel and Access:

- Open Microsoft Excel or Microsoft Access (the Add-Ins are automatically installed when you install Crystal Reports).
- Select a spreadsheet (or selected cells), table, or query and launch the Wizard.
- Use the creation Wizard's familiar screen layout to create a report to your specifications.

Note: The Add-Ins work with only the Office 97 and Office 2000 versions of Excel and Access.

About the Microsoft Excel Add-In

Choose an entire data range in a spreadsheet, or select certain cells to create a Crystal report. The report you create is linked to your Excel spreadsheet and can be refreshed to show changes made to the spreadsheet data.

Note: Excel Add-In reports are updated when you refresh their data in Crystal Reports if you have:

- Added a row to the selected spreadsheet range (but not pre-appended or appended a row).
- Deleted a row from the selected spreadsheet range.
- Modified a cell within the selected spreadsheet range.

After you create a report using the Wizard, you can preview it using the default ActiveX viewer or you can launch Crystal Reports to modify it.

About the Microsoft Access Add-In

Choose a table or query to create a Crystal report. The report you create is linked to your Access table or query and can be refreshed to show changes made to the database.

Note:

- Access Add-In reports will not reflect database changes until you close the modified table or query in Access and refresh the report data in Crystal Reports.
- Access Add-In reports must be based on one table or query (but not a query with input parameters).

After you create a report using the Wizard, you can preview it using the default ActiveX viewer or you can launch Crystal Reports to modify it.

Working with the Microsoft Excel Add-In

When you install Crystal Reports 9, the Excel Add-In is added automatically. A Crystal Report Wizard 9 menu option is added to the Tools menu in Excel and a button is added to the Standard toolbar.

Note: If you are running Crystal Reports 9 in side-by-side mode with Crystal Reports 8.5, you will see two buttons on the toolbar in Excel, one for each version. The buttons look the same, but the tool tips identify the correct version of the Wizard. You can modify the toolbar in Excel to remove these buttons.

You can use the Crystal Report Wizard to create a report anytime you use Excel. The option remains available until you uncheck Crystal Report Wizard in the Add-Ins dialog box (under the Tools menu in Excel).

Note: The Excel Add-In file is placed by default in the `\Program Files\Crystal Decisions\Report Designer Component` directory.

Using the Crystal Report Wizard to create a report in Excel

The Crystal Report Wizard in Excel offers familiar Crystal layout and selection screens to help you create a report from your spreadsheet. You can select a range of cells, or an entire data range in a spreadsheet to use as your report content.

Note: When you create a report from an Excel spreadsheet, Crystal Reports creates a database file (.mdb). This file, the Excel spreadsheet file (.xls), and the report file itself (.rpt) cannot be moved or you'll lose the links between them and won't be able to refresh your report data later. If you want to change the spreadsheet data source for your report, use the Set Datasource Location option. For more information, search for this topic in the Crystal Reports Online Help.

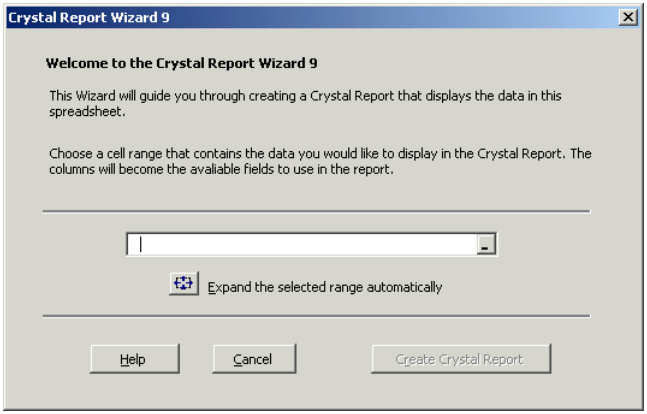
To create a report in Excel using the Crystal Report Wizard

1. On the **Tools** menu in Excel, choose **Crystal Report Wizard**.

   Tip: Another way to open the Crystal Report Wizard is to click the Crystal Report Wizard button.

   If your spreadsheet has unsaved data, Excel warns you to save before running the Wizard.



2 Type a data range in the cell range box, or click its button to choose a range directly from the spreadsheet.

Clicking the button minimizes the Wizard so you can click-drag to select a data range. To return to the Wizard, click the button on the minimized Wizard.

Note: To select an entire data range in the spreadsheet, click "Expand the selected range automatically."

3 With a data range selected, click **Create Crystal Report**.

The Crystal Report Wizard appears.

Note: As you design your report, the program creates a database file (.mdb). The database file, the spreadsheet file (.xls), and the resulting report file (.rpt) cannot be moved or you won't be able to refresh your report data later. If you want to change the spreadsheet data source for your report, use the Set Datasource Location option. For more information, search for this topic in the Crystal Reports Online Help.

4 Add the database fields you want to appear in your report to the **Fields to Display** list.

The arrow buttons on this dialog box enable you to move fields from one list to the other. Single arrows move only the selected field; double arrows move all fields at the same time.

5 Click **Next** to choose fields to group on.

6 Add the database fields you want to group on to the **Group By** box.

When a group field is selected, you can choose a sorting order from the Sort Order box.

7 Click **Next** to choose fields to summarize on.

8 Add the database fields you want to total on to the **Summarized Fields** list.

Summarized fields apply to the group specified in the For the Group box. You can use the same fields in each of several different groups.

When a summary field is selected, you can choose a summary type for it and add a grand total.

9 Click **Next** to sort groups on their summarized totals.

You can select Sort All Groups, or you can specify a Top or Bottom N.

10 Click **Next** to create a record selection for your report.

11 Add the database fields you'll use to filter your report to the **Filtered Fields** list.

Define your record selection by selecting from the box of limiting operators and entering the appropriate field value.

12 Click **Next** to select a report style.

13 Click **Next** to enter a title and saving location for your report.

14 Choose a method for viewing your report:

- Preview (read only)
- Edit with the Crystal Report Designer
- View Later

The Preview option opens the report in the ActiveX viewer. You can view, print, and refresh the report in the viewer.

The Edit option opens the report in the Crystal Reports. You can view, print, refresh, and modify the report while using all the capabilities of the Report Designer.

The View Later option saves the report with the name you gave it to the directory you specified.

15 Click **Finish**.

Now that your report has been created, you can use it and modify it as you would any other report in Crystal Reports. It remains linked to your Excel spreadsheet so if you refresh the report after altering the spreadsheet, your changed data is displayed (see "About the Microsoft Excel Add-In" on page 322).

Working with the Microsoft Access Add-In

When you install Crystal Reports 9, the Access Add-In is added automatically. A menu option called Crystal Report 9 Wizard is added to the Add-Ins submenu of the Tools menu in Access.

Note: Installation of the Access Add-In is not automatic for Office 97. For instructions on how to install the Crystal Report Wizard in Access 97, see the Crystal Reports Online Help.

You can use the Crystal Report Wizard to create a report anytime you use Access. The option remains available until you uninstall Crystal Report Wizard in the Add-In Manager dialog box (under the Tools menu in Access).

Note: The Access Add-In files are placed by default in the `\Program Files\Crystal Decisions\Report Designer Component` directory.

Using the Crystal Report Wizard to create a report in Access

The Crystal Report Wizard offers familiar Crystal layout and selection screens to help you create a report from your table or query.

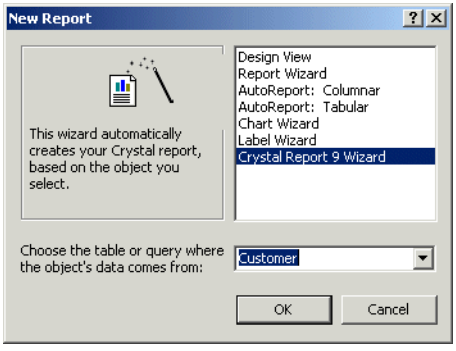
To create a report in Access using the Crystal Report Wizard

1. On the **Tools** menu in Access, point to **Add-Ins** and then click **Crystal Report 9 Wizard**.

   If your database table or query is open, Access warns you to save before running the wizard.

   The Choose a Table or Query dialog box appears if you haven't already highlighted a table or query in the Database dialog box or the New Report dialog box.

   Another way to open the Crystal Report Wizard is to select the Reports option in the Database dialog box, click New, select the Crystal Report 9 Wizard, and then click OK. If you use this method, you can select your table or query on the New Report dialog box, and go to step 3.



2. From the Choose a Table or Query dialog box, select the table or query you want to use and click **OK**.

   The Crystal Report Wizard appears.

3. Add the database fields you want to appear in your report to the **Fields to Display** list.

   The arrow buttons on this dialog box enable you to move fields from one list to the other. Single arrows move only the selected field; double arrows move all fields at the same time.

4. Click **Next** to choose fields to group on.
5. Add the database fields you want to group on to the **Group By** box.

   When a group field is selected, you can choose a sorting order from the Sort Order box.
6. Click **Next** to choose fields to summarize on.
7. Add the database fields you want to total on to the **Summarized Fields** list.

   Summarized fields apply to the group specified in the For the Group box. You can use the same fields in each of several different groups.

   When a summary field is selected, you can choose a summary type for it and add a grand total.
8. Click **Next** to sort groups on their summarized totals.

   You can select Sort All Groups, or you can specify a Top or Bottom N.
9. Click **Next** to create a record selection for your report.
10. Add the database fields you'll use to filter your report to the **Filtered Fields** list.

    Define your record selection by selecting from the box of limiting operators and entering the appropriate field value.
11. Click **Next** to select a report style.
12. Click **Next** to enter a title and saving location for your report.
13. Choose a method for viewing your report:
    - Preview (read only)
    - Edit with the Report Designer
    - View Later

    The Preview option opens the report in the ActiveX viewer. You can view, print, and refresh the report in the viewer.

    The Edit option opens the report in the Crystal Reports. You can view, print, refresh, and modify the report while using all the capabilities of the Report Designer.

    The View Later option saves the report with the name you gave it to the directory you specified.

Now that your report has been created, you can use it and modify it as you would any other report in Crystal Reports.

Using Formulas

21

This chapter explains the basics of formulas and introduces you to the Formula Workshop in order for you to begin to create formulas.

Formulas overview

In many cases, the data needed for a report already exists in database table fields. For example, to prepare an order list you would place the appropriate fields on the report.

Sometimes, however, you need to put data on the report that does not exist in any of the data fields. In such cases, you need to create a formula. For example, to calculate the number of days it takes to process each order, you need a formula that determines the number of days between the order date and the ship date. Crystal Reports makes it easy for you to create such a formula.

What's new in the formula language

Many new features have been added to the formula language for this version of Crystal Reports.

Custom functions

Custom functions are procedures you write that are independent of the report they are being used in. They provide a way for you to share and reuse formula logic across reports. This makes it easier and less time consuming for you and your users to create usable reports.

Custom functions can be written in Crystal or Basic syntax. A report can contain custom functions of both syntax types. To use a custom function in your report, you'll need to call it from a formula you're using. The syntax of the formula calling the custom function is not relevant when determining the syntax of the custom function you are calling.

For more information see Working with custom functions in the online help.

Custom functions and the Crystal Repository

Custom functions can be shared across reports using the Crystal Repository. You can save a custom function to the repository and then copy that custom function into a different report. You can also keep your custom functions connected to the repository and automatically update your report with the latest version after you modify them.

Note: The Crystal Repository is not available in all version of Crystal Reports.

Sample custom functions

Crystal Reports ships with sample custom functions that can be found in the Crystal Repository and in the sample report Custom Functions.rpt. Also see Custom function examples in the online help.

Increased limits

Formulas can now work with memo fields, and they can now process strings up to 64K in length, as opposed to a string-size limit of 255 bytes in previous versions.

The Formula Workshop and enhancements to the Formula Editor

A new component called the Formula Workshop has been added to this version of Crystal Reports. Use the Formula Workshop as a central location from which you can create, modify, and delete formulas. There also have been many enhancements made to the Formula Editor. For more information see "Working with the Formula Workshop" on page 336.

Keyword auto complete

While in the Formula Editor, if you use the key combination CTRL+Space, a list of the functions you are most likely to use appears.

Call stack for debugging evaluation time errors

If you encounter an error in a formula at evaluation time (that is, when you preview your report), the Formula Editor appears with a call stack in the Workshop Tree that shows the current values of all variables used in the formula. An error message appears, and the cursor is placed in your formula where the error occurred. For more information see "Debugging evaluation time errors" on page 344.

Workshop Tree

In the Formula Workshop, there is a tree, which initially appears on the left-hand side, that lists all the formulas, functions, and fields in your report. The tree can be relocated to a different location in the workshop.

Formula Expert

When creating a new report formula, you can use either the Formula Editor (as in past versions) or the Formula Expert. The Formula Expert lets you use an existing custom function as a template for creating a new formula. The custom function supplies the underlying business logic, and you supply the report fields or constant values that tie the logic to your particular reporting situation. For more information see "Creating a formula in the Formula Expert" on page 339.

Extract Custom Function from Formula dialog box

This dialog box lets you extract the logic from report formulas in old reports to create a new custom function. These custom functions can then be reused in a variety of reports, and in a variety of situations. The Extract Custom Function from Formula dialog box is a migration tool to help people extract the business logic used in their old reports in a way that can be shared and reused easily. For more information, see Using the Extract Custom Function from Formula dialog box in the online help.

New functions

New String functions

- Roman—takes a number and returns the Roman numeral version of the number.
- ChrW—takes a Unicode number and returns the matching character.

- AscW—takes a character or a string and returns the Unicode value of that character or the first letter in the string.
- ProperCase—takes a string and returns that string with the first letter of each word capitalized.

New Range functions

- HasLowerBound—returns a Boolean value whose value depends on if the range passed in has a lower bound or not.
- HasUpperBound—returns a Boolean value whose value depends on if the range passed in has an upper bound or not.
- IncludesLowerBound—returns a Boolean value whose value depends on if the range passed in includes its lower bound or not.
- IncludesUpperBound—returns a Boolean value whose value depends on if the range passed in includes its upper bound or not.

Other new functions

- Option Loop statement—allows you to set the maximum number of loop condition evaluations per evaluation of a formula, when the default maximum of 100,000 is not sufficient. Available in both Crystal and Basic syntax.
- DrillDownGroupLevel—returns a number that indicates the group level of the current drill down view. This is a print-state function and is available in both Crystal and Basic syntax.

New Financial functions

There are 42 new financial functions in this version of Crystal Reports, see Financial Functions in the online help for a complete list of the financial functions available.

Typical uses for formulas

There are many uses for formulas. If you have a need for specialized data manipulation, you can do it with a formula.

Creating calculated fields to add to your report

To calculate a price discounted 15%:

Crystal syntax example:

```
{Orders_Detail.Unit Price}*.85
```

Basic syntax example:

```
formula = {Orders_Detail.Unit Price}*.85
```

Formatting text on a report

To change all the values in the Customer Name field to uppercase:

Crystal syntax example:

```
UpperCase ({Customer.Customer Name})
```

Basic syntax example:

```
formula = UCase ({Customer.Customer Name})
```

Pulling out a portion, or portions, of a text string

To extract the first letter of the customer name:

Crystal syntax example:

```
{Customer.Customer Name} [1]
```

Basic syntax example:

```
formula = {Customer.Customer Name} (1)
```

Extracting parts of a date

To determine what month an order was placed:

Crystal syntax example:

```
Month ({Orders.Order Date})
```

Basic syntax example:

```
formula = Month ({Orders.Order Date})
```

Using a custom function

To convert $500 from U.S. currency to Canadian:

Crystal syntax example:

```
cdConvertUSToCanadian (500)
```

Basic syntax example:

```
formula = cdConvertUSToCanadian (500)
```

Formula components and syntax

Formulas contain two critical parts: the components and the syntax. The components are the pieces that you add to create a formula while the syntax is the rules that you follow to organize the components.

Formula components

Creating a formula in Crystal Reports is like creating one in any spreadsheet application. You can use any of the following components in your formula:

Fields

Example: {customer.CUSTOMER LAST NAME}, {customer.LAST YEAR'S SALES}

Numbers

Example: 1, 2, 3.1416

Text

Example: “Quantity”, “:”, “your text”

Operators

Example: + (add), / (divide), -x (negate)

Operators are actions you can use in your formulas.

Functions

Example: Round (x), Trim (x)

Functions perform calculations such as average, sum, and count. All functions available are listed with their arguments and are arranged by their use.

Custom functions

Example: cdFirstDayofMonth, cdStatutoryHolidays

Custom functions provide a way to share and reuse formula logic. They can be stored in the Crystal Repository and then added to a report. Once in the report, custom functions can be used in the Formula Expert when creating formulas.

Control Structures

Example: “If” and “Select”, “For” loops

Group field values

Example: Average (fld, condFld), Sum (fld, condFld, “condition”)

Group field values summarize a group. For example, you could use group field values to find the percentage of the grand total contributed by each group.

Other formulas

Example: {@GrossProfit}, {@QUOTA}

Formula syntax

Syntax rules are used to create correct formula. Some basic rules are:

- Enclose text strings in quotation marks.
- Enclose arguments in parentheses (where applicable).
- Referenced formulas are identified with a leading @ sign.

Crystal and Basic syntax

When creating formulas, you have the option of using either Crystal or Basic syntax. Almost any formula written with one syntax can be written with the other. Reports can contain formulas that use Basic syntax as well as formulas that use Crystal syntax.

Crystal syntax is the formula language included in all versions of Crystal Reports.

If you are familiar with Microsoft Visual Basic or other versions of Basic, then Basic syntax may be more familiar to you. In general, Basic syntax is modeled on Visual Basic except that it has specific extensions to handle reporting.

If you are already comfortable with Crystal syntax, you can continue to use it, and benefit from the new functions, operators and control structures inspired by Visual Basic.

Note:

- Record selection and group selection formulas cannot be written in Basic syntax.
- Report processing is not slowed down by using Basic syntax. Reports using Basic syntax formulas can run on any machine that Crystal Reports runs on.
- Using Basic syntax formulas does not require distributing any additional files with your reports.

Related topics

To learn about Basic syntax, see Creating Formulas with Basic syntax in the online help.

To learn about Crystal syntax, see Creating Formulas with Crystal syntax in the online help.

Specifying formulas

There are several different kinds of formulas in Crystal Reports: report, formatting, selection, search, running total condition, and alert formulas. The majority of formulas in a report are report formulas and conditional formatting formulas.

Report formulas

Report formulas are formulas that you create to stand alone in a report. For example, a formula that calculates the days between the order date and the shipping date is a report formula.

Conditional formatting formulas

Formatting formulas change the layout and design of a report, as well as the appearance of text, database fields, objects, or entire report sections. You format text through the Format Editor. If you need to create a formatting formula, you access the Formula Workshop from the Format Editor. See “Working with conditional formatting” on page 208.

Selection formulas

Selection formulas specify and limit the records and groups that appear in a report. You can either enter these formulas directly or specify the selection using the Select Expert. Crystal Reports then generates the record selection and group selection formula. You have the option to manually edit these formulas, but you must use Crystal syntax. See “Selecting records” on page 108.

Search formulas

Search formulas help you locate data in your report. Like selection formulas, you normally do not enter these formulas directly, but instead specify the search criteria using the Search Expert. Crystal Reports generates the formula. You have the option to manually edit these formulas, but you must use Crystal syntax.

Note: If you already know Basic syntax, you need to know only a small amount of Crystal syntax to modify most selection and search formulas.

Running Total condition formulas

Running Total condition formulas let you define the condition upon which your running total will be evaluated or reset. See "Creating conditional running totals" on page 151.

Alerting formulas

Alerting formulas help you define conditions and messages for report alerts. See "About Report Alerts" on page 316.

Working with the Formula Workshop

You can create most kinds of formulas in the Formula Workshop. The workshop consists of a toolbar, a tree that lists the types of formulas you can create or modify, and an area for defining the formula itself.

Note: Search formulas and Running Total condition formulas are not included in the Formula Workshop. You must use the Search Expert or the Create (or Edit) Running Total Field dialog box to create and maintain these formulas.

Accessing the Formula Workshop

There are many ways to access the Formula Workshop. You see it when you add new Formula Fields, when you define selection formulas, when you work with custom functions, and so on.

You can open the Formula Workshop by itself before you begin adding specific kinds of formulas.

To access the Formula Workshop

1. On the **Report** menu, click **Formula Workshop**.

   Tip: Another way to do this is to click the Formula Workshop button on the Expert Tools toolbar.

   The Formula Workshop appears.

2. Click **New** and select the kind of formula you want to create from the list that appears.

   Tip: You can also select the appropriate folder in the Workshop Tree, and then click the New button.

   The appropriate editor or dialog box appears.

Workshop Tree

The Workshop Tree contains folders for each type of formula you can create in Crystal Reports. It also contains folders for custom functions and SQL Expressions. If the workshop appears as the result of using a specific command (for example, you've selected the Record command on the Selection Formulas submenu), the appropriate folder in the tree is selected, and the appropriate version of the Formula Editor appears.

Expand any folder in the tree to see the formulas that already exist. New formulas can be added, and existing formulas can be edited or deleted as needed.

Tip: The Workshop Tree can be docked. By default, it appears docked on the left-hand side of the Formula Workshop, but you can manually dock it on the right-hand side. In free-floating mode, the Workshop Tree can be dragged to any location in the workshop.

Working with the Formula Editor

The Formula Editor is a component of the Formula Workshop. Use the Formula Editor to create and modify the content of formulas.

Understanding the sections of the Formula Editor

The Formula Editor contains four main windows.

Window	Description of contents
Report Fields	Report fields contain all database fields accessible for your report. They also contain any formulas or groups already created for the report.
Functions	Functions are prebuilt procedures that return values. They perform calculations such as average, sum, count, sin, trim, and uppercase. Custom functions are also listed in this window.
Operators	Operators are the "action verbs" you use in formulas. They describe an operation or an action to take place between two or more values. Examples of operators: add, subtract, less than, and greater than.
Formula text window	Area where you create a formula.

Choosing the syntax

The top right corner of the Formula Editor contains the drop-down list where you choose either Crystal or Basic syntax for the formula you are creating.

Note: Changing the syntax from Crystal syntax to Basic syntax or vice versa will change the list of functions in the Functions window as well as the list of operators in the Operators window. The functions and operators differ from syntax to syntax.

The available report fields remain the same since the report fields are available to each syntax.

Setting the default syntax

When you open the Formula Editor, Crystal syntax appears as the syntax default. If you want to change the syntax default, select Options from the File menu, then click the Reporting tab. Choose the preferred syntax from the Formula Language drop-down list and click OK. When you access the Formula Editor, the syntax you selected appears as the default.

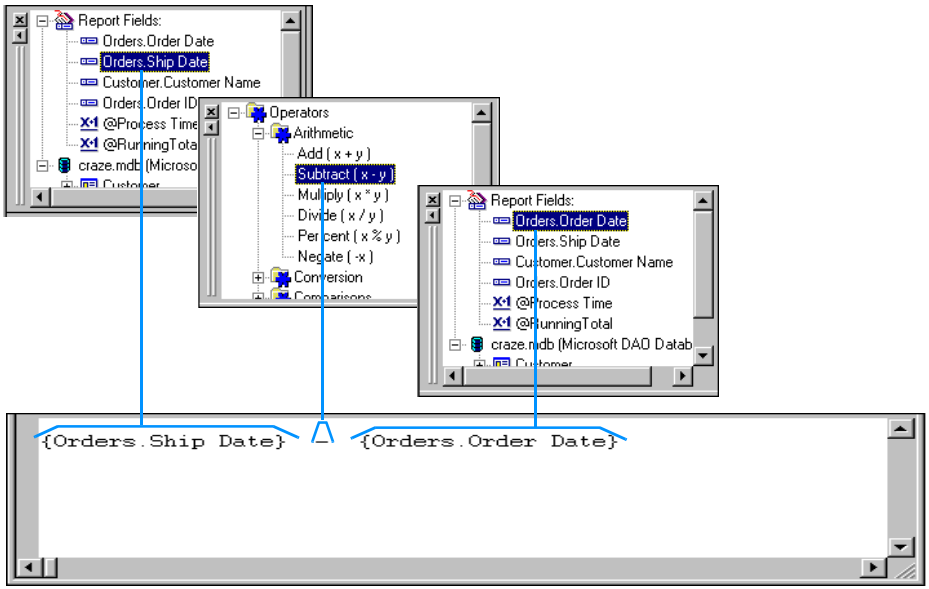
Entering formula components

The Report Fields, Functions, and Operators tree at the top of the Formula Editor contain the primary formula components. Double-click any component from these trees to add this component to your formula.

For example, if you set the syntax to Basic syntax and double-click the Operators > Control Structures > Multi-Line If in the Operators tree, the following text is transferred to the Formula text window with the cursor between the If and Then:

```
If | Then
ElseIf Then
Else
End If
```

The above text helps you organize the parts needed to write your formula.



Creating and modifying formulas

Creating a formula and inserting it into a report

1 On the **View** menu, click **Field Explorer**.

2 In the Field Explorer dialog box, select **Formula Fields** and click **New**.

3 In the Formula Name dialog box, enter the name you want to identify the formula.

4 Click **Use Editor**.

The Formula Workshop appears with the Formula Editor active.

5 In the Formula Editor, choose either Crystal or Basic syntax.

If you are unsure which syntax to choose see "Formula syntax" on page 334.

6 Enter the formula by typing in the components or selecting them from the component trees.

Tip: Ctrl+Space will bring up a list of the available functions. If you've already started typing it will bring up a list of keywords that are possible matches for what you've already typed.

7 Click **Check** to identify any errors in the formula.

8 Fix any syntax errors the Formula Checker identifies.

9 When the formula has the correct syntax, click **Close** on the Formula Workshop toolbar.

10 When prompted, click **Yes** to save your formula.

11 Select the new formula in the Field Explorer dialog box, and drag it to where you want it to appear on your report.

Note: A formula that is placed on a report is indicated by @ (for example, @ProcessTime) on the Design tab.

Creating a formula in the Formula Expert

The Formula Expert is a component of the Formula Workshop. Use the Formula Expert to create and modify formulas based on custom functions.

Note: To learn about the Formula Expert's user interface, see Formula Expert in the online help.

To create a formula in the Formula Expert

Note: Before you begin this procedure, ensure you have a custom function in your report or in the Crystal Repository you have access to.

1 On the **Report** menu, click **Formula Workshop**.

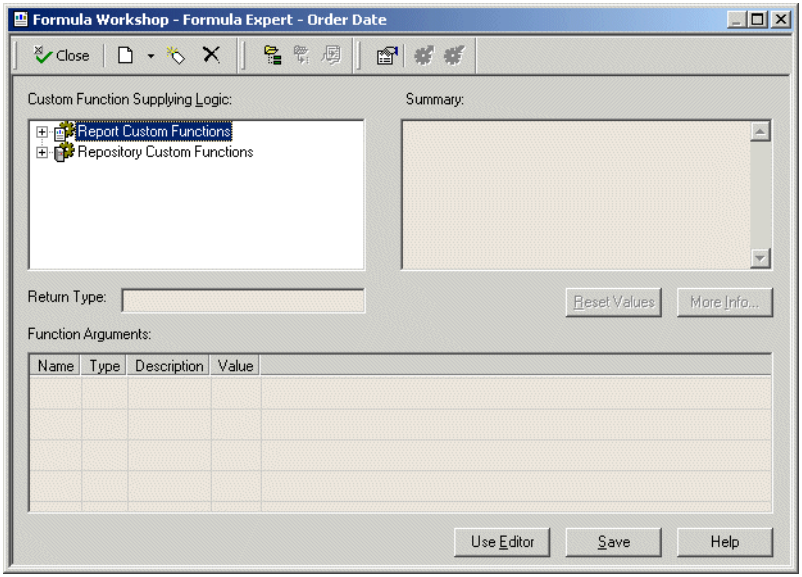
Tip: Another way to do this is to click the Formula Workshop button on the Expert Tools toolbar.

2 Select **Formula Fields** in the Workshop Tree and click **New**.

3 In the Formula Name dialog box, enter the name you want to identify the formula.

4 Click **Use Expert**.

The Formula Expert appears



5 In the Custom Function Supplying Logic area, choose the custom function you want to base your formula on.

You can choose a Report Custom Function (a custom function that exists in the current report) or a Repository Custom Function (a custom function that is stored in the repository).

Note: If you select a Repository Custom Function, that custom function is added to the current report. If that custom function requires other custom functions from the repository, they can be added as well.

6 In the Function Arguments area, specify a value for each argument in the appropriate **Value** field.

You can enter constant values directly, or you can select predefined values or report fields from the associated list.

7 Click **Save** to save the formula to the Formula Fields folder of the Formula Workshop.

You can now use this formula in your report just as you would use a formula you created in the Formula Editor.

Editing formulas

1. On the **View** menu, click **Field Explorer**.
   The Field Explorer dialog box appears.
2. Right-click the formula you want to edit and choose **Edit**.
   The Formula Workshop appears with the Formula Editor active.
3. In the Formula Editor, edit the formula.

4. Click **Check** to identify any errors in the formula.
5. Fix any syntax errors the Formula Checker identifies.

6. When the formula has the correct syntax, click **Close** on the Formula Workshop toolbar.
7. When prompted, click **Yes** to save the changes you made to your formula.

Searching and replacing text

1. On the **View** menu, click **Field Explorer**.
   The Field Explorer dialog box appears.
2. Right-click the formula you want to edit and choose **Edit**.
   The Formula Workshop appears with the Formula Editor active.

3. In the Formula Editor, click **Find or Replace** to open a Find dialog box.
   From this dialog box, you can search and replace text within the Formula text box.
   Tip: Be sure to select the area you want to search from the Search list.
4. Click the **Mark All** button to mark all occurrences of the search text.
5. Click the **Replace All** button to replace all occurrences of the search text with the contents of the Replace with text box.

You can also search (but not replace) within any of the Formula Editor trees (use the Search options to specify which list boxes you are searching). The Mark All, Replace, and Replace All buttons become inactive when you specify a search within a list box.

Copying formulas from online help

Since the formulas you develop using the Formula Editor are text, you can copy useful online formulas directly into the Formula Editor and then modify them to fit your needs.

To copy formulas from online help

1. On the **Help** menu, click **Crystal Reports Help**.
   The Crystal Reports online help appears.

2 Click the **Index** tab.

3 Enter `formulas` in the keyword field and click **Display**.

4 Scroll through the formula topics until you locate the formula you want to copy.

5 Highlight the formula, right-click, and choose **Copy** from the menu.
Windows places a copy of the selected text on the Clipboard.

6 Return to Crystal Reports, choose **View** from the main menu and select **Field Explorer**.
The Field Explorer dialog box appears.

7 Select **Formula Fields** and click the **New** button.
The Formula Name dialog box appears.

8 Enter the name you want to identify the formula and click **Use Editor**.
The Formula Workshop appears with the Formula Editor active.

9 Place the insertion point where you want the text to appear in the **Formula** text box of the Formula Editor and press Ctrl+V to paste the text from the Clipboard.

10 Modify the formula by changing the fields, formulas, group fields, conditional statements, and text strings as necessary for use with the data in the new report.

11 When the formula has the correct syntax, click **Close** on the Formula Workshop toolbar.

12 When prompted, click **Yes** to save your formula.

Copying formulas from one report to another

You may want to copy a formula created in one report for use in another report. Copy the text formula from one report to another via the Clipboard.

To copy a formula from one report to another

1 Select the formula field you want to copy in the report.

2 On the **Edit** menu, click **Copy**.

3 Open the report you want to copy the formula to.

4 Choose **Paste** from the Edit menu.

5 When the program displays the object frame, drag the formula to the new location.

6 To make changes to the formula, right-click the formula and choose **Edit Formula** from the shortcut menu.
The Formula Workshop appears with the Formula Editor active.

7 Delete the old values and type in the new values, or select them from the Fields, Functions, and/or Operators tree.

8 Click **Close** on the Formula Workshop toolbar when finished.

Key points for editing a copy of a formula

When making changes, use the following points as a guide:

- All fields, formulas, and group fields referenced in the formula copy must actually exist in the new report. This means that any database referenced in the original formula (or a database with the same structure, field names, and alias) must be active in the new report.
- If such a database is not active, you must change the field, formula, and group field references in the formula copy to correspond to elements in your new report.
- If the formula contains conditional elements, make certain that the conditions apply to the data in the new report. For example, if the formula in your old report performed an action when the quantity was greater than 100, make sure that the greater than 100 condition makes sense in the new formula. When modifying a formula, you may find that greater than 10 or greater than 2000 makes more sense with your new data.
- If you are using the formula with new data, and if your report contains statements similar to the following:

  ```
  If {file.FIELD} = "text string"
  ```

  Make sure that the text strings used in the formula match values that actually exist in the new data.

Deleting formulas

When a formula is created and added to a report, the Report Designer:

- Stores the specification for creating the formula, using the name you assigned to it.
- Places a working copy of that formula at the point you specify in the report. A working copy is any occurrence of the formula in the report.

In order to completely delete formulas, you must delete the specification and all working copies of the formula.

Note: You cannot delete the specification without deleting all working copies of the formula.

Removing the working formula from your report

1 Right-click the formula you want to delete from the report.

2 Select **Delete**.

Note: Even after the working copies of a formula have been deleted from the report, the formula specification remains unchanged. The specification is listed in the Field Explorer dialog box. It is available if you wish to enter the formula in the report again.

Deleting the formula specification

1. Choose **View** from the main menu and select **Field Explorer**.
   The Field Explorer dialog box appears.
2. Right-click the formula you want to delete and choose **Delete**.

Note: A dialog box appears if this formula is currently in use in a report. If you delete this formula, you will delete all references of it in reports. Click Yes to delete.

Debugging formulas

For help on debugging formulas that occur when you save your formula, see the Debugging tutorial.

For help on debugging evaluation time errors that invoke the formula editor stack list and assisted debugging, see Debugging evaluation time errors.

Debugging evaluation time errors

When the Formula Workshop is being displayed as a result of an evaluation time error, the Workshop Tree will contain a call stack. The root of the tree provides a description of the error which occurred. The nodes in the tree provide the names of the custom functions and / or formulas which were being evaluated when the error occurred. The custom function / formula at the top of the call stack is where the error was detected. The custom function / formula next in the stack has invoked the custom function / formula above it in the stack. If you select a custom function / formula node in the tree, the text of the custom function / formula will be displayed in the editor window and the text of the expression being evaluated when the error occurred will be highlighted. If you expand a custom function / formula node in the tree, the variables being used in the custom function / formula will be shown along with the value they had at the time the error occurred.

Example of an evaluation time error

Using the sample report “Custom Functions.rpt”, create a new formula that divides 1 by the result of the @Calendar Days Between function. The function would look like this in Basic syntax:

```
formula = 1/{@Calendar Days Between}
```

It would look like this in Crystal syntax:

```
1/{@Calendar Days Between}
```

Insert this formula into the details section of the report and preview it. You’ll get a division by zero error and the formula editor will be invoked with the call stack on the left hand side.

Debugging tutorial

Follow the example below to learn the necessary steps for debugging a formula. After completing this exercise, use the same principles to debug your own formulas.

About this tutorial

- This tutorial uses the Xtreme.mdb sample database.
- This tutorial uses Crystal syntax.
- The following formula is the formula you will test for errors:

```
If ({customer.CUSTOMER NAME} [1 to 2 = "Bi" and
ToText({customer,CUSTOMER ID}) [1] = "6") Or
({customer.CUSTOMER NAME} [1] = 'Ro" and
ToText({customer.CUSTOMER ID}) [1] = "5")
       "PREFERRED CUSTOMER"
Else
       "DOES NOT FIT CRITERIA"
```

If correct, this formula should pick out all customers whose names begin with “Bi” and whose customer IDs begin with “6” as well as those companies whose names begin with “Ro” and whose customer IDs begin with “5”. When printing the field, those selections will read “PREFERRED CUSTOMER”, while the rest will read “DOES NOT FIT CRITERIA”.

You will now break down the formula to check and see that each condition of the formula is working individually.

Formula1

1. To get started, create a report using the Customer table in **Xtreme.mdb** and place the following fields from left to right in the Details section:

   ```
   {customer.CUSTOMER ID}
   {customer.CUSTOMER NAME}
   ```

   To test each portion of the formula, you will place a new formula field next to these two fields in the report.

2. Create a new formula called Formula1.

3. Type the following in the **Formula text** box of the Formula Editor:

   ```
   If {customer.CUSTOMER NAME} [1 to 2 = "Bi" Then
         "TRUE"
   Else
         "FALSE"
   ```

4. Click **Check** to test for errors. You will receive the following error message:

   ```
   The ] is missing.
   ```

5. Correct the formula by inserting the missing “] “ after the 2.

6 Click **Check** again. You will receive the following message:

```
No errors found.
```

7 Click **Close** on the Formula Workshop toolbar.

8 Insert the corrected formula field to the right of the two data fields in the Details section of your report.

9 Click **Print Preview** on the Standard toolbar to check the values in the report and compare the fields to see if the field values returned by @Formula1 are correct.

You will find “TRUE” listed next to the customer names that begin with “Bi” and “FALSE” next to all the others. Now you will check the other portions of the formula. Create Formula2, Formula3, and Formula4, by following Steps 1 - 9, using the formulas specified below for each.

Insert each formula field on the same line of the Details section for easy comparison. Check each one for errors, fix as needed, and make sure the values returned are correct before moving on to Formula2.

Formula2

1 Create a new formula called Formula2.

2 Type the following in the **Formula text** box of the Formula Editor:

```
If ToText({customer,CUSTOMER ID}) [1] = "6" Then
      "TRUE"
Else
      "FALSE"
```

3 Click **Check** to test for errors. You will receive the following error message:

```
This field name is not known.
```

4 Correct the formula by replacing the comma (,) in the field name with a period (.).

5 Click **Check** again. The formula should now be error-free.

6 Place the formula to the right of the @Formula1 field.

7 Click **Print Preview** on the Standard toolbar to check the values in the report and compare the fields to see if the field values returned by @Formula2 are correct.

You should see “TRUE” next to all customer numbers that begin with 6 and “FALSE” next to all customer numbers that do not begin with 6.

Formula3

1 Create a new formula called Formula3.

2 Type the following in the **Formula text** box of the Formula Editor:

```
If {customer.CUSTOMER NAME} [1 to 2] = 'Ro" Then
      "TRUE"
Else
      "FALSE"
```

3 Click **Check** to test for errors. You will receive the following error message:

```
The matching ' for this string is missing.
```

4 Correct the formula by changing the single quote (') before Ro to a double quote (").

5 Click **Check** again. The formula should now be error-free.

6 Place the formula to the right of the @Formula2 field.

7 Click **Print Preview** on the Standard toolbar to see the values in the report and compare the fields to check if the field values returned by @Formula3 are correct.

You should see “TRUE” next to all Customer names that begin with “Ro” and “FALSE” next to all Customer names that do not begin with “Ro”.

Formula4

1 Create a new formula called Formula4.

2 Type the following in the **Formula text** box of the Formula Editor:

```
If ToText({customer.CUSTOMER ID}) [1] = "5"
      "TRUE"
Else
      "FALSE"
```

3 Click **Check** to test for errors. You will receive the following error message:

```
The word 'then' is missing.
```

4 Correct the formula by typing in the word “Then” at the end of the first line after “5”.

5 Click **Check** again. The formula should now be error-free.

6 Place the formula to the right of the @Formula3 field.

7 Click **Print Preview** on the Standard toolbar to see the values in the report and compare the fields to check if the field values returned by @Formula4 are correct.

You should see “TRUE” next to all Customer IDs that begin with 5 and “FALSE” next to all Customer IDs that do not begin with 5.

Now that the formulas are error-free and the field values returned are correct, you will create a formula that links the separate components together. You will begin by linking the first two formulas (@Formula1 and @Formula2) and then you will add @Formula3 and @Formula4 to create the final formula @FinalFormula.

Formula1+2

1 Create a new formula called Formula1+2.

2 Type the following in the *Formula text* box of the Formula Editor:

```
If {customer.CUSTOMER NAME} [1 to 2] = "Bi" and
ToText({customer.CUSTOMER ID}) [1] = "6" Then
      "TRUE"
Else
      "FALSE"
```

3 Place the formula to the right of the @Formula4 field.

You should see “TRUE” next to each customer whose name begins with Bi and Id begins with 6, and “FALSE” next to all Customer IDs that do not meet this criteria.

If this formula is working correctly, you can create one last formula adding the code from @Formula3 and @Formula4.

FinalFormula

1 Create a new formula called FinalFormula.

2 Type the following in the Formula text box of the Formula Editor:

```
If ({customer.CUSTOMER NAME} [1 to 2] = "Bi" and
ToText({customer.CUSTOMER ID}) [1] = "6") or
({customer.CUSTOMER NAME} [1 to 2] = "Ro" and
ToText({customer.CUSTOMER ID}) [1] = "5") Then
      "PREFERRED CUSTOMER"
Else
      "DOESN'T FIT CRITERIA";
```

3 Place the formula where you want it to appear in the Details section of the report. You can now delete all other formula fields from the report. See “Deleting formulas” on page 343.

You can use this same process of condition-by-condition testing for any formulas as a means of systematically checking them.

Parameter Fields

22

This chapter explains what parameter fields are and how they can be applied to create a single report that can be used to access different types of data depending on the user's needs.

Parameter overview

Parameters prompt the user of a report to enter information. Think of a parameter as a question that the user needs to answer before the report is generated. The information users enter, or the way they respond, determines what appears in the report. For example, in a report used by salespeople, there might be a parameter that asks the user to choose a region. The report would return the results for the specific region, instead of returning the results for all of the regions.

By using parameter fields in formulas, selection formulas, and in the report itself, you can create a single report that you can modify whenever your needs change. Parameter fields can also be used in subreports.

Parameter field considerations

There are a number of things to keep in mind when working with parameter fields:

- Parameter fields support the following data types:
  - Boolean: Requires a yes/no or true/false answer.
    Example: Include planned budget numbers in the summary?
  - Currency: Requires a dollar amount.
    Example: Display customers with sales over XXXXX.
  - Date: Requires an answer in a date format.
    Example: Enter the start and end dates of the quarter.
  - DateTime: Requires both date and time.
    Example: Display statistics for 07/04/1999 between 1:00pm-2:00pm.
  - Number: Requires a numeric value.
    Example: Enter the customer identification number.
  - String: Requires a text answer.
    Example: Enter the region.
  - Time: Requires an answer using a time format
    Example: Display the total number of calls from 1:00pm-2:00pm.
- Parameter field prompting text can be up to four lines long with approximately 60-70 characters per line (depending on character width, up to the 254 character limit). Text over one line in length will automatically word wrap.
- You can create a pick list for the user to choose the parameter value rather than having them enter it manually.
- A parameter field does not have to be placed in a report in order to be used in a record or group selection formula. You create the parameter field and then enter it in your formula as you would any other field.

Creating a parameter field

Use the following steps to create a parameter that enables the user to specify a list of customers for a specific country. This procedure is made up of two sets of steps. The first is creating the parameter, and the second is using the Select Expert to incorporate the parameter.

To create a parameter field

1. Check to make sure your report is open in the **Design** tab.

   This procedure uses Group.rpt, a sample report that is included with the Crystal Reports software.

2. On the **View** menu, click **Field Explorer.**

   Tip: Another way to do this is to click the Field Explorer button on the Standard Toolbar.

   The Field Explorer dialog box appears.

3. Select **Parameter Fields** and click **New**.

   The Create Parameter Field dialog box appears.

4. Enter a name for the parameter in the **Name** field (up to 255 alphanumeric characters).

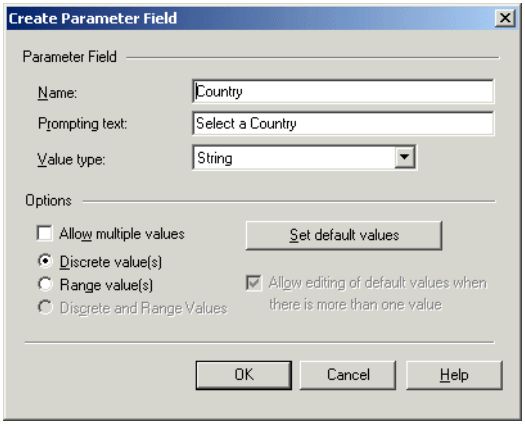
   This example uses Country.

5. Enter the desired prompting text in the **Prompting text** field (up to 255 alphanumeric characters).

   This is the text that appears in the Enter Parameter Values dialog box when the report is refreshed. This example uses “Select a Country.”

6. Select the appropriate **Value type** from the list.

   This example uses String.



Note: When creating a parameter whose Value type is either Date or DateTime, you can change the date format to suit your needs. For details, see "Changing your default field formats" on page 201.

7 Click **Set default values**.

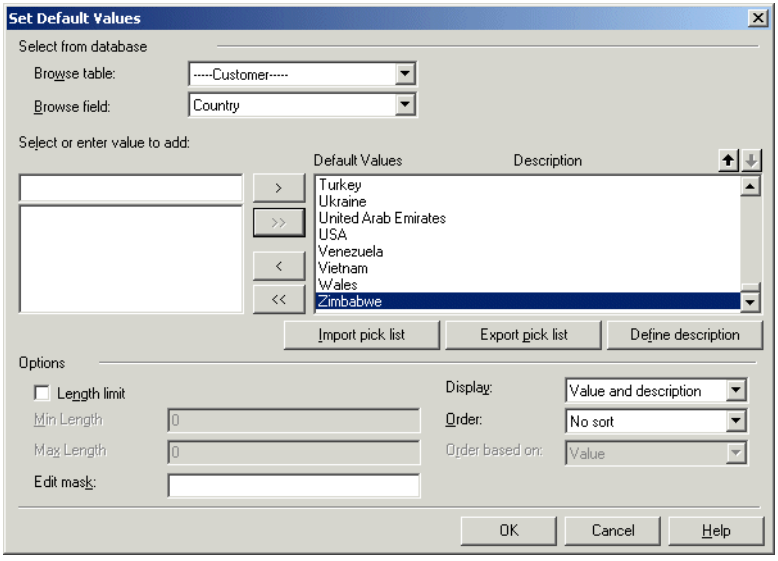
The Set Default Values dialog box appears.

8 Check to make sure the **Browse table** is set to **Customer**.

9 From the **Browse field** drop-down list, select **Country**.

10 Click >> to move all of the countries to the **Default Values** area.

This example will enable the user to choose from any of the countries. If you want to limit the selection, move only the countries you would like the user to choose from.



11 Click **OK**.

The Create Parameter Field dialog box appears.

12 Click **OK**.

The Field Explorer dialog box appears with the Country parameter selected.

13 Drag and drop the **Country** parameter into your report.

Note: If you don't want to see the parameter field you dropped in your report, place it in a section you can suppress, such as a report header or footer.

To incorporate the parameter into the selection

1 On the **Report** menu, click **Select Expert**.

Tip: Another way to do this is to click the Select Expert button on the Expert Tools toolbar.

The Choose Field dialog box appears.

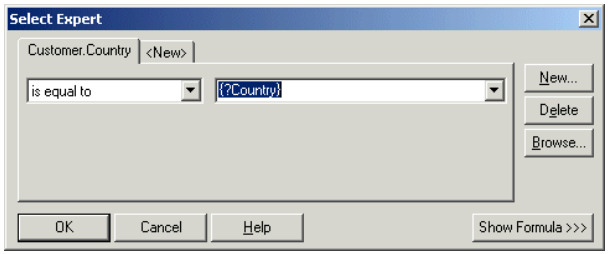
2 Select **Country** from the Customer table, then click **OK**.

The Select Expert appears.

3 Choose **is equal to** from the drop-down list.

4 Choose the parameter from the adjacent drop-down list.

This example uses {?Country}.



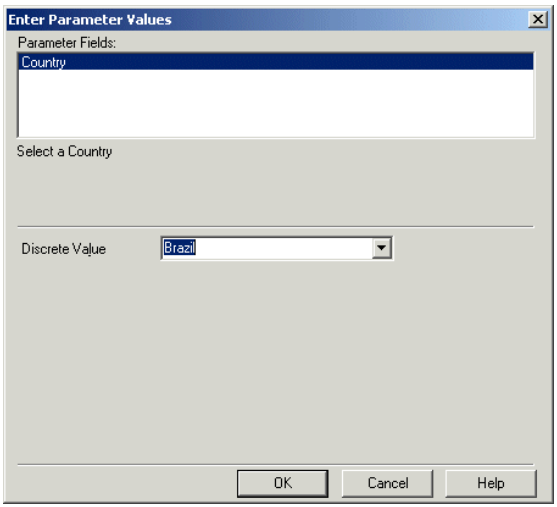
5 Click **OK**.

6 Click the **Preview** tab.

The Enter Parameter Values dialog box appears.

7 Select the country to base the report on.

This example uses Brazil.



8 Click **OK**.

The Change in Record Selection Formula Detected dialog box appears.

9 Click **Refresh Data**.

The report appears with the information for Brazil.

With parameter fields, you can create a single report that can be customized quickly to meet a variety of needs.

Deleting parameter fields

There are several methods for deleting parameters within a report. The type of parameter you are deleting determines the method you can use.

To delete a parameter that is not used in a formula

1 On the **View** menu, click **Field Explorer.**

The Field Explorer appears.

2 Expand the **Parameter Fields** folder and click the parameter you want to delete.

3 Click **Delete**.

A Crystal Reports dialog box appears confirming whether you want to delete the parameter.

4 Click **Yes**.

This parameter is removed from the Parameter Fields folder and from your report.

To delete a parameter used with the Select Expert

1 On the **Report** menu, click **Select Expert**.

Tip: Another way to do this is to click the Select Expert button on the Expert Tools toolbar.

The Select Expert dialog box appears.

2 Choose the tab whose selection criteria uses the parameter you want to delete.

3 Click **Delete**.

4 Click **OK** to close the Select Expert.

5 On the **View** menu, click **Field Explorer**.

6 Expand the **Parameter Fields** folder and click the parameter you want to delete.

7 Click **Delete**.

To delete a parameter that is used in a formula

1. On the **View** menu, click **Field Explorer**.

   The Field Explorer appears.
2. In the **Formula Fields** folder, select the formula that contains the parameter you want to delete.
3. Click **Edit** and delete the parameter field from the formula.

   Note: If the parameter is used in more than one formula, it must be deleted from each formula.
4. Close the Formula Workshop.
5. Expand the **Parameter Fields** folder and click the parameter you want to delete.
6. Click **Delete**.

Responding to parameter field prompts

Previewing a report for the first time

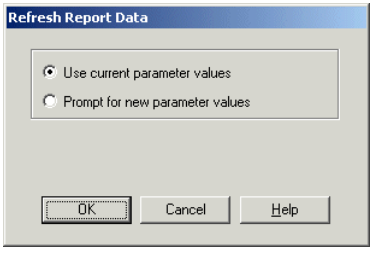
When you preview a report for the first time, the Enter Parameter Values dialog box appears, prompting you for a value.

- If you specified a default value when you created the parameter field, the program will use that value unless you specify a new one.
- If you did not specify a default value, the program will not refresh the data until you supply a new value.

Note: If the parameter is a string value type, without a default value, and nothing is entered into the discrete value field, an empty string appears.

Refreshing report data

When you refresh data from the Preview tab, the Refresh Report Data dialog box appears.



Select the “Use current parameter values” option to use the current parameter value.

Select the “Prompt for new parameter values” option to enter a new parameter value. When you select this option and click OK, the Enter Parameter Values dialog box appears.

- Enter String values exactly as they will appear in the field. If the parameter field allows multiple values, you can enter additional parameter values using the Add.
- Enter Boolean values using the following format: TRUE or FALSE.
- Enter Number values exactly as they will appear in the field.
- Enter Currency values exactly as they will appear in the field.
- Enter Date values to match the format used on-screen. If the format is unspecified, enter as Date (Year, Month, Day). For example, Date (1997, 5, 21). To access the calendar, click the drop-down arrow beside the date.
- Enter Time values to match the format used on-screen. If the format is unspecified, enter as Time (Hour, Minutes, Seconds AM / PM). For example, Time (4:32:12 PM). You can also select the unit of time and then use the up and down arrows to scroll through the numbers.
- Enter DateTime values using the following format: Date (Year, Month, Day), Time (Hour, Minutes, Seconds AM / PM). For example, Date (1997, 5, 21), Time (4:32:12 PM). You can also enter DateTime values by using the associated drop-down arrow to access the calendar, and the up and down arrows to scroll through the time.

  Note: The drop-down arrow to access the calendar for dates and the up and down arrows to scroll for times will only be available if you have the correct version (4.70 or later) of comctl32.dll.

To use a different value than the default displayed, type a new value in the text box, and click OK.

To use the default value, click OK.

- If the parameter field is range limited, then you can only enter values within a certain range. The range limit is specified in the Set Default Value dialog box when creating or editing a parameter.
- If the parameter is a string value type, you can limit the values the user is able to enter by using the length limit option or by using an edit mask to restrict the format.

The program now runs the report using the new value(s) you specified.

Advanced parameter features

There are a variety of ways that parameters can be used within a report. This section covers some of the advanced methods of using parameters:

- Creating a parameter with multiple values
- Applying conditional formatting using parameter fields
- Creating a report title using parameter fields
- Specifying single or ranges of values

- Incorporating a parameter into a formula
- Defining sort order using parameter fields
- Defining entry type and format using the Edit Mask

Creating a parameter with multiple values

1 On the **View** menu, click **Field Explorer**.

2 Select **Parameter Fields** and click **New**.

3 Enter a Name and Prompting text.

4 Select a Value type.

5 Select the **Allow multiple values** check box to enable more than one value to be entered into the parameter. Both discrete and range type values are allowed individually or in combination.

Note: For details on single or range values, see "Specifying single or ranges of values" on page 359.

6 Click **Set default values** to determine the type of entries that are allowed to be entered.

The Set Default Values dialog box appears.

7 From the **Browse table** drop-down list, select the table for the default values.

8 From the **Browse field** drop-down list, select the field for the default values.

The values associated with this table and field appear in the "Select or enter value to add" area.

9 Select the values listed, or create you own, and move them to the Default Values area using the arrow buttons.

10 Adjust the display, order, and length limit as required.

11 Click **OK**.

Applying conditional formatting using parameter fields

Parameter fields can be used to create conditional formatting formulas. You can customize these formulas whenever you refresh the report data. A conditional formatting formula could be used for color-flagging data that meets certain conditions. For example:

- Sales representatives who sell more than 10% over quota.
- Customers who have not ordered in the last quarter.
- Inventory items that have not had any movement in the last month.

If the conditions under which you flag these items never change, you do not need to use parameter fields. You can just use formulas (for text flags) or conditional formatting (for border flags). However, to change the conditions from report to report, you need to use parameter fields in formulas and conditional formatting formulas.

To apply conditional formatting using parameter fields

1 Create the parameter field of the data type you need for the formula.

2 Create the formula and use the parameter field in place of the fixed value you would normally use.

For example, to be prompted for all the customers whose last year's sales were over a certain value, and to print their names in red, select the Last Year's Sales field and click Format from the Expert Tools toolbar.

The Format Editor appears.

3 Click the Conditional formula button next to the Color property on the **Font** tab, and format the field using a conditional formatting formula like this:

```
If {customer.LAST YEAR'S SALES} > {?SalesTarget} Then
     Red
Else
     Black
```

Now, when you refresh the data, the program will prompt you for the value that triggers the color flag (known as the threshold value). It then runs the report and flags all the customers that had sales last year above the threshold figure. You can change the figure each time you run the report and the program will flag a different set of Customer Names.

For more information see "Conditional formatting functions" in the online help.

Creating a report title using parameter fields

Crystal Reports allows you to use parameter fields to create a report title that can be changed each time the report is refreshed.

To creating a report title using parameter fields

1 On the **View** menu, click **Field Explorer**.

The Field Explorer appears.

2 Select **Parameter Fields** and click **New**.

The Create Parameter Field dialog box appears.

3 Type a name for the parameter field in the **Name** field.

4 Type in any prompting text you wish in the **Prompting text** field.

5 Select String from the **Value type** drop-down list.

6 Click **Set default values**.

The Set Default Values dialog box appears.

7 To specify a default title, type the desired text into the **Select or enter value to add** field and click > to add the title to the **Default Values** area.

Note: Continue adding titles as required.

8 Click **OK**.

The Create Parameter Field dialog box appears.

9 Click **OK**.

The Field Explorer appears with the new parameter selected.

10 Place the parameter field in the Page Header section of the report to have the title appear on every page, or in the Report Header section if you want the title to appear on only the first page of the report.

Now, when you refresh the data, the program will prompt you for a report title. If you wish, you can change the title each time you run the report.

Specifying single or ranges of values

You can create parameters that require users to enter a single (discrete) value, or a range of values. If you then include these parameters in your report's record selection, you can help users find specific information. For example, in the case of a record selection parameter than uses a discrete value, a user might enter a single country name to see sales figures for only that country. In the case of a record selection parameter that uses a range of values, a user might enter a range of countries (for example, Canada and France) to see sales figures for all the countries in the range (in this case, Chili, China, Denmark, England, and so on).

To specify single or range values

1 Select the **Discrete value(s)** option or the **Range value(s)** option to specify whether the parameter field will accept a range of values.

- If you select **Discrete value(s)**, the parameter field will accept single values (rather than ranges of values).
- If you select **Range value(s)**, then when you are prompted for parameter values, you can enter a start value and an end value. For example, if you enter the values "5" and "10", the range is 5-10, and a report that uses this parameter for record selection will display all records with values between 5 and 10. This also works for string parameter fields. With a start value of "A" and an end value of "E", a report that uses this parameter for record selection will display all records within an alphabetical range of A-E.

Note: If the "Allow multiple values" and the "Discrete value(s)" options are selected, the parameter field will accept multiple single values. In this case, you can enter more than one value, but these values will be evaluated individually and will not be interpreted as a range. If the "Allow multiple values" and "Range value(s)" options are selected, the parameter field will accept multiple ranges.

2 If applicable, click **Set default values** to select the **Length limit** check box to designate the length of the field. If you select **Length limit**:

- For a Currency or Number parameter field, options for entering the "Min Value" and "Max Value" appear.

- For a DateTime parameter field, options for entering the “Start Date-time” and “End Date-time” appear.
- For a Date parameter field, options for entering the “Start Date” and “End Date” appear.
- For a Time parameter field, options for entering the “Start Time” and “End Time” appear.

Incorporating a parameter into a formula

1. On the **View** menu, click **Field Explorer**.

   The Field Explorer appears.

2. Select **Parameter Fields** and click **New**.

   The Create Parameter Field dialog box appears.

3. Create a parameter field and save it.

4. Select **Formula Fields** and click **New**.

   The Formula Name dialog box appears.

5. Enter the name of the formula, then click **Use Editor**.

6. Create a formula using the parameter field as you would any constant value. For example, rather than creating a formula that hard-codes the country name:

   {customer.COUNTRY} = “USA”

   Use a parameter field instead of “USA”.

   {customer.COUNTRY} = {?Country}

   To do this double-click the database field, press =, then double-click the parameter.

   Tip: Identify parameter fields easily by looking for (?).

7. Click **Close** on the Formula Workshop.

   The Field Explorer dialog box appears. The name of the formula you have just created is highlighted in the Formula list box.

8. Drag and drop the formula into the report.

9. Click **Refresh** to generate the report.

   A dialog box appears, prompting you for values.

Defining sort order using parameter fields

To set the sort order using parameter fields, you need to first create a formula that includes a parameter field and then sort based on that formula. For example, assume that you have a customer list report based on the Customer table. For each customer, you show the Customer Name, City, Region, Country, and Phone Number. You want to be able to sort the report by Country, by Region, or by City, depending on your needs at the time.

1. Create a parameter field and call it **SortField**.
2. In the **Prompting text** edit box, enter a prompt similar to this:

   ```
   Type R to sort by Region or C to sort by City; otherwise, data will be
   sorted by Country.
   ```

3. Select **String** from the **Value type** drop-down list.
4. You may want to limit the number of characters the user can type to one. To do this, click **Set default values**.

   The Set Default Values dialog box appears.
5. Select the **Length limit** check box and type the numeral 1 into the **Min Length** and **Max Length** text boxes and click **OK**.

   Now the parameter field will only accept single-character values. The field will accept "C" as a value, but not "City."
6. Create a formula similar to this and call it **Sort**:

   ```
   If {?SortField} = "C" Then {customer.CITY}
   Else
        If {?SortField} = "R" Then {customer.REGION}
        Else
             {customer.COUNTRY}
   ```

   This formula prompts for a value for the parameter field {?SortField}. If you enter "C", the formula will sort by the City field. If you enter "R" it will sort by the Region field. If you enter anything else, or do not enter anything at all, the formula will sort by the Country field.

   For more information see "If statements" in the online help.

7. Place the formula in the Report Header section of the report and select **Suppress (No Drill-Down)** in the Section Expert so that it does not print.

8. Click **Sort Records**.
9. Choose your formula and click **Add**.
10. Click **OK**.

Now when you run the report, the program will prompt you for a sort field, the formula will return a value based on your selection, and the sort facility will use that value as your sort field.

Defining entry type and format using the Edit Mask

1. On the **View** menu, click **Field Explorer**.

2. Select **Parameter Fields** and click **New**.

   The Create Parameter Field dialog box appears.
3. Enter the Name, Prompting text, and Value type.
4. Click **Set default values**.

For a string parameter field that you are setting the default values for, you can choose to enter an Edit Mask in the Edit Mask field, rather than specifying a range. An Edit Mask can be any of a set of masking characters used to restrict the values you can enter as parameter values (the Edit Mask also limits the values you can enter as default prompting values).

You can enter any of the following masking characters, or any combination of them:

- “A” (allows an alphanumeric character and requires the entry of a character in the parameter value).
- “a” (allows an alphanumeric character and does not require the entry of a character in the parameter value).
- “0” (allows a digit [0 to 9] and requires the entry of a character in the parameter value).
- “9” (allows a digit or a space, and does not require the entry of a character in the parameter value).
- “#” (allows a digit, space, or plus/minus sign, and does not require the entry of a character in the parameter value).
- “L” (allows a letter [A to Z], and requires the entry of a character in the parameter value).
- “?” (allows a letter, and does not require the entry of a character in the parameter value).
- “&” (allows any character or space, and requires the entry of a character in the parameter value).
- “C” (allows any character or space, and does not require the entry of a character in the parameter value).
- “. , : ; - /” (separator characters). Inserting separator characters into an Edit Mask is something like hard coding the formatting for the parameter field. When the field is placed on the report, the separator character will appear in the field object frame, like this: LLLL/0000. This example depicts an edit mask that requires four letters followed by four numbers.
- “<“ (causes subsequent characters to be converted to lowercase).
- “>” (causes subsequent characters to be converted to uppercase).
- “\” (causes the subsequent character to be displayed as a literal). For example, the Edit Mask “\A” would display a parameter value of “A.” If the Edit Mask is “00\A00,” then a valid parameter value would consist of two digits, the letter “A,” and then two additional digits.
- “Password”. Allows you to set the Edit Mask to “Password,” you can create conditional formulas specifying that certain sections of the report become visible only when certain user passwords are entered.

Note: Some of the Edit Mask characters require that you enter a character in their place (when entering a parameter value), while others allow you to leave a space, if needed. For example, if the Edit Mask is 000099, you can enter a parameter value with four digits, five digits, or six digits, since the ‘9’ Edit Mask character does not

require the entry of a character. However, since ‘0’ does require such an entry, you could not enter a parameter value with less than four digits.

5 Enter the default prompting values by adding to or highlighting values on the “Select or enter value to add” list and using the > and >> buttons to add these values to the **Default Values** list.

You can use the < and << buttons to remove prompting values from the list. The items you add appear in a drop-down list on the Enter Parameter Values dialog box with the default prompting values you specify.

6 Click **OK**.

The Create Parameter Field dialog box appears.

Note: When you have more than one item in the “Default Values” list box, the “Allow editing of default values” check box appears. This check box is selected by default to specify that you can edit or enter new values when prompted for parameter values. Select or clear this check box as needed.

7 Click **OK**.

The Field Explorer appears with the parameter selected.

8 Drag and drop the parameter into the report.

Subreports

23

A subreport is a report within a report. With subreports, unrelated reports can be combined into a single report. You can coordinate data that otherwise cannot be linked and present different views of the same data in a single report. This chapter shows you how to create and use subreports.

What are subreports?

A subreport is a report within a report. The process for creating a subreport is similar to the process of creating a regular report. A subreport can have most of the characteristics of a report, including its own record selection criteria. The only differences between a subreport and a primary report are that a subreport:

- Is inserted as an object into a primary report; it cannot stand on its own (although a subreport can be saved as a primary report).
- Can be placed in any report section and the entire subreport will print in that section.
- Cannot contain another subreport.
- Does not have Page Header or Page Footer sections.

There are four instances in which a subreport would typically be used:

- To combine unrelated reports into a single report. See "Combining unrelated reports by using subreports" on page 373.
- To coordinate data that cannot otherwise be linked. See "Using subreports with unlinkable data" on page 374.
- To present different views of the same data within a single report. See "Showing different views of the same data in a report" on page 378.
- To perform one-to-many lookups from a field that is not indexed on the lookup field. For more information, see "One-to-many links" in the online help.

Note: You can increase the performance of reports containing subreports by using on-demand subreports instead of regular, in-place subreports.

Unlinked vs. linked subreports

Unlinked

Unlinked subreports are free-standing; their data is not in any way coordinated with the data of the primary report.

In unlinked subreports, there is no attempt to match up the records in one report with records in the other. An unlinked subreport does not have to use the same data as the primary report; it can use the same data source or a different data source entirely. In addition, the subreport is not limited to reporting on a single table. An unlinked subreport can be based on a single table or on multiple tables. Regardless of the underlying data sources, the reports are treated as unrelated.

Linked

Linked subreports are just the opposite; their data is coordinated. The program matches up the records in the subreport with records in the primary report. If you create a primary report with customer information and a subreport with order information and link them, the program creates a subreport for each customer and includes in that subreport all the orders for that customer.

How subreport linking works

When you link a subreport to a primary report, the program creates the link by using a parameter field.

When a subreport link field is selected, the program creates:

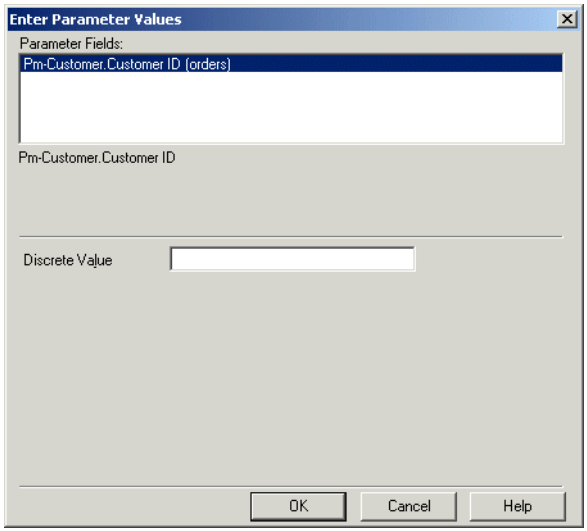
- A parameter field in the subreport which is then used to retrieve values passed to it by the primary report.
- A record selection formula for the subreport using the parameter field.
- The selection formula limits the subreport to those records in which the link field is equal to the parameter field value.

When the report is run, the program finds the first primary field record it needs and passes the value in the link field to the parameter field in the subreport. The program then creates the subreport with record selection based on the parameter field value. Here is an example:

- You create a report that shows customer data and a subreport that shows order data and then you link the two reports using the Customer ID field.
- When you run the report, the program finds the first customer record it needs and passes the Customer ID value from that record to the subreport parameter field.
- The program runs the Orders subreport. Since the subreport selection formula selects only those records in which the Customer ID value is equal to the parameter field value, and since that parameter field value is equal to the Customer ID in the first record in the primary report, the subreport contains only those records that have the same customer ID. Namely, those records that are orders for the first customer.
- When the subreport is finished, the program locates the second record it needs in the primary report, prints the customer data, and then passes this customer's ID number to the parameter field.
- The program then runs a subreport including only those order records for the second customer.
- The process continues until the report is finished.
- All of this parameter field manipulation takes place behind the scenes. You simply pick the fields that will link the primary report with the subreport and the program does the rest. The values are passed without the parameter field prompting you for a value.

Note: If you have a linked subreport and you click the Print Preview button on the Standard toolbar (from the Subreport Design tab), the program runs the subreport on its own, without waiting to receive a parameter field value from the primary report and without evaluating the tab text formula. In this case, the program displays the Enter Parameter Values dialog box and prompts you for a value.

The value entered in the box is the value the program uses to run the subreport.



Database links vs. subreports in one-to-many situations

When two tables in a report have a one-to-many relationship, the program retrieves the data in different ways depending on:

- Data source.
- Index situation.
- Record selection criteria.
- Whether you are creating a single report based on linked tables or a primary report that contains a subreport.

When you are considering whether to use a subreport or linked tables, you need to understand the ramifications of each. These issues are discussed fully in "Performance considerations in one-to-many links" on page 386.

As a general rule, if you have indexed tables, linked indexed fields, or range limiting record selection criteria based on the indexed fields, the program needs to read the same number of records whether you are linking tables in a single report or using subreports. Since each subreport is run as a separate report, linked tables may have a performance advantage. See "Indexed tables" on page 383.

Inserting subreports

1 On the **Insert** menu, click **Subreport**.

Tip: Another way to do this is to click the Insert Subreport button on the Insert Tools Toolbar.

The Insert Subreport dialog box appears.

2 To choose an existing subreport, click **Choose a report** and type the name. If you do not know the name, click the **Browse** button and locate it in the dialog box that appears.

To create a new subreport, click "Create a subreport" and type a name; then click the Report Wizard button. See "Creating a new report" on page 65 for information about using a wizard to create a report.

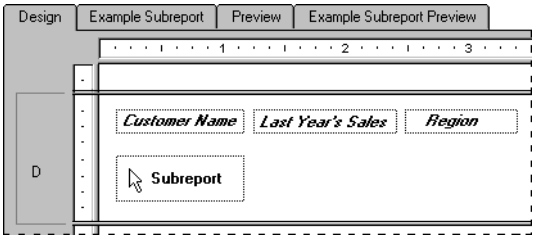
3 Choose **On-demand subreport** to have the ability to retrieve the data on the subreport when needed. Otherwise, all the subreport data will appear with the report.

4 Click **OK**.

The program displays an object frame.

5 Move the frame where you want it to appear in the report and click once to place it.

The program creates a Subreport Design tab, which is labeled with the subreport name. To edit the report, click the design tab and make your modifications.



6 Click the **Preview** tab to see your report.

7 If you chose the "On-demand subreport" option, click the subreport preview tab to see your subreport.

This tab is labeled with the name of your subreport.

Note: Using on-demand subreports will increase the performance of reports that contain subreports.

For information about creating a custom caption for the Subreport Preview Tab, see "Adding captions to on-demand subreports" on page 377.

Previewing subreports

There may be times when you want to preview a subreport on its own instead of previewing it as a part of the main report. For example, you may want to preview the subreport in order to view and analyze the data for a particular set of parameter values.

To preview a subreport

1. Click the subreport's Design tab.

   This tab is labeled with the name of your subreport.

2. Click **Print Preview** on the Standard toolbar.

   The program displays a preview of the selected subreport.

Saving a subreport as a primary report

You may find it advantageous to save a subreport as a primary report for the sake of distributing the information to a variety of audiences. For example, the primary report containing the subreport may be relevant for a stockholders meeting at the end of the fiscal year; however, the data contained in the subreport may be relevant for everyday use by your managers. In such cases, it is easy to save a subreport as a primary report.

To save a subreport as a primary report

1. In the Design tab, right-click the subreport and click **Save Subreport As** from the shortcut menu.

   The Save As dialog box appears.

2. Search for the appropriate directory in the Save As dialog box.
3. Type a new name for the subreport.
4. Click **Save**.

   The program saves the subreport as a primary report so that you can open it separately when necessary.

Updating subreports

In order to maintain the most up-to-date subreports, you may want to re-import a subreport automatically when opening the main report.

Note: Re-importing is available only for subreports that were created from a report file.

Re-importing not only updates the data, but updates the formatting, grouping, and structure of the subreport if any changes have been made. If you change the report the subreport was originally based on, you can have these changes reflected in the report containing the subreport.

You can either globally specify that all subreports should be re-imported, or set the automatic re-import for an individual subreport.

To globally update subreports when opening a main report

1. On the **File** menu, click **Options**.
2. In the Options dialog box, click the **Reporting** tab.
3. Click **Re-import Subreport When Opening Reports**.
4. Click **OK**.

   Any subreport will be updated when its main report is opened and refreshed.

To update a specific subreport when opening a main report

1. From the **Format** menu, click **Format Subreport**.
2. In the Format Editor dialog box, click the **Subreport** tab.
3. Click **Re-import When Opening**.

   The current subreport will be updated when the main report is opened and refreshed.

Manually updating subreport data

You can update your subreport data at any time.

1. On the **Design** tab, right-click the subreport.
2. Click **Re-import subreport** from the shortcut menu.
3. Click **Yes** to update the subreport data.

Linking a subreport to the data in the primary report

Frequently, the data in a subreport supplements the data in the primary report. You might, for example, have customer data in a primary report and then use subreports to show the orders for each customer.

In such cases, you will need to coordinate the data in the primary report with the data in the subreport so that the orders in each subreport match up with the correct customer.

To do this, you need to specify a field that is common to both the subreport and the primary report. With the Subreport Links dialog box, you create a link between the two common fields. Crystal Reports uses the link to match up records from the primary report to those in the subreport. The link makes certain that the “orders” data in the subreport sits on the same row as the corresponding “customer” data in the primary report.

To link a subreport to the data in the primary report

1. If you are creating a new subreport or importing an existing report as a subreport, from the **Insert** menu, click **Subreport**. Choose or create a report and click the **Link** tab.

   - or -

   If you have already placed a subreport in the primary report, but did not create a link at setup, navigate to the Subreport Links dialog box by choosing **Subreport Links** from the **Edit** menu.

   The Subreport Links dialog box appears.

2. Choose the subreport you want to link from the **For subreport** list (if it is not already selected).

3. Select the field you want used as a link field in the primary (containing) report from the **Available Fields** list.

4. Click the > arrow.

   The field is added to the "Field(s) to link to" list box, and is now selected as a link field.

5. Repeat steps 3 and 4 for each additional link, as desired.

6. Use the **Field link** section (which will only appear if you have selected a link field) to set up the link for each link field:
   - Select the field you want linked to the primary report from the "Subreport parameter field to use."
   - Select the "Select data in subreport based on field" check box on and select a field from the adjacent drop-down list to organize the subreport data based on a specific field (this is the quick equivalent of using the Select Expert). If nothing is specified here, the subreport will adopt the organization of the primary report.

7. Click **OK**.

When you run the report, the program will coordinate the data in the primary report with the data in the subreport.

Note: The field type of the Containing Report field determines which subreport fields are visible. Because the Report Designer reads dates as either strings, dates, or date/time fields, you must make sure your subreport parameter field type matches the field type set up in Report Options in the main report for the field you want linked.

Linking a subreport to the main report without modifying the selection formula

Crystal Reports uses a parameter field mechanism for linking subreports to main reports.

When linking a main report field that is not a parameter field to a subreport field, the program:

- Automatically creates a parameter field to complete the link.
- Modifies the subreport record selection formula to select those records in which the subreport field is equal to the parameter field value.

The need for a parameter field is implied; it is called an “Implicit Link” situation.

At times, you may wish to use a linked parameter field in a subreport without using it as part of the selection formula for the subreport. For instance, you may want the main report to pass in a summary value that can be used in calculations by the subreport, or you may want the main report to pass in the title of the subreport.

When you link a field in the main report to a parameter field that you have created in the subreport, the program:

- Checks the link you have specified.
- Does not create any additional parameter fields.
- Does not modify the subreport record selection formula.

Specifying a link is called an “Explicit Link” situation.

To link a subreport to a main report without modifying the selection formula

1. Create a parameter field in the subreport.
2. Link a field in the main report to that parameter field.

Combining unrelated reports by using subreports

At times, you may wish to combine unrelated reports into a single report. For example, you may want to create a single report that presents:

- Sales grouped by sales representative.
- Sales grouped by item.

While both reports deal with sales data, there is no real linear relationship between the reports.

Subreports can be used to combine unrelated reports into a single report like this. While the reports could be based on the same data set, they do not have to be. They could each be based on entirely different data sets.

Each of these reports is free-standing; the data in any of the reports is not linked in any way to data in another report. This is the easiest of the subreport options to work with.

Combining two or more unrelated reports

To combine two unrelated reports

1. Create the report you want printed first as the primary report.
2. Import an existing report for use as a subreport or create a new subreport.
3. Place the subreport into the Report Footer and it will print immediately after the primary report.

To combine three or more unrelated reports

1. Create the report you want printed first as the primary report.
2. Import or create each of the other reports you want to use as subreports.
3. Use the Section Expert to insert enough Report Footer sections to match the number of subreports that you are using.

   For example, if you want to use three subreports, insert two new Report Footer sections so that you have a total of three Report Footer sections.
4. In Report Footer A, place the subreport you want printed immediately after the primary report. In Report Footer B, place the subreport you want printed next, and so forth.

   The primary report will print first and then the subreports in the order that you placed them in the report.

   Note: Subreports can be placed side-by-side in the same Report Footer section. They will print next to each other at the end of the report.
5. Place the subreports into the Report Footer sections and they will print sequentially after the primary report.

Related topics

"Working with sections" on page 158

Using subreports with unlinkable data

Tables can be linked in a report as long as the following criteria are met:

- The link fields are both database fields.
- The link fields contain similar data.
- The link fields are the same length.
- The link field in the link to (lookup) table is indexed (PC databases only).

Linking tables is rarely a problem. However, there are some circumstances in which you cannot coordinate data from different tables because the data does not meet the linking criteria.

For example, linking to or from a formula field, or linking two unindexed tables cannot be done in a single report. Subreports must be used.

Linking to/from a formula field

There are situations in which you may need to link to or from a formula (calculated) field. For example, an employee ID could be an 11 character value that consists of a two-character department code followed by the employee's nine-character Social Security Number (for example, HR555347487).

The formula language makes it easy to extract the Social Security Number from this field:

```
{employee.EMPLOYEE ID} [-9 to -1]
```

- or -

```
{employee.EMPLOYEE ID} [3 to 12]
```

For the value HR555347487, either formula would return the value 555347487.

While the return value is a valid Social Security Number, the fact that it comes from a formula prevents you from using the field to link to a Social Security Number field in another table. You can report on and coordinate the values in the two tables, however, by using a subreport.

To link to/from a formula field

1. Create the primary report using a table that includes the Social Security Number field.
2. Create (or import) a subreport using the formula that extracts the Social Security Number from the Employee ID field (for this example, {@EXTRACT}). See "Inserting subreports" on page 369.
3. Place the subreport where you want it to appear in the primary report.
4. Link the subreport to the primary report by linking the Social Security Number field in the primary report ({file.SSN}) to the formula that extracts the number in the subreport ({@EXTRACT}). See "Linking a subreport to the main report without modifying the selection formula" on page 373.

Linking unindexed tables

When using PC (not SQL or ODBC) databases, the link field in the lookup database needs to be indexed in order to create a valid link. You can not link the tables in a single report when two tables contain related data yet neither is indexed on the field which you want to use as a link field, or when the primary table is indexed but the lookup table is not. You must use subreports if you want to coordinate the data in both tables.

Note: It is important to note that linking unindexed tables or linking from an indexed primary table to an unindexed lookup table may cause inefficient reporting. If your data set is large, this kind of report will take considerable time to run. Use this technique only if you do not have other options.

To link unindexed tables

1. Create the primary report.
2. Create (or import) the subreport and insert it into the primary report.
   See "Inserting subreports" on page 369.
3. Use the unindexed fields (or the indexed field in the primary table and the unindexed field in the lookup table) to link the subreport to the primary report.
   See "Linking a subreport to the data in the primary report" on page 371.

Creating an on-demand subreport

On-demand subreports can be especially useful when you want to create a report that contains multiple subreports. In this case, you can choose to have these subreports appear only as hyperlinks.

The actual data is not read from the database until the user drills down on the hyperlink. This way only data for on-demand subreports that are actually viewed will be retrieved from the database. This makes the subreports much more manageable.

Note: Data for an on-demand subreport is not saved unless the subreport is actually open in a preview window.

To create an on-demand subreport

1. Place an ordinary subreport in your primary report.
2. 
   Click the **Format** button on the Expert Tools toolbar.
   The Format Editor dialog box appears.
3. Click the **Subreport** tab and select the **On-demand Subreport** check box on.
4. Click **OK**.

Adding captions to on-demand subreports

To further organize a report, captions can be created for the Subreport Preview tab and for the placeholder frame of an on-demand subreport.

Captions are written by using formulas. Both placeholder frame captions and Subreport Preview tab captions can include field names from the main report.

A tab text caption replaces the subreport file name caption on the Subreport Preview tab.

Note: Placeholder frame captions only apply to on-demand subreports, while tab text captions apply to both on-demand subreports and regular subreports. Since the data from a regular subreport is visible on the Preview tab, there is no need for a frame caption when you format a regular subreport.

To add a caption

1. Select the subreport and click the **Format** button on the Expert Tools toolbar.

   The Format Editor dialog box appears.

2. Click the **Subreport** tab.

3. You can enter either an on-demand subreport caption or a preview tab caption by clicking the appropriate **Formula** button and opening the Formula Workshop.

4. Enter your formula in the **Formula text** box.

   Crystal syntax formula example:

   ```
   "More Information About " + {Customer.Customer Name}
   ```

   Basic syntax formula example:

   ```
   formula = "More Information About" + {Customer.Customer Name}
   ```

   Using the Xtreme.mdb sample database, these formulas would give you a caption like “More Information About Pathfinders” or “More Information About Rockshocks for Jocks.”

5. Click **Check** to check the formula for errors. If the program finds an error, it will prompt you with a message box detailing the nature of the error.

6. After fixing any errors, click **Save and close**.

7. Click **OK** to return to the report.

Note: A tab text formula is only evaluated when you drill-down on a subreport. If you preview a subreport separately from the primary report, the formula will not be evaluated.

Showing different views of the same data in a report

Subreports can be used to provide a different view of the data in the primary report. For example, you might want to show summary values at the top of a report and details at the bottom.

This can be done in a variety of ways. The two easiest methods are:

- By creating the summary report as the primary report and the details report as the subreport. In this method, the details subreport would be placed in the Report Footer section.
- By creating the details report as the primary report and the summary report as the subreport. In this method, the summary report would be placed in the Report Header section.

Use the appropriate link fields to link the report and coordinate the data.

Understanding Databases 24

This chapter describes the essentials you'll need to know to understand database design and use. After describing relational databases (including those from ODBC data sources), indexing, and table linking, the chapter introduces the concepts of server-side processing (pushing report processing to the server level) and field mapping (re-establishing report and database field mappings after a database changes).

Databases overview

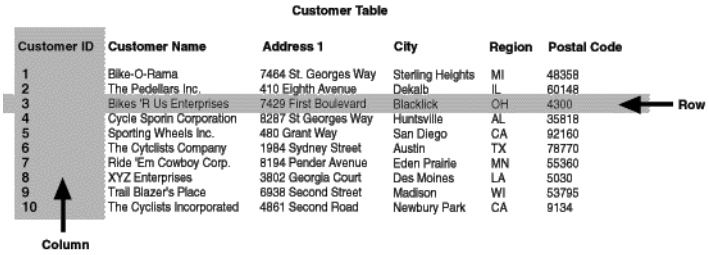
Though there are hundreds of Database Management Systems (DBMS) available, Crystal Reports eliminates many of the differences once it connects to the actual database files. The process of working with database files, tables, fields, and records is much the same, regardless of the actual type of data being accessed.

This chapter discusses several concepts and tasks common to working with database files. Using database aliases, locating moved or renamed database files, working with indexed tables, and linking tables are subjects common to anyone who designs reports in Crystal Reports. "Using SQL and SQL databases" on page 403, is especially important for anyone who accesses data in SQL databases and other database formats that are accessed through ODBC.

Relational database basics

The most popular architecture for database files used in the corporate world is based on the relational model. Applications that allow you to create databases with the relational model are, therefore, often referred to as Relational Database Management Systems (RDBMS).

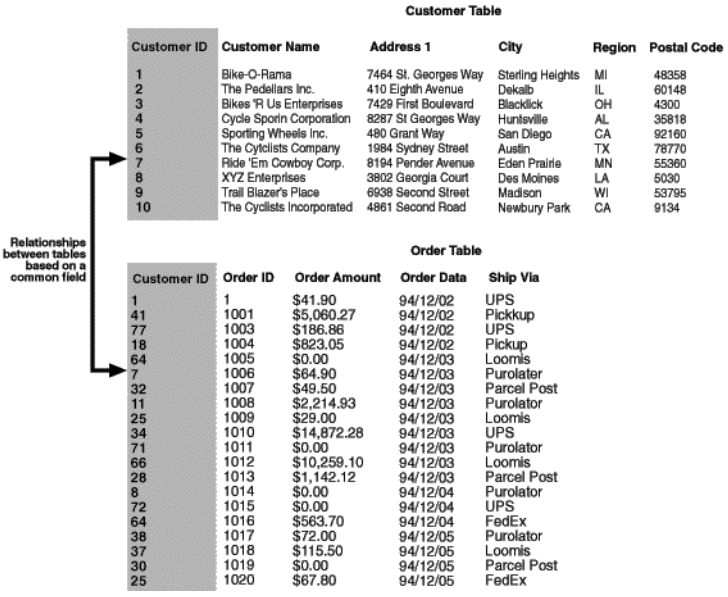
In a relational database, data is organized in a system of rows and columns. The rows are called records, and the columns are called fields. Each record contains a collection of related data, all information relating to a specific customer, for example. Each field refers to a common type of data that exists in all records, the names of the customers, for example. Records and fields are stored in a database table. The following diagram illustrates the basic relational database model:



Customer ID	Customer Name	Address 1	City	Region	Postal Code
1	Bike-O-Rama	7464 St. Georges Way	Sterling Heights	MI	48358
2	The Pedellars Inc.	410 Eighth Avenue	Dekalb	IL	60148
3	Bikes 'R Us Enterprises	7429 First Boulevard	Blacklick	OH	4300
4	Cycle Sporin Corporation	8287 St Georges Way	Huntsville	AL	35818
5	Sporting Wheels Inc.	480 Grant Way	San Diego	CA	92160
6	The Cyticlists Company	1984 Sydney Street	Austin	TX	78770
7	Ride 'Em Cowboy Corp.	8194 Pender Avenue	Eden Prairie	MN	55360
8	XYZ Enterprises	3802 Georgia Court	Des Moines	LA	5030
9	Trail Blazer's Place	6938 Second Street	Madison	WI	53795
10	The Cyclists Incorporated	4861 Second Road	Newbury Park	CA	9134

Often, data in two different tables can be related by a common field. For example, a Customers table will have a Customer ID for each customer, and an Orders table will have the Customer ID of each customer who placed an order, demonstrating a relationship between tables. The two tables can be linked by a common field see "Linking tables" on page 385.

The following diagram displays how two tables can have a relationship:



Customer ID	Customer Name	Address 1	City	Region	Postal Code
1	Bike-O-Rama	7464 St. Georges Way	Sterling Heights	MI	48358
2	The Pedellars Inc.	410 Eighth Avenue	Dekalb	IL	60148
3	Bikes 'R Us Enterprises	7429 First Boulevard	Blacklick	OH	4300
4	Cycle Sporin Corporation	8287 St Georges Way	Huntsville	AL	35818
5	Sporting Wheels Inc.	480 Grant Way	San Diego	CA	92160
6	The Cytclists Company	1984 Sydney Street	Austin	TX	78770
7	Ride 'Em Cowboy Corp.	8194 Pender Avenue	Eden Prairie	MN	55360
8	XYZ Enterprises	3802 Georgia Court	Des Moines	LA	5030
9	Trail Blazer's Place	6938 Second Street	Madison	WI	53795
10	The Cyclists Incorporated	4861 Second Road	Newbury Park	CA	9134

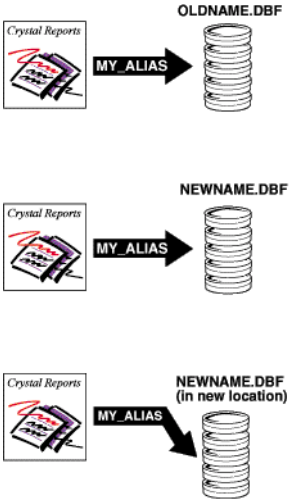
Customer ID	Order ID	Order Amount	Order Data	Ship Via
1	1	$41.90	94/12/02	UPS
41	1001	$5,060.27	94/12/02	Pickkup
77	1003	$186.86	94/12/02	UPS
18	1004	$823.05	94/12/02	Pickup
64	1005	$0.00	94/12/03	Loomis
7	1006	$64.90	94/12/03	Purolater
32	1007	$49.50	94/12/03	Parcel Post
11	1008	$2,214.93	94/12/03	Purolator
25	1009	$29.00	94/12/03	Loomis
34	1010	$14,872.28	94/12/03	UPS
71	1011	$0.00	94/12/03	Purolator
66	1012	$10,259.10	94/12/03	Loomis
28	1013	$1,142.12	94/12/03	Parcel Post
8	1014	$0.00	94/12/04	Purolator
72	1015	$0.00	94/12/04	UPS
64	1016	$563.70	94/12/04	FedEx
38	1017	$72.00	94/12/05	Purolator
37	1018	$115.50	94/12/05	Loomis
30	1019	$0.00	94/12/05	Parcel Post
25	1020	$67.80	94/12/05	FedEx

Aliases

For a variety of reasons, database names and locations get changed. If you create a report, then change the name or location of a table or file, the Report Designer must be able to find the new name or location. This is especially important when you create formulas in your report that access a table that has been renamed or moved. To fix the reference for a single field would not be difficult, but to find every formula that uses that field could be a difficult and time consuming task.

To solve this problem, the Report Designer uses aliases to refer to database tables and files. Aliases are pointers, internal devices that tell the program where it should look for a database field. Now, if you change the name or location of the database, you simply reset the pointer. See “Locating files” on page 382. The name of the alias does not change, so your formulas are not affected. The Report

Designer looks to the alias for the location and name, goes to the new location for the database field, and executes the formula without a problem.

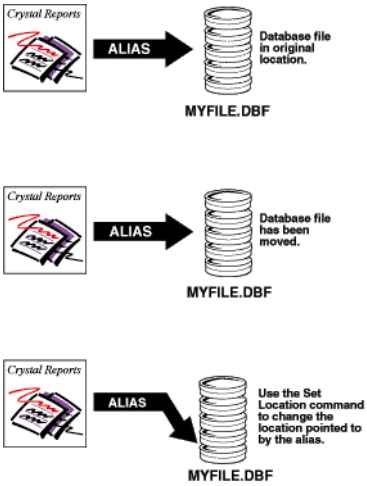


The Report Designer automatically assigns default alias names to database tables when you first select the table or file. By default, an alias matches the original name of the table. In databases where the database table is a separate file (for instance, dBASE), the name of the database file is used without the file name extension. For example, if you are using the dBASE database file Company.dbf, the program will assign a default alias name of Company to the file. You can accept the default alias or assign a new one to the database table.

You can change an alias at any time using the Database Expert. However, if you have already created formulas in your report using the original alias name, you will need to edit the formulas to use the new alias.

Locating files

When a database file is moved or renamed, Crystal Reports will not be able to find the data the next time the report is printed. On other occasions, a report may be created on one machine where all of the database data is stored in a certain directory, then the report is copied or moved to another machine that stores the same data in a different directory. In any of these events, you need to verify the location of the database files accessed by the report and reset the alias pointers to the new database location or name.



The Verify Database command on the Database menu checks the alias pointers stored in a report file to verify that the database files expected are located in the indicated directories. If the databases are not found in the specified location, the program notifies you of the discrepancies.

Use the Set Datasource Location command on the Database menu to change the alias pointers stored by Crystal Reports. The Set Datasource Location command provides a simple way to indicate the new name or location of database files. In addition, the Set Datasource Location command automatically converts your database driver to the data source you have chosen. For example, you can automatically convert a direct access data source to an ODBC data source using the Set Datasource Location command.

Related topics

See “Changing the data source accessed by a report” in the online help.

Indexed tables

Creating indexes for database tables can increase the speed of data access and reduce the time it takes for the program to evaluate data. Some DBMS applications automatically index your database tables, while others require that you create an index yourself. For the best report generation performance, make sure each of your database tables has a corresponding index.

Note: Some DBMS applications do not support indexed tables. Refer to the documentation for your DBMS to find out if it supports indexes and how to create them. If your DBMS documentation does not mention indexed tables, it may not support them, and you should link tables based on common fields. The Links tab of the Database Expert can also help you determine if your tables include indexes.

Indexes organize the records in a relational database table so that data can be located easier. For example, assume you have a table with the following data:

Order#	Customer	Amount
10444	Allez Distribution	25141.50
10470	BG Mountain Inc.	19164.30
10485	Sierra Mountain	8233.50
10488	Mountain Toad	24580.50
10495	SFB Inc.	7911.80
10501	La Bomba de Bicicleta	1956.20
10511	BG Mountain Inc.	1683.60
10544	Sierra Bicycle Group	19766.20
10568	Mountain Tops Inc.	29759.55
10579	Sierra Bicycle Group	12763.95

The information in this table is organized according to the Order# field. This is fine anytime you want to look up information in the table based on order numbers. However, what if you want to look up information specific to a certain customer?

Say you want to look up all orders made by Sierra Bicycle Group. The database engine must begin by looking at the first order number in the list and checking to see if the customer name matches the request. If not, it goes to the second order number, and checks that customer name. When an order number is reached that contains the correct customer name, the database engine retrieves the information, then continues to the next order number. Using this technique, both the Order# field and the Customer field must be read for every single record in the table. This takes a long time and a large amount of computer processing effort for examining extensive database tables with thousands, or even millions of records.

Instead, you can create an index for the table based on the Customer field. Such an index might look like this:

Customer	Pointer to Order#
Allez Distribution	10444
BG Mountain Inc.	10470
BG Mountain Inc.	10511
La Bomba de Bicicleta	10501
Mountain Toad	10488
Mountain Tops Inc.	10568
SFB Inc.	10495
Sierra Bicycle Group	10544

Customer	Pointer to Order#
Sierra Bicycle Group	10579
Sierra Mountain	10485

In this index, information is organized by customers, not order numbers. Also, notice that the second column actually contains pointers to specific order numbers in the original table. By using this index, the database engine can search just the information in the Customer column until it finds the customer you are interested in, Sierra Bicycle Group.

For each correct customer entry the database engine finds in the index, it looks up the matching order in the table according to the pointer in the second column of the index. Only the orders for the correct customer are read. Finally, since information in the index is organized according to the customer names, the database engine does not need to continue searching through the index or the table as soon as it finds an index entry that does not match the requested customer.

The advantage of this highly organized search through a database table according to an index is speed. Using indexes speeds up data retrieval and report generation, important factors when reporting on large database files.

Linking tables

You link tables so records from one table will match related records from another. For example, if you activate an Orders table and a Customers table, you link the tables so that each order (from the Orders table) can be matched up with the customer (from the Customer table) that made the order.

When you link, you are using a field that is common to both tables. Crystal Reports uses the link to match up records from one table with those from the other. In this example, the link assures that the data in each row of the report refers to the same order.

Link from and link to

When you link two tables, you link from one table to another table. The *from* table is used as a primary table, while the *to* table acts as a lookup table where records are looked up by the primary table. In a simple link, the Report Designer examines the first record in the primary table and finds all matching records in the lookup table. Once all matches have been found in the lookup table for the first record in the primary table, all matches in the lookup table for the next record in the primary table are found.

Link relationships

When you link records from one table to another table, the records will typically fall under one of two relationship types:

- One-to-one
- One-to-many

One-to-one relationships

In a one-to-one relationship between records in two linked tables, for every record in the primary table there is only one matching record in the lookup table (based on the linked fields). For example, in the Xtreme.mdb database, the Employee table can be linked to the Employee Addresses table based on the Employee ID field in each table. The Employee table contains information about employees at the company, the positions they hold, their salaries, hiring information, and so on. The Employee Addresses table contains each employee's home address. There is only one record for each employee in each of these tables. Therefore, if the Employee table is linked to the Employee Addresses table, only one record will be found in the Employee Addresses table for each record in the Employee table. This is a one-to-one relationship.

One-to-many relationships

In a one-to-many relationship between records in two linked tables, for every record in the primary table, there may be more than one matching record in the lookup table, based on the linked fields. In the Xtreme.mdb database, the Customer table can be linked to the Orders table based on the Customer ID field in each table. The Customer table contains information about each customer that has placed an order with the company. The Orders table contains information about orders that customers have placed. Since customers can place more than one order, there may be more than one record in the Orders table for each customer record in the Customers table. This is a one-to-many relationship.

Performance considerations in one-to-many links

The information provided in this section is intended to help you maximize processing speed and minimize network traffic when you are running your reports. You will learn about the best ways to use selection formulas and indexes in one-to-many situations to make your reporting more efficient. If you do not use the information in this section, your reports may end up processing dozens or even thousands more records than necessary.

When a one-to-many situation exists between two database tables and the program matches up records from the tables, there are a number of factors that determine how many records the program reads and evaluates.

The tables that follow show the effects of the different factors on the number of records the program ultimately has to read. The charts are based on these assumptions:

- Table A contains 26 records (one for each letter in the alphabet).
- Table B contains 2600 records (100 matching records for every record in Table A).
- The scenario is to produce a report that finds two specific records in Table A and the 200 records (100+100) in Table B that match those two records in Table A.

  In a best case scenario, the program would only have to read about 200 records to accomplish the task.

  In a worst case scenario the program would have to read about 67,600 records to accomplish the same task.

Note: The performance considerations for data files are different from the considerations for SQL databases. A data file is any non-SQL database that is accessed directly from Crystal Reports. For the purpose of this discussion, an SQL database is any database capable of accepting SQL commands accessed directly from Crystal Reports or through ODBC as well as any other database types that are accessed through ODBC. For a better understanding of the difference between direct access databases and ODBC data sources, see Accessing Data Sources in the online help.

Extended descriptions of chart columns

The performance charts use the following columns:

- **Linking or Subreport**
  Are you creating a report from linked databases or are you inserting a subreport and binding it to the data in your primary report?
- **Selection Formula**
  Does your primary report include a record selection formula that sets range limits on the key (indexed) field in Table A?
- **Index A**
  Is Table A on the field you are going to use indexed to match up the records?
- **Index B**
  Is Table B on the field you are going to use indexed to match up the records?
- **Reads A**
  How many records does the program have to read out of Table A to find the two records it is looking for?
- **For each A reads in B**
  How many records does the program have to read in Table B to find the 200 records it is looking for?
- **Total Records Read**
  What is the total number of records the program has to process to complete the task?

PC Data						
Linking/ Subreport	**Selection Formula**	**Index A**	**Index B**	**Reads A**	**For each A reads in B**	**Total Records Read**
Linking	No	Yes or No	Yes	26	100 (26*100)	2600
Linking	Yes	No	Yes	26	100 (26*100)	2600
Linking	Yes	Yes	Yes	2	100 (2*100)	200

PC Data						
Subreport	No	No	No	26	2600 (26*2600)	67,600
Subreport	No	Yes	No	2	2600 (26*2600)	67,600
Subreport	No	Yes	Yes	26	100 (26*100)	2600
Subreport	Yes	No	No	2	2600 (2*2600)	5200
Subreport	Yes	No	Yes	26	100 (26*100)	2600
Subreport	Yes	Yes	Yes	2	100 (2*100)	200

SQL Data				
Linking/ Subreport	**Selection Formula**	**Reads A**	**For each A reads in B**	**Total Records Read**
Linking	No	26	100 (26*100)	2600
Linking	Yes	2	100 (2*100)	200
Subreport	No	26	100 (26*100)	2600
Subreport	Yes	2	100 (2*100)	200

Data file considerations

When working with data files, one-to-many links can occur when you link tables in a single report or when you add a subreport to your report.

Linking data files

When retrieving data from linked data files in one-to-many situations, the program uses the following process:

- If there is a selection formula, the program parses the selection formula and passes what it can down to the database DLL. This is generally range limit information. Consider the following record selection formula:

  ```
  {customer.REGION} in “CA” to “IL” AND
  Remainder ({customer.CUSTOMER ID},2)=0
  ```

  In this formula, the part before the “and” operator contains range selection criteria for the Region field. The region must fall alphabetically between “CA” and “IL.” The program passes this kind of condition down to either the database DLL (for PC data) or the server (for SQL data). See “Record Selection” on page 107.

The second half of the selection formula, however, requires processing that must be done in the Report Engine. It uses a built-in function to manipulate and evaluate a field value and it cannot be done in the database DLL or the server. The program does not pass this condition to the database DLL.

- If there is an index on Table A, and the range limit selection condition is based on the indexed field ({customer.REGION} in this example), the program goes directly to the record it is seeking in Table A (the first CA record) and reads it.
  - For that record, the program locates the first matching record in Table B, using the Table B index.
  - The program passes this merged record (A+B) back to the Report Designer, which tests the record against the entire selection formula.
  - The program then reads the second matching record and passes the merged record on, and then reads the third matching record, and so on, until it has read all of the matching records.
  - The program then returns to Table A and reads the next record. There is no need to test the record to see if it meets the CA condition; the field is indexed and the records are in alphabetic order. But the program tests the record to see if it goes beyond the “IL” condition (for example, could the next record be from Mississippi or Tennessee?). If the record is still within the specified range, the program begins the matching process again for that record.
  - The program continues the process until it has located all targeted Table A records and the matching Table B records.

    To find two records in Table A and the 100 records in Table B that match the Table A records, the program reads 200 records.
- If there is no index on Table A, or if there is an index but the range limit selection condition is not based on the indexed field, the program reads the first record it finds.
  - For that record, the program uses the Table B index to locate the first matching record in Table B.
  - The program passes this merged record (A+B) back to the Report Engine, which tests it against the entire selection formula.
  - The program then locates the second matching record in Table B and passes that merged record back, then the third record, and so on, until it has located, merged, and passed back all the records in Table B that match the first record in Table A.
  - The program then moves on to the next record in Table A and begins the matching and merging process all over again.

    To find two records in Table A and the 100 records in Table B that match the Table A records, the program reads 2600 records.

Subreports and data files

If your primary report is based on Table A, the subreport is based on Table B, and the records are linked, your primary considerations are as follows:

- The number of subreports that are run by the program is determined by the index and the selection formula situation in the primary report:
  - If Table A is indexed, and if the primary report has a selection formula that passes down range limit conditions for the indexed field, the program runs two subreports.
  - If Table A is not indexed, or if Table A is indexed but the selection formula does not pass down range limit conditions for the indexed field, the program runs 26 subreports.
- The number of records read for each subreport is determined by the index situation on Table B:
  - If you have an index on Table B, the program will read only the matching records (100) when it runs a subreport.
  - If you do not have an index on Table B, the program will always read every record in Table B (2600) when it runs a subreport.

SQL database considerations

Since indexes are not critical with SQL data, the primary concern with both linked tables and subreports is whether or not there is a selection formula in the primary report that puts range limits on Table A. See "Linking data files" on page 388.

Linked SQL tables

If there are range limit conditions in the selection formula, the program passes those conditions down to the server.

- If there is a selection formula that puts range limits on Table A, the server locates the records in Table A that satisfy the selection criteria (2), matches them up with the appropriate records in Table B (100), and returns 200 merged records to the Report Engine.
- If there is no selection formula, or if there is a selection formula that does not put range limits on Table A, the server matches up each record in Table A (26) with the appropriate records in Table B (100), and returns 2600 merged records to the Report Engine.

In either case, the Report Engine will then apply the entire selection formula to the merged records.

Subreports and SQL databases

If you are creating a primary report based on Table A and a subreport based on Table B:

- The number of subreports that are run is determined by the selection formula situation in the primary report:

  - If there is a selection formula and it passes down range limits on Table A, the program runs a subreport only for those records that satisfy range limit conditions (2).
  - If there is no selection formula, or if the selection formula does not pass down range limits on Table A, the program runs a subreport for every record in Table A (26).
- The number of records read by each subreport remains the same regardless of whether there was range limit selection on Table A. Each subreport will read only those records in Table B that match each record read in Table A (100).

Performance considerations for all reports

Consideration 1

With both data files and SQL databases, the program parses the entire selection formula and passes down whatever parts of the criteria it is able to translate (pass), wherever they may physically appear in the formula. Thus, if the formula finds criteria it can pass, then criteria that it cannot, then criteria that it can, it passes down the first part, skips the second, and then passes down the third.

- In the case of data files, the program passes down the criteria that it can to the database translation layer.
- In the case of SQL databases, the program passes down to the server the criteria that it can in the form of a WHERE clause.

While there are exceptions, as a general rule the program can pass down any part of the record selection formula that compares a field with a constant. Typically, this means that it can pass down any kind of record selection criteria that can be set up in the Select Expert (equal to, one of, less than, greater than, less than or equal, greater than or equal, between, starting with, or like constant).

There are two special selection formula situations that you need to consider. In these situations, the record selection formula includes multiple conditions, some of which can be passed down while others cannot.

- AND situations

  ```
  {customer.REGION} = "CA" and
  {customer.CUSTOMER ID}[3 to 5] = "777")
  ```

  In this situation, the program sees that it can pass down the condition before the And operator but not the condition after. Since the only records that will meet the second condition will have to meet the first as well, the program passes down the first condition, retrieves the data set that satisfies the condition, and then applies the second condition only to the retrieved data. The rule for AND situations is that the program passes down whatever conditions it can.

  Note: If all of the conditions in an AND situation can be satisfied on the server or in the database DLL, the program passes them all down.

- OR situations

  ```
  {customer.REGION} = "CA" or
  {customer.CUSTOMER ID}[3 to 5] = "777")
  ```

  In this situation, the program also sees that it can pass down the condition before the Or operator but not the condition after. Since there are records that can satisfy the second condition without satisfying the first, passing the first condition down does not make any sense because it will retrieve an incomplete data set. In other words, even if it retrieves all the data that satisfies the first condition, it will still have to retrieve all the data in the table(s) before it can apply the second condition in Report Designer. Thus, instead of duplicating parts of the data retrieval, the program passes nothing down. It retrieves all the data and then runs both tests in Report Designer. The rule for OR situations is that the program either passes down all the tests, or none of the tests.

  Note: If all the tests in an OR situation can be performed on the server or in the database DLL, the program passes them all down.

Consideration 2

To make certain the program can use the index on Table A to enhance performance, make certain:

- There is a selection formula.
- There are range limits in the selection formula on the key (indexed) field in Table A.
- Use Indexes is selected in the Options dialog box.

Consideration 3

If the fields you are using from Table A are not indexed, but there is an indexed field that you can use in your record selection request, use that field. For example, assume that you have three products (Product 1, Product 2, and Product 3) and you want to identify all sales of Product 2 in the U.S. There is no index on the Product field but there is an index on the Order Date field. Since you know that Product 2 did not begin shipping until July of 1995, you can improve speed by limiting your report to orders placed in and after July 1995 using the selection formula. In such a case, the program uses the Order Date index to retrieve only those orders from July 1995 and afterward (a small subset of the entire database) and then searches for the occurrences of Product 2 in that subset, not in the entire database.

The Database Expert Links tab

The Database Expert Links tab lets you easily link two or more tables. Choose the Database Expert from the Database menu to display all current tables; then choose the Links tab to display all current links.

The easiest way to link database tables is to select Auto Link in the Database Expert Links tab. Auto Link automatically chooses links for your tables based on common fields in tables or indexed fields (if your database supports indexed fields).

Linking indexed tables

When you are linking direct-access database tables, you link from a primary table to a field in the lookup table. The link field in the primary table can be indexed, but does not have to be. The link field in the lookup table does not have to be indexed, either, unless you *aren't* using a native driver for your connection.

In addition, the fields used to link two tables must have the same data type. For example, you can link a string field in one table to a string field in another table, or a numeric field in one table to a numeric field in another table, but you cannot link a numeric field in one table to a string field in another table.

Note:

- Some DBMS applications allow you to convert the field value to another data type in the index. For instance, the field in the table can be numeric, while the index converts the field value to a string. However, if you choose to use that field to link to another table, you must link to a field of the original data type. You cannot link a string value to a numeric field that has been converted to a string in the index.
- If you are linking tables from two different ODBC data sources, MS SQL Server and Oracle, for example, you can only use string fields to link the tables. String fields are stored in databases in the same manner, regardless of the data source. Other types of values, however, may not be stored the same way in different data sources, so you cannot link different data sources in Crystal Reports using anything other than string values.
- When linking direct-access database tables using native drivers (non-SQL), the only join type available is Left Outer join.

Changing the index used in linking

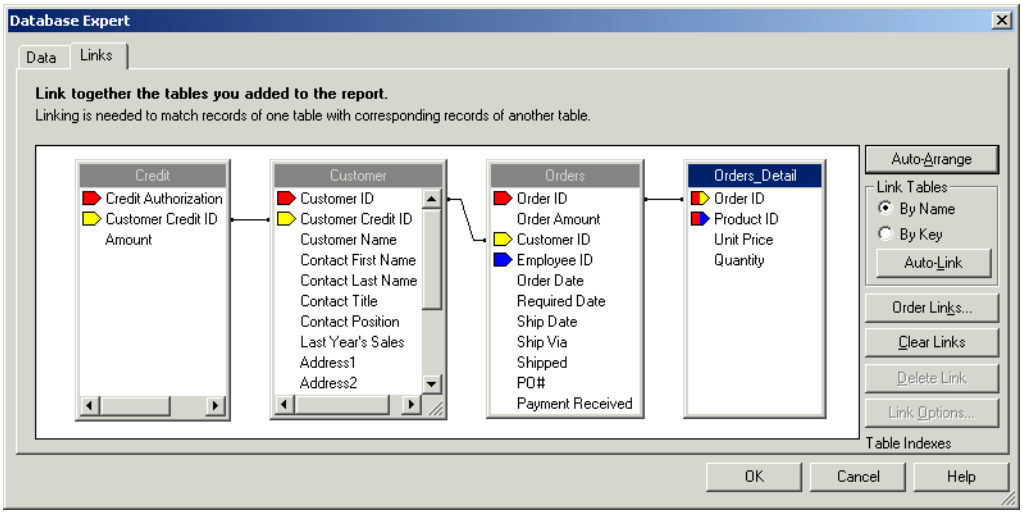
When using the Smart Linking feature to link tables using a field that is a component of multiple indexes (two or more), Crystal Reports selects one of the indexes for the link. That index may or may not be the one you want to use. Delete the link made by the Smart Linking feature and manually link the tables. See for Linking multiple tables more information.

Note: Not all DBMS applications support indexed tables. Verify that your database uses indexes before trying to select an index for linking. Refer to your DBMS documentation to find out if your DBMS can use indexes, and to learn how to create them.

Link processing order

When there is more than one link, Crystal Reports needs to know in what order it should process the links. By default, the processing order matches the order in which the links appear on the Links tab. Use the arrow buttons in the Links Order dialog box to change the default order.

For example, if you have chosen the Credit, Customer, Orders, and Orders Details tables from the Xtreme sample database, the links tab shows the tables linked as follows.



In this case, the links will be processed first between the Credit/Customer tables, then between the Customer/Orders tables, and finally between the Orders/Orders Details tables. The Links Order dialog box shows you this default order and allows you to change the positions in the processing hierarchy as you require.

Note: Different link orders may result in different data sets returned for use in your report. As well, link order has a significant effect on performance.

Linking options

Crystal Reports enables you to specify the type of join and type of link you want to use when linking tables. Joins and links indicates how linked fields in two tables are compared when records are read. Join and link options can be specified in the Link Options dialog box.

Note: When you link fields using joins, no indexed fields are required.

The join types are:

- "Inner join" on page 395
- "Left Outer join" on page 395
- "Right Outer join" on page 396
- "Full Outer join" on page 397

The link types are:

- "Equal [=] link" on page 398
- "Greater Than [>] link" on page 398

- "Greater Than Or Equal [>=] link" on page 399
- "Less Than [<] link" on page 400
- "Less Than Or Equal [<=] link" on page 401
- "Not Equal [!=] link" on page 402

Inner join

An Inner join is the standard type of join. The result set from an Inner join includes all the records in which the linked field value in both tables is an exact match. For instance, you can use an Inner join to view all customers and the orders they have placed. You will not get a match for any customer who has not placed orders.

Customer Table Customer ID	Customer Table Customer Name	Orders Table Order Amount
52	Allez Distribution	25141.50
53	BG Mountain Inc.	19164.30
53	BG Mountain Inc.	1683.60
57	Hansen MTB Inc.	15716.40
58	La Bomba de Bicicleta	1956.20
60	Mountain Toad	24580.50
62	SFB Inc.	7911.80
63	Sierra Bicycle Group	19766.20
63	Sierra Bicycle Group	12763.95
64	Sierra Mountain	8233.50

Left Outer join

The result set from a Left Outer join includes all the records in which the linked field value in both tables is an exact match. It also includes a row for every record in the primary (left) table for which the linked field value has no match in the lookup table. For instance, you can use a Left Outer join to view all customers and the orders they have placed, but you also get a row for every customer who has not placed any orders. These customers appear at the end of the list with blanks in the fields that would otherwise hold order information:

Customer Table Customer ID	Customer Table Customer Name	Orders Table Order Amount
52	Allez Distribution	25141.50
53	BG Mountain Inc.	19164.30
53	BG Mountain Inc.	1683.60
57	Hansen MTB Inc.	15716.40

Customer Table Customer ID	Customer Table Customer Name	Orders Table Order Amount
58	La Bomba de Bicicleta	1956.20
60	Mountain Toad	24580.50
62	SFB Inc.	7911.80
63	Sierra Bicycle Group	19766.20
63	Sierra Bicycle Group	12763.95
64	Sierra Mountain	8233.50
54	Bicicletas Aztecas	
55	Deely MTB Inc.	

Note: Left Outer and Right Outer joins are handled differently in the SQL language from other join types. If the database is accessed through ODBC, Crystal Reports uses ODBC syntax in the SQL statement. If you are connecting to an SQL database directly (not through ODBC), Crystal Reports uses a syntax native to the database. For more information about what an Outer join looks like in an SQL statement, refer to Microsoft ODBC™ documentation or to the documentation for your SQL database.

Right Outer join

The result set from a Right Outer join includes all the records in which the linked field value in both tables is an exact match. It also includes a row for every record in the lookup (right) table for which the linked field value has no match in the primary table. If you link the Customer table to the Orders table, you get one row in the table for each order a customer has placed. You also get a row for every order found that cannot be linked to a customer. Theoretically, this should not happen, but if an inexperienced sales person forgot to assign a customer ID to an order, you can quickly locate that order with a Right Outer join. The resulting table leaves a blank in any of the Customer fields for the order without a customer.

Customer Table Customer ID	Orders Table Order ID	Orders Table Order Amount
52	6	25141.50
53	11	19164.30
53	21	1683.60
57	4	15716.40
58	20	1956.20
60	16	24580.50
62	19	7911.80
63	28	19766.20

Customer Table Customer ID	Orders Table Order ID	Orders Table Order Amount
63	32	12763.95
64	14	8233.50
	25	10320.87

Note: Left Outer and Right Outer joins are handled differently in the SQL language from other join types. If the database is accessed through ODBC, Crystal Reports uses ODBC syntax in the SQL statement. If you are connecting to a SQL database directly (not through ODBC), Crystal Reports uses a syntax native to the database. For more information about what an Outer join looks like in an SQL statement, refer to Microsoft ODBC documentation or to the documentation for your SQL database.

Full Outer join

A Full Outer join is a bidirectional outer join where you can see all records in your linked tables. The result set from a Full Outer join includes all the records in which the linked field value in both tables is an exact match. It also includes a row for every record in the primary (left) table for which the linked field value has no match in the lookup table, and a row for every record in the lookup (right) table for which the linked field value has no match in the primary table. If you link the Customer table to the Orders table, you get one row in the table for each order a customer has placed. You also get a row for every order found that cannot be linked to a customer, and a row for every customer for whom an order cannot be found.

Customer Table Customer ID	Orders Table Order ID	Orders Table Order Amount
52	6	25141.50
53	11	19164.30
53	21	1683.60
57	4	15716.40
58	20	1956.20
60	16	24580.50
62	19	7911.80
63	28	19766.20
63	32	12763.95
64	14	8233.50
65		
66		
	25	10320.87

Equal [=] link

The result set from an Equal link includes all the records where the linked field value in both tables is an exact match. In the following example, the Customer table is linked to the Orders table by the Customer ID field. When the program finds a Customer ID in the Orders table that matches a Customer ID in the Customer table, it displays information for the corresponding records in both tables.

SQL uses the following syntax to describe an Equal link:

```
SELECT Customer.‘Customer ID‘,
     Customer.‘Customer Name‘,
     Orders.‘Order Amount‘
FROM ‘Customer‘ Customer,
     ‘Orders‘ Orders
WHERE Customer.Customer ID =
     Orders.Customer ID
```

This statement produces the following data:

Customer Table Customer ID	Customer Table Customer Name	Orders Table Order Amount
52	Allez Distribution	25141.50
53	BG Mountain Inc.	19164.30
53	BG Mountain Inc.	1683.60
57	Hansen MTB Inc.	15716.40
58	La Bomba de Bicicleta	1956.20
60	Mountain Toad	24580.50
62	SFB Inc.	7911.80
63	Sierra Bicycle Group	19766.20
63	Sierra Bicycle Group	12763.95
64	Sierra Mountain	8233.50

Greater Than [>] link

The result set from a Greater Than link includes all records in which the linked field value from the primary table is greater than the linked field value in the lookup table. As an example, a company may want to compare the salaries made by all their sales representatives to the salaries made by all their sales managers. The company executives want to make sure no sales representative is making more money than any manager.

With this in mind, you can link the SalesRep table to the Manager table by the Salary field in each table using a Greater Than link:

```
SELECT SalesRep.‘Last Name‘,
     SalesRep.‘Salary‘,
     Manager.‘Last Name‘,
     Manager.‘Salary‘
FROM ‘SalesRep‘ SalesRep,
     ‘Manager‘ Manager
WHERE SalesRep.‘Salary‘ >
     Manager.‘Salary‘
```

This SQL statement might produce data similar to this:

SalesRep Table Last Name	SalesRep Table Salary	Manager Table Last Name	Manager Table Salary
Davolio	$35,000.00	Fuller	$32,000.00
Davolio	$35,000.00	Brid	$30,000.00
Davolio	$35,000.00	Buchanan	$29,500.00
Dodsworth	$48,300.00	Hellstern	$45,000.00
Dodsworth	$48,300.00	Fuller	$32,000.00
Dodsworth	$48,300.00	Brid	$30,000.00
Dodsworth	$48,300.00	Buchanan	$29,500.00
Dodsworth	$48,300.00	Martin	$35,000.00
Patterson	$30,000.00	Buchanan	$29,500.00

In this table, there is no relationship established between sales representatives and sales managers. Since all managers have seniority over all sales representatives, a company might find it necessary to check if any representatives make more money than any managers, evidence of a salary problem that needs to be remedied.

Greater Than Or Equal [>=] link

The result set from a Greater Than Or Equal link includes all records in which the linked field value in the primary table is greater than or equal to the linked field value in the lookup table. The example here is identical to the example for the Greater Than join, except that it uses the Greater Than Or Equal link:

```
SELECT SalesRep.‘Last Name‘,
     SalesRep.‘Salary‘,
     Manager.‘Last Name‘,
     Manager.‘Salary‘
FROM ‘SalesRep‘ SalesRep,
     ‘Manager‘ Manager
WHERE SalesRep.‘Salary‘ >=
     Manager.‘Salary‘
```

This statement might produce data such as this:

SalesRep Table Last Name	SalesRep Table Salary	Manager Table Last Name	Manager Table Salary
Davolio	$35,000.00	Fuller	$32,000.00
Davolio	$35,000.00	Brid	$30,000.00
Davolio	$35,000.00	Buchanan	$29,500.00
Davolio	$35,000.00	Martin	$35,000.00
Dodsworth	$48,300.00	Hellstern	$45,000.00
Dodsworth	$48,300.00	Fuller	$32,000.00
Dodsworth	$48,300.00	Brid	$30,000.00
Dodsworth	$48,300.00	Buchanan	$29,500.00
Dodsworth	$48,300.00	Martin	$35,000.00
Patterson	$30,000.00	Brid	$30,000.00
Patterson	$30,000.00	Buchanan	$29,500.00

Less Than [<] link

The result set from a Less Than link includes all records in which the linked field value in the primary table is less than the linked field value in the lookup table. Using the Less Than link, you can compare sales representative and manager salaries in a different direction. Once again, the Salary field in each table is used as the link field. This time, though, you link from the Manager table to the SalesRep table using a Less Than link on the linked Salary fields:

```
SELECT Manager.‘Last Name‘,
     Manager.‘Salary‘,
     SalesRep.‘Last Name‘,
     SalesRep.‘Salary‘
FROM ‘Manager‘ Manager,
     ‘SalesRep‘ SalesRep
WHERE Manager.‘Salary‘ <
     SalesRep.‘Salary‘
```

This SQL statement produces a slightly different table than that produced by the Greater Than link:

Manager Table Last Name	Manager Table Salary	SalesRep Table Last Name	SalesRep Table Salary
Fuller	$32,000.00	Davolio	$35,000.00
Fuller	$32,000.00	Dodsworth	$48,300.00
Brid	$30,000.00	Davolio	$35,000.00

Manager Table Last Name	Manager Table Salary	SalesRep Table Last Name	SalesRep Table Salary
Brid	$30,000.00	Dodsworth	$48,300.00
Buchanan	$29,500.00	Davolio	$35,000.00
Buchanan	$29,500.00	Dodsworth	$48,300.00
Buchanan	$29,500.00	Patterson	$30,000.00
Martin	$35,000.00	Dodsworth	$48,300.00
Hellstern	$45,000.00	Dodsworth	$48,300.00

Less Than Or Equal [<=] link

The result set from a Less Than Or Equal link includes all records in which the linked field value in the primary table is less than or equal to the linked field value in the lookup table. The example here is identical to the example for the Less Than link, except that it uses the Less Than Or Equal link:

```
SELECT Manager.‘Last Name‘,
     Manager.‘Salary‘,
     SalesRep.‘Last Name‘,
     SalesRep.‘Salary‘
FROM ‘Manager‘ Manager,
     ‘SalesRep‘ SalesRep
WHERE Manager.‘Salary‘ <=
     SalesRep.‘Salary‘
```

This SQL statement produces data similar to the following:

Manager Table Last Name	Manager Table Salary	SalesRep Table Last Name	SalesRep Table Salary
Fuller	$32,000.00	Davolio	$35,000.00
Fuller	$32,000.00	Dodsworth	$48,300.00
Brid	$30,000.00	Davolio	$35,000.00
Brid	$30,000.00	Dodsworth	$48,300.00
Brid	$30,000.00	Patterson	$30,000.00
Buchanan	$29,500.00	Davolio	$35,000.00
Buchanan	$29,500.00	Dodsworth	$48,300.00
Buchanan	$29,500.00	Patterson	$30,000.00
Martin	$35,000.00	Davolio	$35,000.00
Martin	$35,000.00	Dodsworth	$48,300.00
Hellstern	$45,000.00	Dodsworth	$48,300.00

Not Equal [!=] link

The result set from a Not Equal link includes all records in which the linked field value in the primary table is not equal to the linked field value in the lookup table. This type of link can be used to find possible combinations of items when a table is joined to itself (a self-join). For example, a company can have a table listing all products they sell. When they decide to hold a sale where their customers buy one item and get the second item half price, they may need a list of all possible two item combinations:

```
SELECT Product1.‘Product Name‘,
     Product2.‘Product Name‘,
FROM ‘Product‘ Product1
     ‘Product‘ Product2
WHERE Product1.‘Product Name‘ !=
     Product2.‘Product Name‘
```

In this SQL statement, the Product table is opened twice. The first time, it is given the alias name Product1. The second time, it is given the alias name Product2. Then the Product Name field is used to link from the Product1 table to the Product2 table. This is the same table, but since it has been opened twice using different aliases, Crystal Reports considers it two separate tables. A Not Equal link is used to link the tables by the Product Name field. As a result, each product is paired with every other product offered, but is not paired with itself:

Product1 **Product Name**	**Product2** **Product Name**
Xtreme Adult Helmet	Xtreme Mtn Lock
Xtreme Adult Helmet	InFlux Lycra Glove
Xtreme Adult Helmet	Roadster Micro Mtn Saddle
Xtreme Mtn Lock	Xtreme Adult Helmet
Xtreme Mtn Lock	InFlux Lycra Glove
Xtreme Mtn Lock	Roadster Micro Mtn Saddle
InFlux Lycra Glove	Xtreme Adult Helmet
InFlux Lycra Glove	Xtreme Mtn Lock
InFlux Lycra Glove	Roadster Micro Mtn Saddle
Roadster Micro Mtn Saddle	Xtreme Adult Helmet
Roadster Micro Mtn Saddle	Xtreme Mtn Lock
Roadster Micro Mtn Saddle	InFlux Lycra Glove

Note: The symbol != is used to represent a Not Equal link, if the ODBC data source driver for the data being accessed supports this symbol. If not, the default symbol <> is used to represent a Not Equal link.

Using SQL and SQL databases

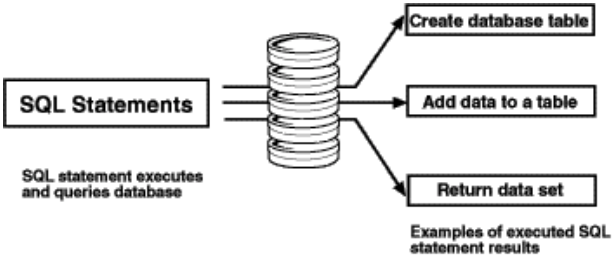
Perhaps the most popular and most powerful database formats are DBMS applications based on the Structured Query Language (SQL). SQL databases usually work over a client/server network architecture, providing an SQL Server to create, store, and manipulate database files, tables, fields and records, and an SQL Client interface allowing workstation users not only to design and work with database files, but also to retrieve useful and meaningful data that will help them in their everyday work.

What is SQL?

SQL is a query language designed for organizing, managing, developing and querying large relational databases over computer networks. SQL is a common language in the Information Science (IS) and Information Management industry. The language has been standardized by the American National Standards Institute (ANSI) and the International Standards Organization (ISO), meaning that there are specific features that must be present in any version of SQL produced by a software company in order for that version to be officially called SQL. Many software vendors add more advanced features to their version of SQL in an effort to improve the language and attract customers, but it must retain the original standards established by ANSI and ISO.

SQL is not a true computer language. It cannot be used to create stand-alone computer applications or operating systems. SQL is often referred to as a sub-language, since it can be used within other languages or applications. Most importantly, the purpose of the SQL language is specific to working with relational databases.

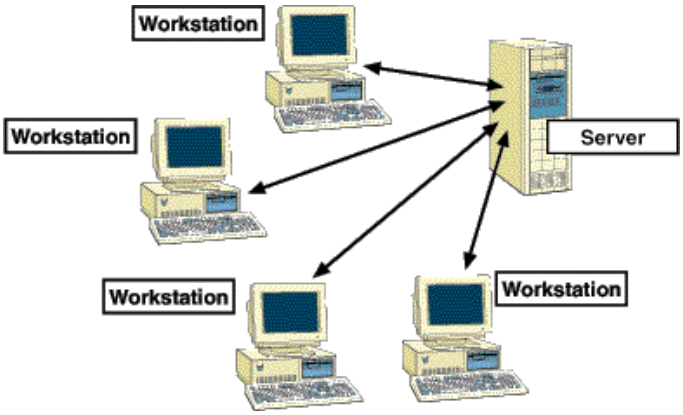
The syntax of the SQL language is built on a system of sending SQL statements to the SQL database server. Each statement is a request to perform a database operation, such as creating a database file, adding tables and fields to a database, adding records to tables, or retrieving data from databases. The SQL server analyzes the SQL statement and performs the required operation. For example, if the statement is a request for data, the server gathers the data and returns it to the client workstation for the user to view.



An SQL query is an SQL statement designed specifically to request data from one or more SQL databases. Some SQL applications require that you type in an SQL query directly using a text editor, while others provide graphical user interfaces that lead you through the process of querying an SQL database. In the latter case, the application must create an SQL statement based on the information you provide. This statement is the actual SQL query, and it is the SQL query that is used to request the data. Crystal Reports falls into both categories of SQL-compliant applications.

Client/server architecture

One of the most powerful features of SQL DBMS applications is their ability to efficiently use the client/server architecture of a network.

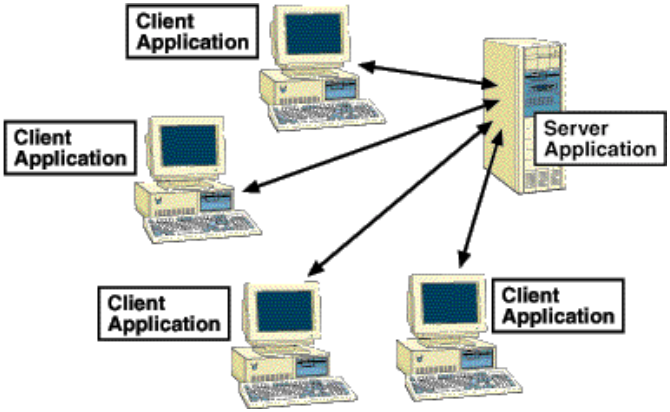


A simple network structure consists of one or more network servers that provide a common location where all users on the network can obtain data and applications. Many network servers also provide network security, automated services such as backing up data, and network resource monitoring to provide the best service possible to all workstations on the network. Because of the high processing demands required by a network server, the computer used as the server is often a high-powered, fast machine that may contain multiple processors, multiple hard drives, and multiple CD-ROM drives.

A network client is a single computer workstation that is used regularly by one or more company employees. A user works on the client and accesses data and applications from the server over the network. Large processing jobs that require a lot of time and resources are handled by the server, and the finished results are sent back to the client. This provides more efficient time management for users because the local workstation has less processing time and more “up” time available to the user.

Many modern computer applications are based on this client/server architecture. A simple client/server application has two parts: a server-based application that is located on a network server machine, and a client-based application that is located on a user’s workstation. The server application handles complex, time-consuming,

or power-demanding processes, taking advantage of the network server's power and resources, while the client application provides an easy-to-use user interface designed to help get the work done faster and better than it could otherwise be done.



Often, a client/server application will be available with a certain number of seats, depending on the cost of the application. Each seat represents either a single client workstation, or a single client user (depending on the software vendors' specifications) that can be connected to the client/server software. Software vendors often sell additional seats for their applications, each seat coming with a complete set of client application software.

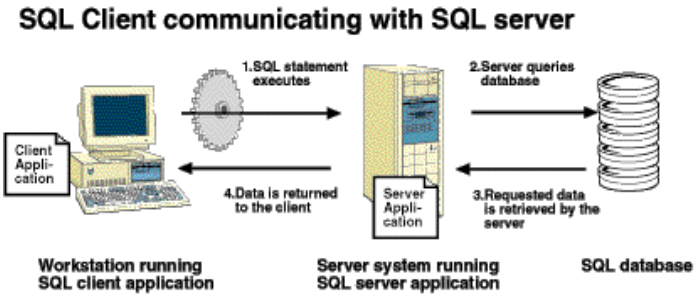
Do not confuse server applications with network server computers. Both are often referred to as servers. However, a server application resides on a network server, taking advantage of the hardware and operating system capabilities of the server machine, while a network server is a physical machine to which network clients are connected by cables or some other connection device.

SQL DBMS

An SQL Database Management System is a common example of a client/server software package. A standard SQL DBMS will include an SQL server application that handles all the actual work of building and working with databases and database data. The DBMS will also include at least one set of SQL client software (one seat) that can connect to the SQL server over your network. SQL client software usually consists of, at the very least, an SQL statement editor that you can use to write and execute SQL statements, and an underlying communications layer that works with the SQL server application over the network.

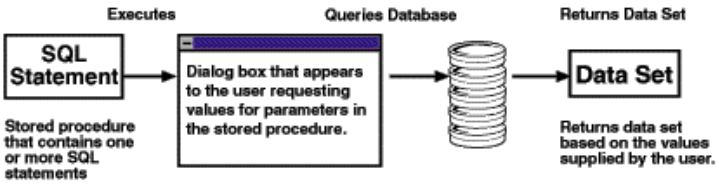
When you execute an SQL statement, the client software passes the statement to the communications layer, which sends the statement over the network to the server software. The SQL server analyzes the statement, performs the requested

operation, and returns any data requested to the client software. If the server returns any data, the client software displays the data to the user.



Stored procedures

In addition to the common relational database attributes (tables, fields, records, and so on) many SQL DBMS systems support stored procedures. A stored procedure is a compiled SQL program consisting of one or more SQL statements. A stored procedure can be used to define an SQL query that you can use over and over again. Furthermore, variables, conditional expressions, and variable arguments can be defined in the stored procedure so that you are prompted to provide information before the procedure is executed.



Since stored procedures can return a result set, they can provide a specific set of data when executed. In fact, Crystal Reports allows you to execute a stored procedure on an SQL database and use the returned data to design a report. If the stored procedure is designed to prompt a user for information to base its query on, Crystal Reports will prompt you for that information when you select the stored procedure for your report.

How does Crystal Reports use SQL?

When you connect to an SQL database, Crystal Reports acts as an SQL client application, connecting to your SQL server through your network.

When you design a report that accesses SQL data, Crystal Reports builds an SQL query. This query can be seen by choosing Show SQL Query from the Database menu.

This SQL query is a representation of the SQL statement that Crystal Reports sends to the SQL server. By interpreting as much as possible from the report design into an SQL query, Crystal Reports can off-load much of the report processing onto the

server machine. Rather than having to sift through an entire database to find the data you requested, Crystal Reports lets the server do the sifting and gets back a much smaller set of data, thus reducing the time and resources your workstation must use in order to finish the report.

The SQL language

Since Crystal Reports uses the SQL language to access client/server databases through ODBC, you can better understand the report generating process by understanding some of the SQL clauses (commands) used:

SELECT

The SELECT clause indicates specific data items to retrieve from the database tables. The item retrieved may be the values in a database field (column), or it may be the result of a calculation performed while gathering the data. For example:

```
SELECT
     TABLEA.‘CUSTNAME‘,
     TABLEA.‘STATE‘
```

DISTINCT

The DISTINCT clause forces the query to retrieve only unique (distinct) sets of data. When using the DISTINCT clause, a row of results will be retrieved only once. The previous SELECT statement can be modified to use the DISTINCT clause:

```
SELECT DISTINCT
     TABLEA.‘CUSTNAME‘,
     TABLEA.‘STATE‘
```

FROM

The FROM clause indicates the sources of the database fields specified in the SELECT clause. FROM lists actual database tables that include the fields and records containing the requested data. The FROM clause generated by Crystal Reports precedes the name of each table with the alias it uses to identify the table in your report. The following example illustrates the FROM clause used with the SELECT clause:

```
SELECT
     TABLEA.‘CUSTNAME‘,
     TABLEA.‘STATE‘
FROM
     ‘TABLEA‘ TABLEA
```

WHERE

The WHERE clause has two purposes:

- To specify record selection criteria.
- To show how two database tables are joined.

When WHERE is used to specify record selection criteria, it includes a search condition in order to determine which records (rows of data) are to be retrieved. For example:

```
SELECT
     MYTABLE.‘SALESPERSON‘,
     MYTABLE.‘SALESTOTAL‘
FROM
     ‘MYTABLE‘ MYTABLE
WHERE
     MYTABLE.‘SALESTOTAL‘ < 10000.00
```

When WHERE is used to specify how two tables are linked, an SQL join operator sits between the two table names. See “Linking options” on page 394.

The following is an example of the WHERE clause joining two tables:

```
SELECT
     CUSTOMER.‘CUST_ID‘,
     CUSTOMER.‘CUST_NAME‘,
     ORDERS.‘AMOUNT‘
FROM
     ‘CUSTOMER‘ CUSTOMER,
     ‘ORDERS‘ ORDERS
WHERE
     CUSTOMER.‘CUST_ID‘ = ORDERS.‘CUST_ID‘
```

ORDER BY

The ORDER BY clause specifies that the database records retrieved be sorted according to the values in a specific field. If you do not use the ORDER BY clause, the program retrieves records in the order in which they appear in the original database. If you specify more than one field after the ORDER BY clause, the program sorts the records according to the values in the first field specified, then, within that sort, the program sorts the records by the values in the second field specified, and so on. The following SQL statement uses the ORDER BY clause:

```
SELECT
     MYTABLE.‘COMPANY‘,
     MYTABLE.‘CITY‘,
     MYTABLE.‘STATE‘
FROM
     ‘MYTABLE‘ MYTABLE
ORDER BY
     MYTABLE.‘STATE‘ ASC,
     MYTABLE.‘CITY‘ ASC
```

Note: ASC indicates that the values in the field are sorted in ascending order rather than descending order (DESC). Ascending order sorts letters from A to Z and numbers from 0 to 9.

GROUP BY

The GROUP BY clause retrieves a set of summary data. Instead of retrieving the data itself, GROUP BY groups the data and summarizes each group with an SQL aggregate function. The server returns only the summarization information for each group to Crystal Reports.

For example:

```
SELECT
     MYTABLE.‘STATE‘,
     MYTABLE.‘ZIPCODE‘,
     SUM (MYTABLE.‘SALES‘)
FROM
     ‘MYTABLE‘ MYTABLE
GROUP BY
     MYTABLE.‘STATE‘,
     MYTABLE.‘ZIPCODE‘
```

Server-side processing

Server-side processing allows you to set up a report that performs the majority of its processing on the server and pushes only relevant details to your computer.

Server-side processing provides you with a number of benefits:

- Less time connected to the server.
- Less memory needed to process the report on your computer.
- Lower transfer time from the server to the client.

Here’s how server-side processing works: by using SQL pass-through technology to send an SQL statement to the database server and retrieve an initial set of data, Crystal Reports off-loads much of the data retrieval and sorting work onto the server system, thus freeing up local memory and resources for more important tasks. That is why server-side processing works only for reports that have been sorted and grouped; if a report has not been sorted and grouped (for example, if it is a simple list report), then there is no processing to push to the server. You should also note that server-side processing works only for reports based on SQL data sources.

Note:

- This description applies only to off-loading grouping and sorting to the server.
- Large amounts of server resources can be used for the temporary databases required when pushing grouping to the server.

Keep in mind that in order to perform the grouping on the server, your report must conform to the following conditions:

- The Perform Grouping on Server option (in the Options dialog box) is enabled. For more information, see “Enabling server-side processing” on page 411.
- The report uses some form of grouping.

- The report is at least partially hidden (at the very least, the Details section must be hidden). Since the server will process those sections that are hidden, the greater the portion of the report that is visible, the greater the amount of processing that must take place on the client side. If the Details section is shown, server-side processing will not be possible.
- In some cases, formula fields must be processed on the client side. If grouping is based on a formula field, or if a formula is used in a summary field, then all the records must be transferred to the client side before the formula can be evaluated. This will increase the amount of time required to run the report. Therefore, you may want to use SQL expressions as an alternative to formulas.

  Note: Formulas used for record selection are an exception and can be pushed down to the server.
- In order for a report to be processed on the server, any running totals appearing in the report must be based on summary fields (since the data needed for the running totals will be pulled over to the client side).
- In order for a report to be processed on the server, the report must not contain Average or Distinct count summaries.
- The report does not contain specified value grouping.

Note:

- When you drill-down on a hidden section of a report, with the processing being done on the server, connection to the server will be automatically initiated. If the client is disconnected from the server (for example, if you download a report onto your laptop and you work on it from a remote location), then drilling-down on data will produce an error since the database is not available.
- If you save a report that has been partially processed on the server using the Save Data with Report option, the program will save only those records that have been transferred to the client side. In other words, if you have drilled-down on a hidden section and there is a tab for that data in Crystal Reports (indicating the data has been transferred), those records will be saved with the report.

How server-side grouping affects the SQL query

When a report pushes most of its processing to the server, this by necessity alters the SQL query. Thus, when the Perform Grouping on Server option is enabled, individual aspects of server-side processing will modify the SQL statement in different ways.

- If you select Use Indexes Or Server For Speed (in the Report Options dialog box), the program adds an ORDER BY clause to the SQL statement and a WHERE clause for the record selection formula, if possible.
- If you group on a linkable data type in the DBMS, the program adds a GROUP BY clause to the SQL statement. The program uses the GROUP BY clause to perform the grouping on the server.

- If you summarize on a linkable data type, the program adds a summary field to the SELECT clause of the SQL statement.
- If you drill-down on a linkable data type, the program adds a WHERE clause to the SQL statement.
- If you group in descending order on a linkable data type, the program adds an ORDER BY clause to the SQL statement.

The statement also varies between tabs:

- If you are working in the Preview tab, the statement includes the GROUP BY clause, as well as any aggregates that the report is pushing to the server.
- If you are drilling-down, the statement varies depending on the underlying data and the level of drill-down. With each drill-down, the WHERE criteria changes. Also, if you drill-down to the details, the statement will not include a GROUP BY clause (since you no longer have any groups on that drill-down tab).

To view the current SQL statement for the active tab, choose Show SQL Query from the Database menu. The Show SQL Query dialog box appears, displaying the SQL statement.

Note: You can use the Formula Workshop to edit SQL expressions to be processed on the server.

Enabling server-side processing

1. On the **File** menu, click **Report Options**.
1. Select **Perform Grouping on Server** on the Report Options dialog box.

   This check box is inactive if Use Indexes or Server for Speed is not selected.
2. Click **OK**.

Note: You can also enable or disable this option quickly by selecting or deselecting Perform Grouping on Server on the Database menu as needed. If Use Indexes or Server for Speed is not selected in the Report Options dialog box, this command is inactive.

Mapping database fields

The Map Fields dialog box allows you to link report fields to their corresponding database fields when you have made changes to the structure of the database, or when you have created a report based on one database from a template of another report based on some other database that has the same table and field structure. In this way, the dialog box helps you to make sure your reports print with the current version of the active database.

When you first create a report, the report draws its fields from the database as it exists at the time. If you change the structure of the database after you create the report, the program needs to adapt the report to the new structure.

About the Map Fields dialog box

The Map Fields dialog box contains four list boxes:

- The upper-left box displays the names of all unmapped report fields (for which the program detects a change in the active database). The top name is selected by default.
- The upper-right box displays the names of unmapped database fields (in which the program detects a change). Since the Match Type box is selected by default, this box displays only the names of unmapped database fields of the same type as the unmapped report field you selected in the upper-left box. To display all unmapped database fields, regardless of type, clear the Match Type box.
- The lower-left box displays the names of mapped report fields. When you map fields in the upper boxes, they appear in the lower boxes.
- The lower-right box displays the names of mapped database fields. When you map fields in the upper boxes, they appear in the lower boxes.

For each database field that you have changed, highlight the report field and the database field in the upper sections and click Map. The field names move from the upper boxes to the lower boxes.

You do not have to remap every report field. For example, if you delete a database field, it is not necessary to remap its corresponding report field, since you will not need that field in your report. You can click OK to close the dialog box without remapping each field listed in the upper-left box.

On some occasions, you might want to unmap or unlink report and database fields that you have already mapped. For example, you might mistakenly map the wrong fields or want to map a report field to a database field that you already mapped. In these cases, you can select a report field in the lower boxes (the program then automatically selects the database field) and click Unmap. The field names will move from the lower boxes to the upper boxes.

Note: Crystal Reports does not automatically refresh the report data when you close the Map Fields dialog box. To refresh your data, click Refresh on the Standard toolbar.

Remapping processes

If you make changes to the active database fields that require you to remap the corresponding report fields, you can display the Map Fields dialog box by choosing any of these commands from the Database Menu:

- Verify Database
- Verify on First Refresh
- Set Datasource Location

Note: There is also a global option called Verify When Database Driver Upgraded (set on Database tab of the Options dialog box) that can make the Map Fields dialog box appear when you first refresh a report's data after update its database driver.

You can use each of these commands for a specific function; however, any of these commands will open the Map Fields dialog box if the program detects a mismatch between the field names in the report and the field names in the database. In order to detect any possible mismatches, the program checks each field name in the report against the field names in the database. If one of the field names does not match any of the field names in the database, the Map Fields dialog box appears.

Note: Report fields that are left unmapped are removed from the report.

Using the Verify Database process

When you choose Verify Database from the Database menu, the program checks the active databases and reports. If it detects changes, the report must be adapted to prevent errors.

The program displays the Map Fields dialog box when it detects either of these types of changes to the database:

- The name of a database field that is used in the report has changed.
- The database has been upsized from a PC data source to an SQL data source.

For a tutorial on the Map Fields dialog box, see "Remapping altered database fields" on page 414. Crystal Reports automatically adapts the report (and does not display the Map Fields dialog box) if it detects any of these changes:

- Fields have been added to the database.
- Fields that are not used in the report have been deleted from the database.
- Field positions have changed in the database.
- Data types have changed for fields in the database.

Using the Verify on First Refresh process

Verify on First Refresh triggers the Verify Database command the first time you refresh your report data per session.

- If there is a check mark beside Verify on First Refresh, the option is active. It triggers Verify Database every time you print. See "Using the Verify Database process" on page 413.
- If there is no check mark beside it, the option is inactive. The option is active by default for new reports.

Using the Set Datasource Location process

When you choose Set Datasource Location from the Database menu and specify a new location for the active database, the program checks the database for changes.

Set Datasource Location displays the Map Fields dialog box when it detects any of the following changes in the database structure:

- A database field has been deleted.
- A database field has been renamed.
- The database is completely new.

Note: The program checks for these changes only if the databases have different names or if the database name has changed. If the databases have the same name, the Map Fields dialog box does not appear, and you need to verify the database when you have finished setting the location. For more information, see "Using the Verify Database process" on page 413.

Remapping altered database fields

Use the Map Fields dialog box to remap existing report fields in the active database if they have been altered.

To remap an altered database field

1. With the report active in the Design tab, choose **Verify Database** from the **Database** menu.

   The Verify Database message box appears.

   - If the program detects no changes in the active database, the message box displays this message: "The database is up to date." In this case, click OK and return to your work.
   - If the program detects a change(s) in the active database, the message box displays this message: "The database file ["table name"] has changed. Proceeding to fix up the report!"

2. Click **OK**.

   If the program detects that a field name has been altered in the active database, the Map Fields dialog box appears.

   Note: The program automatically adapts the report to changes in other data within the database (number of fields, field position, data type, and so on). It is not necessary to remap fields in which these changes have been made.

3. Highlight the first report field that you want to remap in the upper-left box.

4. In the upper-right box, highlight the unmapped database field to which you want to remap the selected report field.

5. Click **Map**.

   The highlighted report and database fields no longer appear in the upper boxes. Instead, they appear in the corresponding lower boxes.

6. Repeat Steps 3 through 5 for each unmapped report field you want remapped.

   Note: If the names of any report fields remain in the upper-left box when you exit the dialog box, the program removes them from your report.

7. Click **OK**.

   The program remaps the report fields to the altered database fields.

Saved Data Indexes

You can increase the performance of a Crystal report by indexing its saved data. When you create a Saved Data Index on a particular field, Crystal Reports can more efficiently filter on that field. In particular, you will achieve considerable performance gains—especially in larger reports—by indexing fields that are referred to by record selection formulas.

Note: The benefits of Saved Data Indexes are largely unnoticeable in reports whose record selection returns fewer than 10,000 records.

How Report Indexing works

With a non-indexed report, Crystal Reports has to look at every record in order to locate values that meet specified criteria. For instance, when a user requests a particular subset of the saved data, or when a user requests the report but only has rights to access certain records, Crystal Reports filters the saved data by checking each record for the appropriate values.

If you have indexed the saved data by one or more fields, however, Crystal Reports already knows which records contain particular values. Consequently, when a user accesses a particular subset of the saved data from the indexed field, Crystal Reports can locate and format the appropriate records more efficiently.

Once you have created Saved Data Indexes, they work entirely in the background. Users don't know that the saved data is indexed, and the grouping, sorting, or formatting of the report doesn't change at all. The indexes merely allow Crystal Reports to locate particular records quickly, without passing through the saved data in its entirety.

Indexing the right field(s)

These guidelines describe the best ways to index saved data and what to avoid when indexing:

- Index fields that users frequently add to their record selection formulas.
- Index fields that are referred to by the report's record selection formula.
- Don't index all of the fields in the report.

  Doing so can result in increased processing times. It is best to index only on the fields that meet the criteria specified above. If all of the fields meet these criteria, then you should prioritize the fields and index only some of them.
- Don't index fields that contain unique values only.

  For instance, don't index a field such as "Last Year's Sales," whose values are likely to be distinct from one another. If you do so, a separate index is created for each and every value in the field.

To index saved data

1. Open your report in the Crystal Reports.
2. On the **Report** menu, click **Report Bursting Indexes**.
3. In the Saved Data Indexes dialog box, select the fields that you want indexed within the saved data.
4. Click **OK** to return to Crystal Reports.
5. If you want to create the index immediately, refresh and save the report.

Unicode support in Crystal Reports

Crystal Reports supports Unicode by converting data from non-Unicode databases as it accesses it (this data conversion happens within Crystal Reports; the data in your database is not affected). Conversion is done using the identifier of the non-Unicode data and the machine's locale setting (usually found in the Control Panel's Regional Settings). To take advantage of the Unicode support in Crystal Reports, ensure the locale setting is correct on each machine that uses Crystal Reports.

For additional information

This chapter has only touched on some of the more important aspects of database access, relational databases, and SQL. If you are interested in learning more about database topics, refer to the documentation provided with your DBMS application.

Note: In addition, there are hundreds of books available on the market that discuss database theory and design in depth. Look for the computer-related section at your local bookstore.

Report Processing Model

A

This appendix provides you with in-depth information about the Report Processing model. This model determines the order in which data is accessed and manipulated during report generation.

Overview

Crystal Reports uses a three-pass reporting method to generate reports. The sections below describe what happens during each step of this process. To see a visual representation, refer to the flow-chart at the end of this appendix.

What is a “pass”?

A pass is a process that Crystal Reports uses each time the data is read or manipulated. Depending on the complexity of the report Crystal Reports may make 1, 2, or 3 passes over the data. This feature allows for complex reporting and formula manipulation.

Pre-pass 1

When previewing a report, the first elements to be evaluated are “constant” formulas. Constant formulas are those that have a constant value for the entire report. They do not change from record to record. For example, 100* 30 would be a constant formula. Constant formulas are evaluated at the beginning of the print generation process and are never evaluated again. This process is known as “BeforeReadingRecords.” If you were to place a constant formula field (i.e., 100*30) in the Details section, the result would be 3000 for each record displayed.

Pass 1

After the “BeforeReadingRecords” process has taken place, Crystal Reports begins reading the database records. During the record reading process, the following will occur:

- Record retrieval. Where possible record selection and sorting are pushed down to the database in this step.
- Evaluation of recurring formulas. These formulas are those that contain database fields but do not contain references to subtotals or summary information. This evaluation time is known as “WhileReadingRecords.” Formulas that contain references to subtotals or summary information are processed in the second pass.
- Application of the record selection locally. If the record selection is too complex to be pushed down to the database, it is applied by Crystal Reports in this step.
- Sorting, grouping, and totaling. In this step, Crystal Reports sorts the records, separates them into groups, and then calculates the subtotals and summaries needed for each group.
- Cross-Tab, chart, and map generation. Only Cross-Tabs, charts, and maps that are based entirely on database fields and recurring formulas are generated in Pass 1. If these objects include running totals and/or PrintTime formulas, they are generated in Pass 2.

- Storage of saved data. After the totaling process is complete, all of the records and totals are stored in memory and to temporary files. Crystal Reports does not read the database again, but instead uses this saved data during all subsequent processing.

Pre-pass 2

During Pre-Pass 2, Crystal Reports orders the groups in the report for Top / Bottom N or Hierarchical Grouping. The records are not read in this process, instead Crystal Reports only looks at group instances from Pass 1, and takes the Top N as appropriate, or orders the groups based on the Hierarchical Grouping settings specified.

Pass 2

Crystal Reports enters the second pass through the data to format pages. The pages are formatted on demand. This means that Crystal Reports will not format a page until it is requested by the user, or until it is required for the total page count in Pass 3.

During page formatting, Crystal Reports does the following:

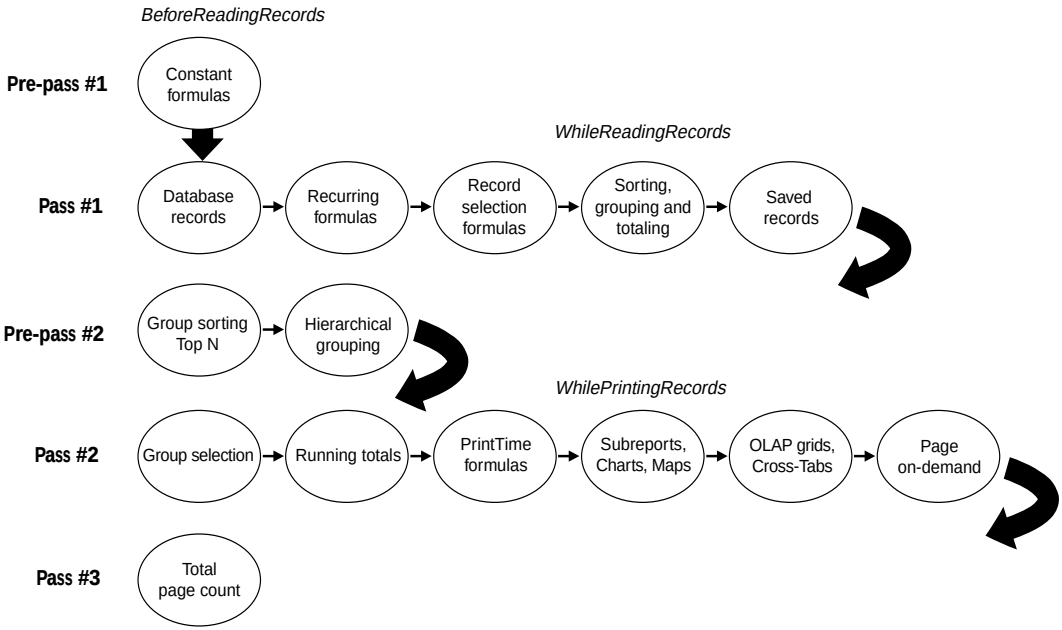
- Group selection formula.
- Running totals.
- Calculation of formulas marked “WhilePrinting Records.”
  These are formulas that contain references to subtotals or summary information, also known as “PrintTime” formulas. This evaluation time is known as “WhilePrinting Records.”
- Cross-Tabs, charts, and maps.
  Cross-Tabs, charts, and maps that include running totals and / or PrintTime formulas, and charts that are based on Cross-Tabs are generated in Pass 2.
- OLAP grids.
- Subreports.
- Generate Pages on Demand.

Note: Subtotals, grand totals, and summaries may appear to be incorrect if the report has a group selection formula. This occurs because the grand totals and summaries are calculated during Pass 1, but the group selection formula filters the data again in Pass 2. Running total fields can be used instead of summaries to total data in reports with a group selection formula.

Pass 3

In the third, and final pass, the total page count is determined. This applies to reports that use the total page count, or Page N of M special fields.

Multi-pass reporting flow chart



Note: Although subreports appear in Pass 2 in the flow chart, you can use on-demand subreports to ensure that your main report remains a single-pass report. With on-demand subreports, Crystal Reports must still make a second pass through the data; however, this second pass does not begin until you drill down on the subreport. Consequently, you can increase the performance of reports that contain subreports by using on-demand subreports.

Glossary

absolute formatting

Formatting that is always applied to an object. See also conditional formatting.

access

To access data means to retrieve data.

Acrobat Reader

An application used to review and print online manuals.

active database

An active database is a database that has been selected for use in a report. You activate databases with the New command on the File Menu and the Database Expert command on the Database Menu.

Active Server Page

Active Server Pages are web pages that run under Microsoft's Internet Information Server (IIS) version 3.0 and later. Active Server Pages combine HTML, VBScript or JScript, and ActiveX controls to create dynamic web pages that can be viewed from any of the most popular web browsers.

ActiveX Control

A Custom Control for Visual Basic 4.0 and above that incorporates the Object Linking and Embedding (OLE) technology. Formerly known as an OLE Control (OCX).

aggregate functions

An operation that summarizes data (sums, calculates an average, identifies a maximum value, and so on). The term "aggregate functions" is often associated with SQL data sources.

alias

An alias is an alternative name assigned to a data source or a database table. If a database is called CUSTOMER.DB, for example, you can assign the alias customer, cust, company, DB1, or any other name that suits your needs. Aliases make it easier for you to use a report created with a database whose name and/or location has changed since the report was created.

area

An area is a group of related sections (i.e., Details A and Details B) that all share the same characteristics but can be formatted differently.

argument

An argument is an item, or one of a group of items, that receives the action of a function. It provides information that the function needs in order to operate. The Truncate function, for example, can not operate by itself. It needs an argument that identifies the item to be truncated. Thus, in the formula in Crystal syntax:

```
Truncate ({orders.ORDER AMOUNT})
```

array

An array is a group of values, separated by commas. Arrays are used with a variety of Crystal Reports functions: Average ([array]), Maximum ([array]), and so on. In these functions, the array is the argument for the function. The function works on the items in the array. Items in an array can be constants, data fields, or formula results.

attribute

An attribute is a quality applied to an object (i.e., font size, color, and so on).

bitmap

A graphic file that can be added to a report.

BLOB fields

A BLOB field is a database field whose data consists of Binary Large Objects—such as bitmap graphics, images, OLE objects, metafiles, and so on. Inserting a BLOB field into your report allows you to access these binary objects as you would other data types.

Boolean

Boolean formulas are formulas that return a Yes/No (TRUE/FALSE) value. For example, the Boolean formula {orders detail.QUANTITY} > 6 compares the value in the {orders detail.QUANTITY} field to 6. If the value is greater than 6 it returns a Yes; if it is 6 or less, it returns a No. Contrast this with a non-Boolean formula like {orders detail.QUANTITY} * 6. In this case the program returns a number, the value of {orders detail.QUANTITY} multiplied by 6. Each selection formula and group selection formula must be Boolean.

Boolean expression

A Boolean expression is an expression that defines a logical relationship between two or more items. A Boolean expression is either TRUE or FALSE. A>5 And B<10 is a Boolean expression that uses the Boolean operator And. For the expression to be TRUE, both conditions (joined with the And operator) must be true. The value of A must be greater than 5 and the value of B must be less than 10. If the values do not fall into those ranges, then the expression is FALSE. Boolean expressions are useful in If-Then-Else formulas. For example:

```
If A>5 And B<10 Then
     "In Range"
Else
     ""
```

This Crystal syntax formula says: if the Boolean expression A>5 and B<10 is TRUE, print “In Range” otherwise (if the Boolean expression is FALSE), print nothing (as designated by the empty string "").

browser

A browser is an application that enables viewing of documents in HTML format.

calculated data field

A calculated data field is a field that holds a value that comes from a calculation rather than coming directly from a database. For example, if the database you are using includes a {file.SALES} field and a {file.COST} field but no Gross Profit field, you can still show gross profit on your report by using a calculated data field. To create a calculated data field, you create a formula that subtracts {file.COST} from {file.SALES}. The formula calculates a Gross Profit value for each row and prints that value wherever you place the formula field.

case-sensitive

Case-sensitive means that a program differentiates between uppercase and lowercase letters when evaluating a text string. Thus, a case-sensitive search for the word “house” will return only the value “house,” but a non-case-sensitive search will return “house,” “House,” “HOUSE,” and similar mixed-case strings. Crystal Reports operators (Equal, In string, and so on) are case-sensitive.

column

A column is the display of data from a single field or formula. Columns run up and down the page. Compare with row.

command

If the database you are using supports a query language such as SQL, you can write your own command which will be represented in Crystal Reports as a Table object. This allows experienced database users complete control of the data processing that gets pushed down to the database server.

comments

Comments are blocks of descriptive text that accompany formulas. Crystal Reports ignores comments when it runs the formula.

concatenate

To concatenate is to join two or more text strings together to form a single contiguous string.

condition

In an If-Then-Else formula, the condition is the If part of the formula, the set of circumstances that must take place (be true) to trigger the Then (or consequence) part of the formula. In the formula *If x<5 Then x Else 5*, the expression *x<5* is the condition.

conditional formatting

Conditional formatting is formatting that applies only if certain situations occur. For example, you can conditionally format numeric database fields to display in red when negative.

conditional formatting formula

A conditional formatting formula is an expression that applies specific attributes to objects or sections only if certain criteria are met.

conditional properties

Conditional properties are properties that are performed on an object only if a comparison statement returns a value of True.

consequence

In an If-Then-Else formula, the consequence is the Then part of the formula; the action that takes place if the If condition is met. In the formula If *x<5 Then x Else 5*, the expression Then *x* is the consequence.

constant

A constant is a value that is fixed and unchanging as opposed to a variable value, which can take on various values depending on the circumstances.

The value 5 is a constant; the value of the Quantity field (which may be 5 sometimes, but may be a different number at other times) is a variable value. For example, in the formula for converting pounds to ounces (Ounces = Pounds * 16), *16* is a constant while *Ounces* and *Pounds* are variables. In the formula (Today - January 1, 1900), *January 1, 1900*, is a constant, while *Today* is a variable that changes whenever the current date changes. In Crystal Reports, constants can be numbers, text strings, dates, dollar amounts, time, date/time, or the result of a formula that itself contains no variables (i.e., 14-9).

container document

A file that contains an embedded or linked OLE object.

conversion interface file

A conversion interface file (.cif extension) is the file in which the program saves the formatting and highlighting from the Document Import Tool. When you highlight something in the Document Import Tool and set properties for database fields, these settings are saved in the .cif file. This file can be used to quickly format the same report later, without highlighting.

cross-tab

A cross-tab is a report that summarizes and presents data in a compact row and column format that makes it easy to compare data and identify trends.

custom function

A custom function is a procedure you create in Crystal Reports to evaluate, make calculations on, or transform data. When you use a custom function in a formula, all the operations in its definition are performed without you having to specify them individually in the formula itself. Thus, the custom function provides a way for you to share and reuse formula logic which, in turn, makes it easier and less time consuming for you and your users to create reports.

database

A database is a bank of related data. Each unit (record) of the database is typically organized in a fixed format to make it easier to retrieve selected portions of the data on demand. Each record is made up of one or more data fields, and each data field can hold one piece of data (known as a value).

data field

A data field (or field) is the basic building block of a record. Each record is made up of one or more data fields, and each data field can hold one piece of data (known as a value). A customer record in a typical customer mailing list database might contain data fields similar to these: Name, Address, City, State, Zip, Phone, Fax. A data field can be empty or contain a value. Data field data is generally displayed or printed in columns in the Details section of a report.

data source

A data source is a database, table, query, dictionary, Info View or stored procedure result set that provides the data for a report.

data type

A data type is a classification of the data that appears in a field or formula. Each piece of data used in a report or formula has one of the following data types: string, currency, number, date, date/time, time, or Boolean (TRUE/FALSE). It is important to understand data types because each function and operator works with only a limited number of data types (often as few as one). For some operators (+ and - for example), the program uses a different set of calculation rules for one type of data than it uses for another.

debug

Eliminating errors that occur when you run a formula.

default

A default is a pre-loaded response to a software request for data. It is the response the computer accepts automatically if you do not enter different data.

Details area

A collection of one or more Details sections (i.e., Details A, Details B, and so on).

Details section

The Details section of a report is the core section of the report. You structure the report in this section by inserting data fields, formulas, and other report elements.

dictionary

A one-stop, ready-to-use source of data that is usually created for end users by computer professionals within the organization. The dictionary takes away the need for the end user to search multiple databases, struggle with links, build formulas, and decode cryptic field names. The user just selects the data he or she needs from the dictionary and builds the report.

divide by zero protection

PCs will not allow you to divide a number by zero. If you attempt such a division, you will get a system error message. To prevent a system error, the program refuses to print a report which contains a formula that divides a value by zero.

drag

Drag can mean different things, depending on the context in which the word is used:

- When referring to moving a field, drag means to click the field frame and, while keeping the mouse button pressed, move the frame to a new position. You release the mouse button when the field is in the desired position.
- When referring to resizing a field, drag means to click one of the field frame handles and, while keeping the mouse button pressed, make the field bigger or smaller. You release the button when the field is the desired size.
- When referring to formatting text, drag means to highlight text by moving the I-beam cursor across it while the mouse button is pressed. You release the button when you have finished highlighting.

Dynamic Link Library (DLL)

A Dynamic Link Library (DLL) is a special kind of file that contains Windows functions. DLLs are used by developers to extend the capabilities of Windows applications. The library is activated whenever an application or another DLL calls a function in the library. DLLs link on the fly, at runtime, whenever an included function is called. DLL functions are available on an as-needed basis to any program that can call DLLs; they do not need to be linked to the program via the compiler. The Crystal Report Engine can be called as a DLL by developers for use with applications they are developing.

element

The word element is used in the documentation to describe individual report components such as database fields, formulas, and group fields. The Design Tab uses rectangular frames to represent fields.

embed, embedded object

An embedded object contains a presentation of the object, all of the data pertaining to the object, and information about the application used to create it. When you modify the original object in the server document, nothing happens to the embedded object unless you specifically update that object.

empty date

An empty date [designated as Date (0, 0, 0)] is a date that contains neither month, day, or year, and thus does not print. Use an empty date in If-Then-Else formulas that either return a date or not. For example, the formula:

```
If PageNumber = 1 Then
     PrintDate
Else
     Date(0,0,0)
```

Prints the print date on the first page and prints nothing on every other page.

Since the Then part of the formula is a date (PrintDate), the Else part of the formula must be a date as well, but a non-printing date. To create a non-printing (empty) date use the Date function and the arguments (0, 0, 0).

empty number

An empty number [designated as zero (0)] is a field value that is typically printed when a value does not meet a specific condition in a numeric If-Then-Else formula. Use an empty number to specify that 0 be printed. For example, in the formula:

```
If {file.FIELD} = 3.5 Then
     {file.FIELD}
Else
     0
```

you are specifying that the numeric Gradepoint be printed (Then) if the grade point is 3.5 or higher. You are using the empty number 0 to indicate that 0 is to be printed (Else) if the grade point is below 3.5. Often a user will format the field that contains this formula to be suppressed if 0. In this event, nothing gets printed in the case of a zero value.

empty string

An empty string (designated as "") is a string that contains no characters. Use an empty string to specify that nothing be printed. For example, in the formula:

```
If {file.FIELD} = 3.5, Then
     "Cum Laude"
Else
     ""
```

you are specifying that the words Cum Laude be printed (Then) if the grade point is 3.5 or higher. You are using the empty string “” to indicate that nothing is to be printed (Else) if the grade point is below 3.5.

evaluation time

Evaluation time refers to the stage in the reporting process when a formula gets evaluated. The three evaluation time functions are:

- BeforeReadingRecords
- WhileReadingRecords
- WhilePrintingRecords

Expert

Crystal Reports offers you several Experts. Experts are tools designed to take you through various aspects of report creation by using a series of tabs. Simply begin at the first tab and proceed through the remaining tabs.

export

To export is to distribute your report to a disk file or through email. Crystal Reports allows you to export your reports in many popular spreadsheet, database, word processor, HTML, and data interchange formats.

field

See data field.

field value

See value.

field width

Field width is the size of the field in the originating database. A field width is generally fixed, and values in the field may take up all or only a part of the allotted width. The program includes Trim functions for removing excess white space from field values that do not fill their respective fields.

file

A file is a collection of related data stored under one name. In Crystal Reports, each report is stored as a single file.

fixed properties

Properties that will always be performed on the object.

flag

A flag is a character or group of characters used to highlight or identify items of interest to call them to the user's attention. For example, in an accounts receivable report, the words "past due" might be printed as a flag beside every past due account.

footer

The footer is the text that appears at the bottom of a report page. Footer text frequently includes page numbers, and may include other information that describes or identifies the report. Crystal Reports gives you the option of printing the footer on all pages or on only selected pages of your report.

form letter

In Crystal Reports, a form letter is a letter that can be reproduced, personalized, and customized using the program's powerful text object capabilities. Form letters generally include both text and field values. You create the letter, and the program runs it each time inserting values from a different record in the database.

formula

A formula is a symbolic statement of the modifications you want performed on certain data before it is printed on your report.

For example, if your report is to contain a {file.SALES} field and a {file.COST} field, you may want to create a GrossProfit field and designate its text strings as {file.SALES} - {file.COST}. This is a simple formula that tells the program to subtract the value of the {file.COST} field from the value of the {file.SALES} field and then to print the result.

You can use formulas to calculate numeric values, compare one value to another and select alternative actions based on the comparison, join multiple values into a single string, and to perform a multitude of other operations. Creating a formula in Crystal Reports is much like creating a formula in your favorite spreadsheet.

Note: The term "formula" in Crystal Reports is equivalent to the term "expression" in Microsoft Access.

Formula Workshop

The Formula Workshop is used to create and edit most kinds of formulas in Crystal Reports. It consists of a toolbar, a tree that lists the types of formulas you can create or modify, and an area for defining the formula itself. The area for defining the formula changes depending on the type of formula you are creating.

formula syntax

Formula syntax is the set of grammar rules you are required to follow when creating formulas using the formula language.

free form

Free form implies that placement of objects is not limited to grids (vertical or horizontal).

function

A function is a built-in procedure or subroutine used to evaluate, make calculations on, or transform data. When you specify a function, Crystal Reports performs the set of operators built into the function without needing each operator specified separately. In this way, a function is a kind of shorthand that makes it easier and less time consuming for you to create reports.

Crystal Reports comes with a wide range of functions, and it also includes tools that allow you to build and save additional functions for yourself.

grand total

A grand total is the summary of all values in a column, for the entire report.

grid

In Crystal Reports, the grid is an underlying network of "lines" that are similar to the lines on graph paper. You can use these lines to help align fields and graphics. If you have the *Snap to Grid* option toggled on in the Options dialog box, Crystal Reports will automatically align any fields you insert or resize to the nearest grid coordinate.

group

A group is a set of records that are related to each other in some way. In a customer list, for example, a group might consist of all those customers living in the same Zip Code, or in the same Region. In a sales report, a group might consist of all the orders placed by the same customer, or all of the orders generated by a particular sales representative. Crystal Reports offers you a great deal of flexibility in the way you group the data on a report.

Group Footer

A Group Footer is a section created by the program whenever you insert a group, a summary, or a subtotal. The Group Footer section is typically used to display the summary or subtotal.

Group Header

A Group Header is a section created by the program whenever you insert a group, a summary, or a subtotal. The Group Header section is typically used to display the name of the group or some other identifying information.

guidelines

Guidelines are non-printing lines that you can use to align, move, and resize objects with precision. Guidelines allow you to work in a free form environment (without a grid), while retaining absolute control over the placing of objects in your report.

header

A header is text that appears at the top of a report page, above the body of the report. While a header can contain virtually any information, it often contains such items as the report title, company name, date, range of dates covered by the report, and so on. Crystal Reports gives you the option of printing the header on all pages or on only selected pages of a report.

Highlighting Expert

The Highlighting Expert is most commonly used for highlighting number or currency field values that are in some way distinguished from the other values in the report. The Highlighting Expert offers a wide range of conditional formatting, including font color, background color, and border style.

You can think of the Highlighting Expert as an advanced formula editor that runs the following equation: If Condition is True, Then Apply These Formatting Specifications.

HTML

The language used by the World Wide Web to publish linked web pages on the Internet.

index

An index is a small file that identifies the location of each record in a database. Since a tiny index file can be searched or sorted much quicker than a large database, Crystal Reports uses index files to speed up the report generation process. In a search, for example, Crystal Reports searches the index for the correct field location. Once found, the program goes directly to the database field. Such a search does away with the need for searching every field of every record in a database. A database may have several indexes, each based on a specific field (or fields).

indexed fields

Fields in the database that are in a specific order to speed up the retrieval of particular records. Instead of searching through all the data in all the records, the program goes first to the index, and finds a pointer that direct it to the specific record it is looking for. Indexed fields are tagged with arrowheads in the Section Expert.

in-place editing

The ability to change an OLE object's properties while in Crystal Reports. The menu items change to provide the editing tools from the server application so that you can make changes easily.

in-place ruler

The ruler that appears when you are editing a text object. This ruler enables you to set tabs, and position objects with precision.

insertion point

The insertion point is a vertical line that indicates the point at which Crystal Reports will insert any text that you type in. You set the insertion point by moving the I-beam cursor to the position you want to insert text and click. When typing text for the first time in a Design Tab section, the program sets the insertion point flush left in the section, regardless of where you click the I-beam cursor.

integer

An integer is a positive or negative whole number or zero. Integers have no decimal places.

link

A link is a field that is common to two or more databases and that serves as a connecting point between those databases. Crystal Reports uses the link to match up records from one database with those from the other(s). For example, if the databases each contain a customer number field (even though the fields might have different names), Crystal Reports can use those fields to electronically connect all records in one database with corresponding records in the other(s). When you create a single report based on multiple databases, the link assures that all the data in each row on that report refers to the same customer (transaction, invoice, and so on).

Note: The term “link” in Crystal Reports is equivalent to the term “relationship” in Microsoft Access.

linked object

A linked object contains a presentation of the object, and a pointer to a defined part of the server document. When you modify the original object in the server document, the links assure that the object in your report is modified automatically as well. Conversely, if you modify the object in the container document, the original object file is modified as well.

live header

A live header is a header that changes dynamically with the content of a field. If you group your data by region, for example, a typical live group header would print the name of the region at the beginning of each group.

Map Expert

The Map Expert is a powerful tool for better organizing your data by placing geographic maps on a report. You can drill-down on these maps to see the underlying data.

nesting

In Crystal Reports, nesting means to use one If-Then-Else expression inside another. For example, If employee's degree is not Ph.D. Then (if employee's sex is male, use the salutation Dear Mr. Else use the salutation Dear Ms.) Else use the salutation Dear Dr. In this example, the nested If-Then-Else statement is surrounded by parentheses. The example says, check the degree field on the employee record to verify that the employee is not a Ph.D. If that condition is true (the employee is not a Ph.D.), then use a letter salutation based on the sex indicated on the employee record. (If the sex is male, Then use a male salutation. Else [if the sex is female] use a female salutation.) Else (that is, if the employee is a Ph.D.), use a Dr. salutation. By using this type of formula construction, you can create a wider set of conditions and a wider set of consequences easier than you could without nesting.

null

Null means there is no value within a database field for a given record. It does not mean zero because zero is a value.

null string

A null string is an empty string. It contains no characters. If you were to use the Count function to count the string, it would return a length of zero. “” is used to designate a null string.

numeric

Numeric data is data on which you can perform arithmetical calculations. The designation numeric refers to the way the data is treated by Crystal Reports and database programs, not to the way the data looks to you.

For example, a serial number 12345 looks numeric, that is, every character is a number. But a serial number is not the kind of data on which you would want to perform arithmetic, so you would probably store a serial number as text instead of as numeric data.

Numeric is one of several data types. Database programs require you to designate a data type when you create a field for use in a database. The data type you select determines the rules the program follows when dealing with the values stored in that field.

object

An object is one of several kinds of report elements that generally contain data and have specific properties that define their behavior or appearance. The program uses the following kinds of objects:

- Field
- Text
- Cross-tab
- Graph
- Subreport
- Picture
- OLE

Each of these objects can be individually formatted, moved, resized, duplicated, and so on.

object frame

An object frame is a rectangular cursor that appears as an aid to placing database fields and formulas on your report. Once you have selected a field or created a formula, the object frame appears. When you move the frame to the place in the report you want the field or formula to appear and click the button, the program inserts the item at the point specified.

ODBC

ODBC stands for Open Database Connectivity. It is an interface that gives applications the ability to retrieve data in data management systems using SQL for accessing the data. Such an interface allows a developer to develop, compile, and ship applications without targeting specific database management systems. Also called interoperability.

OLE

OLE is an acronym for Object Linking and Embedding. It refers to the ability to create compound reports, that is, reports that contain elements from other applications and that can be edited using the original application.

OLE container application

An OLE container application is an application that can contain and process OLE objects created elsewhere (like Paint or Paintbrush). Crystal Reports is a container application.

OLE server application

An OLE server application is an application that can create OLE objects that can then be placed in documents created by container applications. Crystal Reports is a container application, whereas Microsoft Word and Excel are examples of server applications.

on-demand subreport

On-demand subreports appear only as hyperlinks in the main report.

The actual data is not read from the database until the user drills down on the hyperlink. Only data for the on-demand subreport that is actually viewed will be retrieved from the database. On-demand subreports may also be called “real time” subreports in some cases.

one-to-many

One-to-many refers to a situation occurring in linked databases in which one record in one database can be matched with many records in another database. An example of a one-to-many link would occur when linking a customer table to an orders table. In such a case, for every one customer in the primary database, there would typically be many orders in the second (lookup) database.

operator

An operator is a special symbol that describes an operation or an action to take place between two or more values.

The symbol / for example, is an operator that means divide. A / B means Divide A by B. Crystal Reports reads the operators in a formula and performs the actions specified. Crystal Reports contains arithmetic, string, comparison, Boolean, conversion, date, and range operators, among others.

order of precedence

The order of precedence is a set of rules that determines the order in which arithmetic operations take place in a formula that involves multiple arithmetic operations. Multiplication (*) and division (/) are performed first (first tier operations), followed by addition (+) and subtraction (-) (second tier operations). When there are multiple operations involving the same tier, the order of precedence dictates that the operations are performed from left to right. You can use parentheses, if you wish, to alter the normal order.

Page Footer

A section that prints at the bottom of each page. Page footers are typically used for page numbers, chapter names, and other identifying information.

Page Header

A section that prints at the top of each page. Page headers are typically used for titles and other identifying information.

parameter field

A special kind of field that prompts the user for a value. You can use parameter fields for report titles, record selection, sorting, and a variety of other uses. Using parameter fields enables you to create a single report that you can modify quickly to fit a variety of needs.

Note: The term "parameter field" in Crystal Reports is equivalent to the term "parameter queries" in Microsoft Access.

paste

Paste means to retrieve and place data from the Clipboard into a report or formula. The data may have been cut from the same report or formula or from a different one.

population

A population is the entire set of values that might be tested statistically, as opposed to a sample which is a subset of the population. A population does not necessarily refer to a group of people; it can refer to the number of automobiles produced on an assembly line or the number of construction companies bidding on a project.

For example, a real estate agent might sell 20 houses in one year. The population of houses sold by that agent in that year is 20.

population standard deviation

Population standard deviation is a statistical test of how the values in an entire population (all values) deviate from the mean or average value for that population. Population standard deviation is most often used when all values are being evaluated as opposed to just a sample of those values (StdDev).

Note: This comparison simply suggests typical usage. In practice, some users prefer a calculation based on N values (PopulationStdDev) while others prefer a calculation based on N-1 values (StdDev). Both forms of standard deviation are provided by the program.

population variance

Population variance is the square of the standard deviation. It is a measure of the amount by which the values in an entire population vary from the mean (average) value for that population.

Population variance is typically used when all values are being evaluated as opposed to just a sample of those values (Variance).

Note: This comparison simply suggests typical usage. In practice, some users prefer a calculation based on N values (Population-Variance) while others prefer a calculation based on N-1 values (Variance). Both forms of variance are provided by the program.

properties

Properties are qualities that define the appearance or action of an object or a section. There are two kinds of properties:

- **On/Off**
  A property that can only be toggled on or off.
- **Attribute**
  A property for which you have to supply a value.

range

A range is a set of values that fall between and include a defined upper and lower limit. For example, the range 10 to 20 includes 10, 20, and all the numbers that fall between. Also, the range January 1, 1991 to January 30, 1991, includes January 1, January 30, and all the dates that fall between. In Crystal Reports, a range can consist of numbers, dollar amounts, or dates.

record

In a database, a record is a complete unit of related information, an electronic file folder that holds all of the data on a given entity. Each record contains one or more fields that contain the specific pieces of data of interest. In a customer database, for example, a record would store all of the data on a single customer. In an inventory database, a record would store all of the data on a single inventory item. Data from an individual record is displayed or printed as a row of data on a columnar report.

report

A report is simply an organized presentation of data. As a management tool, a report is used to provide management with the insight it needs to run an organization effectively. Crystal Reports allows you to create comprehensive, customized, attractive management reports quickly and easily. But report in Crystal Reports also refers to invoices, form letters, mailing labels, and other related items that require the organization and output of data.

Report Footer

The Report Footer section is the last section of a report in the Design Tab. You can place a summary in this section that you want to appear only on the last page of a report.

Report Gallery

The Report Gallery is a special dialog box that appears when you click the **New** button on the Standard toolbar or choose the New command from the File menu. The Report Gallery serves as a gateway to all the report creation experts and to the graphical interface for selecting the report and data type for creating custom reports.

Report Header

The Report Header section is the first section of your report in the Design Tab. You can place a title in this section, or any data you want to appear only on the first page of your report.

Report Parts

Report objects displayed by themselves in a viewer—without the rest of the report page—are referred to as Report Parts.More precisely, however, Report Parts are hyperlink definitions that point from a home report object to a destination object.

repository

The Crystal Repository is a central location for you to store and manage your report objects. Data definitions such as custom functions and custom SQL commands can also be stored and maintained in the repository. These objects are then accessible to users and report developers for use in new reports that can be distributed throughout your company.

request

A request is a set of criteria that specifies the subset of data that you want to use for your report. For example, if you want your report to contain only California data, you can create a record selection request that retrieves only California records for your report. You create record and group selection requests using the Section Expert.

returns

The word “returns” refers to the result of a function, an operator, or a formula. For example:

- When using a function, it performs a calculation or manipulation that results in a data change of some kind. The data that results is what the function returns. For example, Average(1, 2, 3, 4, 5) returns the average of the array 1, 2, 3, 4, 5. Truncate(1.2345) returns the integer (whole number) portion of the number 1.2345.
- When using an operator, the result of the operation using that operator is what the operation returns. For example, 5*6 equals 30. You can say that the operation 5*6 returns 30. Also, the operation 100<200 compares the two values and returns True; 200<100 compares the two values and returns False.
- When using a formula that contains functions or operators, each function or operation within the formula returns a result, but the formula taken as a whole returns a result too. When talking about a formula, it is the result of the formula that is of interest, not the result of individual functions or operations. For example, in the following formula:

  ```
  If {file.QTY} < {file.REORDERAMOUNT} Then
       "Reorder "
  Else
       ""
  ```

  an internal operation compares the value of the {file.QTY} field with the value of the {file.REORDERAMOUNT} field. If {file.QTY} is less than {file.REORDERAMOUNT}, that individual operation returns the value True, but that is not what the formula taken as a whole returns. The formula, taken as a whole, returns the flag “Reorder” when the operation internally returns the value True.

row

A row is the display of data from a single record. Rows run horizontally across the page. The words row and record are sometimes used interchangeably in this manual. Contrast with column.

ruler

The ruler provides a visual reference for positioning and resizing fields, graphs, lines, boxes, and bitmaps. The increments on the ruler are based on your measurement settings in the International section of the Windows control panel. The ruler also enables you to change page margins while immediately seeing the results of your changes on the report itself.

The Ruler is visible in both the Design Tab and Preview Tab when their respective check boxes are toggled on in the Options dialog box.

running total

A running total is a total that is displayed generally on a record by record basis. It totals all records (in the report, in the group, and so forth) up to and including the current record. For example, if your first three records have values of 2, 4, and 6, the running total for each of the three records would be 2, 6, and 12, respectively.

sample

A sample, as used in statistics, is a subset of a population used to represent the entire population. Researchers frequently do not have the option of testing an entire population before forming conclusions based on their tests. In such cases, they use a sample to represent the whole.

For example, political polling before elections is often based on questioning only four or five hundred people. From the answers given by this sample, predictions can be made on how an entire nation will vote.

scroll bars, scrolling

Sometimes a window can display only a portion of a document. In such a case, the window includes scroll bars that you can use to move other parts of the document into the window for your review.

Scroll bars also appear with lists that are longer than the available window. The scroll bars allow you to move back and forth through the list. The process of moving through a list or document using scroll bars is called scrolling. In Crystal Reports, the screen automatically scrolls whenever you move the cursor outside the window and press and hold down the button.

section

A section is a part of the report design environment. The program divides the design environment into several sections, each of which has different printing characteristics. You place objects in the various sections to build a report.

select

- With regard to a report element (data field, formula, and so on), select means to point to the element and then click to choose the element as the object of the next menu selection.
- With regard to text, select means to highlight the text by dragging the I-beam cursor over it.
- With regard to records, select means to identify and choose those records of interest while disregarding all others.
- With regard to groups, select means to identify and choose those groups of interest while disregarding all others.

selection formula

A selection formula is a formula that specifies the records, or groups of records, you want included in your report.

server document

A server document is a file that stores the original OLE object.

server-side processing

Server-side processing is a feature that allows you to set up reports that perform the majority of their processing on the server. These reports push only relevant details to your computer, thus saving you time and memory.

shortcut menu

A dynamic menu available in the Design and Preview Tabs. Access the shortcut menu by highlighting an object and right-clicking.

snap property

Snap is a “magnetic” property that attracts nearby objects. Crystal Reports uses two facilities that have the snap property: Guidelines and the Grid. Whenever an object is moved close to a guideline or a grid coordinate, the program snaps it into position for accurate placement and alignment.

sort-and-group-by field

A sort-and-group-by field is a field that triggers the printing of a subtotal (or a group field value) whenever its own value changes.

For example, you may have a customer report that contains the {customer.CUSTOMER ID} and {orders.ORDER AMOUNT} fields. If you want to subtotal by customer (total the orders for each customer), click the {orders.ORDER AMOUNT} field as the field to subtotal and the {customer.CUSTOMER ID} field as the sort-and-group-by field. Crystal Reports sorts the data by customer, so that all orders from the same customer are grouped together. Then, whenever the value in the {customer.CUSTOMER ID} field changes (when it changes from one customer to a different customer), Crystal Reports prints a subtotal of the values in the {orders.ORDER AMOUNT} field (a total of orders for the individual customer). You can also use sort-and-group-by fields to trigger summaries.

sort direction

Sort direction describes the way records or groups are printed in your report. They are printed either in ascending (A to Z, 0 to 9), or descending (Z to A, 9 to 0) order.

sort field

A sort field is a data field on which the sort procedure is based. A mailing list, for example, could be sorted, in ascending order, on the {customer.POSTAL CODE} field; that is, the customers would be sorted so that those with the lowest postal codes would appear first and those with the highest postal codes would appear last. The report could also be sorted in ascending alphabetic order, on the {customer.CONTACT LAST NAME} field; that is, customers with last names beginning with A would appear first and those with last names beginning with Z would appear last.

sort order

Sort order is an indicator of the direction in which you want your data to be presented once it is sorted. Data is typically printed in one of two sort orders: ascending (lowest to highest, earliest to latest, first to last, A to Z, and so on) or descending (highest to lowest, latest to earliest, last to first, Z to A, and so on).

sorting

Sorting is a method of organizing the order in which data appears on your report. Crystal Reports provides you with powerful tools for sorting your report data.

SQL

SQL stands for Structured Query Language; a system for managing, organizing, and retrieving data stored on a computer database. Structured Query Language is a computer language that enables you to interact with a specific type of database called a relational database.

SQL pass-through

The ability to get the SQL Server to process the data retrieval criteria in order to pass the smallest possible result set back to Crystal Reports for final processing. When processing can be passed-through to the server, it makes the reporting process more efficient and it minimizes network traffic.

standard deviation

Standard deviation is the square root of the variance. It is a statistical test of how various values in a set of values deviate from the mean or average value for that set. You can use standard deviation, for example, for assessing the relative difficulty of tests given to students, for evaluating and projecting customer purchase patterns, or for comparing the results delivered by two or more products under evaluation (laboratory blood tests, smoke detectors, radar detectors, and so on). The uses are endless.

Standard deviation (as opposed to population standard deviation) is typically used to project the standard deviation for an entire population (all values) based on testing only a small sample of that population. For example, a company producing batteries with a new manufacturing process might want to test the batteries to determine how long they will last before they go dead. If the company tested all of its batteries, it would have no product left to sell. As an alternative, the company might test thirty batteries selected at random and project the mean burn out time and standard deviation for all batteries based on the results from that thirty battery sample.

Note: This comparison simply suggests typical usage. In practice, some users prefer a calculation based on N values (PopulationStdDev) while others prefer a calculation based on N-1 values (StdDev). Both forms of standard deviation are provided by the program.

static OLE object

A static OLE object is a picture of an object that is stored in a document when it is saved. The picture can be displayed or printed by a user who does not have the application in which the original object was created. The object can not be edited in place, however, without first converting it to an editable type of object. Static OLE objects offer better online and print performance than do standard bitmaps.

string

A string is a series of connected characters (letters, numbers, symbols, spaces) stored and used as text. The word “hello” is a text string as is the phrase “Order # 2453“ and the customer number “B30-124-777”. Strings are sometimes referred to as text strings or character strings.

subreport

A subreport is a report within a report. It has all of the characteristics of a report with one exception: it can not itself include a subreport. Subreports can be free-standing or they can be linked to the data in the primary report. Crystal Reports enables you to insert as many subreports as you wish.

substring

A substring is simply a part of a larger string. “Columbia” is a substring of the string “British Columbia”, “1040” is a substring of the customer number “B-1040-0032456”, and “B” is a substring of the string “President Bill Clinton”.

subtotal

A subtotal is a partial total, a total of a specific, limited group of data in a field. For example, given the following data:

```
1, 2, 3, 4, 5, 6, 7
```

a subtotal after the 3 produces the value 6 (1 + 2 + 3). A second subtotal after the 6 produces the value 15 (4 + 5 + 6).

A subtotal is the sum of all values from a single field, from all the records in a group. In a sales report, for example, if you subtotal the amount ordered by sales representative, Crystal Reports gathers all the records that belong to the sales representative and totals the amounts ordered from all the records.

summary

A summary is the value generated as the result of an evaluation, a tally, or a calculation performed on data from a single group.

In a group average, Crystal Reports averages the values in a group of records; in a group count, it counts the values in a group of records, and so forth. Summary values are important tools for creating powerful reports.

summary field

A summary field is a field that determines the sum of the values, the average value, the maximum value, the minimum value, or count of values in a group of values in a given field. Much like a subtotal, a summary field groups data to your specifications and then performs the requested calculation / determination.

suppress

If you suppress something it does not appear. In Crystal Reports you can suppress report sections (headers, details, and footers), rows and columns that don’t contain a value, as well as specific values.

syntax

Syntax is a set of rules used to make a correct formula. When creating formulas, you have the option of using Crystal or Basic syntax.

tabs

Tabs are used in many dialog boxes and Experts in Crystal Reports. Tabs resemble the tabs on common file folders. Tabs always have text on them to indicate what you will find on the tab.

template

A template can be a copy of a report used as the starting point for creating a new report, or it can be a report whose formatting is applied to a new report created in the Standard Report Creation Wizard. When a report is used as a template, your original report remains unchanged.

text object

A text object is a specialized object that can contain text, database fields, and formula fields. It contains its own mini word processor that can be used for anything from adding a label to creating an entire document.

toolbar

A bar at the top of Crystal Reports application window which contains a number of buttons that you can click to activate the most frequently used commands.

total

A total is a sum of values. See also grand total, running total, and subtotal.

truncate

Truncate means to cut off or eliminate all data that comes after the decimal point. Thus, if you truncate 1.2345, you get the value 1. If you truncate the value 1.9999 you also get the value 1. Truncate does not round data, it simply cuts off unwanted data.

two pass formula/function

A two pass formula is a formula that requires two passes through the data for completion. The first pass performs some calculation or selection and the second pass performs a calculation or selection that uses the result generated by the first pass.

An example of a two pass formula is one that calculates the sales for each sales representative as a percent of total company sales. The first pass sums the sales for each representative to arrive at total company sales. The second pass divides the sales per representative by total company sales to calculate the percent of total sales.

underlay

The ability of an object (a bitmap, a graph, and so on) to print beneath multiple sections which follow the section in which it was placed. For example, you can place a bitmap in one section, format the section to underlay the following sections and then expand the bitmap so it appears as a background for the entire page of your report.

value

A value is the data found in a field. In a field called {customer.CONTACT FIRST NAME}, for example, John or Mary might be the value. In a field called {orders.ORDER AMOUNT}, 1234.55 or $200 might be the value.

variance

Variance is the square of the standard deviation. It is a measure of the amount by which all values in a group vary from the mean (average) value in the group. It is a statistical test that can be used to evaluate the variability in a group of values (for example, the amount bid by each of the bidders on a construction project).

Variance (as opposed to PopulationVariance) is most often used to project the variance for an entire population (all values) based on testing only a small sample of that population. For example, with a limited number of bids in on a construction project, you might want to project the variance for all bids based on the sample already in. Or, based on sales figures for the first three months of the year, you might want to project the variance for orders for the entire year (including the nine months yet to come).

Note: These comparisons simply suggest typical usage. In practice, some users prefer a calculation based on N values (PopulationVariance) while others prefer a calculation based on N-1 values (Variance). Both forms of variance are provided by the program. For a more thorough discussion on the use of variance, consult any reliable statistics text.

verify

In Crystal Reports terms, verify does not mean to repair and compact the database (MDB file) in Access. It means to let the report understand the changes made to the database structure (fields and tables, NOT records).

wildcard

A wildcard is a character that represents any character (?) or any group of characters (*) in a search string. For example, if you are searching for Dan*, the search string will return strings like Danny and Daniel.

wizard

Crystal Reports includes a number of wizards that help you create reports with step-by-step instructions. Wizards do not use a system of tabs; instead they present you with a series of screens. Simply follow the instructions in the screens that appear and the wizard leads you through otherwise complex procedures with a minimum of effort.

word wrap

Word wrap is a word processor-type property of a text object that automatically moves a word to the following line when the word is too long to fit the remaining space on the current line.

Index

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Z