

HYPERION [®] SYSTEM TM 9 BI + TM INTERACTIVE REPORTING TM RELEASE 9.01

USER'S GUIDE



Copyright 1999-2005 Hyperion Solutions Corporation. All rights reserved.

"Hyperion," the Hyperion "H" logo, and Hyperion's product names are trademarks of Hyperion. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

No portion hereof may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the recipient's personal use, without the express written permission of Hyperion.

The information contained herein is subject to change without notice. Hyperion shall not be liable for errors contained herein or consequential damages in connection with the furnishing, performance, or use hereof.

Any Hyperion software described herein is licensed exclusively subject to the conditions set forth in the Hyperion license agreement.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Hyperion license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14, as applicable.

Hyperion Solutions Corporation 5450 Great America Parkway Santa Clara, California 95054

Printed in the U.S.A.

Contents

	Preface
	Purpose xiv
	Audience xiv
	Document Structure xiv
	Where to Find Documentation
	Help Menu Commandsxv
	Conventions xvi
	Additional Support xvii
	Education Services xvii
	Consulting Services xvii
	Technical Support xviii
	Documentation Feedback xviii
	Introducing Interactive Reporting Studio 21
	Interactive Reporting Studio Features
	How Interactive Reporting Studio Works
	Relational Databases 23
	Multidimensional Databases
	About Interactive Reporting Documents
	Data Source Connections
	Data Models
	Queries
	Analysis and Reporting
	Turning Data into Information
CHAPTER 2	Interactive Reporting Studio: A Tutorial
	Starting the Interactive Reporting Studio
	Opening and Saving a Sample File
	Looking at a Simple Query
	Viewing Database Tables 31

	Viewing Results	32
	Reordering Columns	32
	Sorting Columns	32
	Filtering Data	33
	Calculating Data	33
	Pivoting Data	34
	Calculating Totals and Subtotals	35
	Drilling Down	35
	Hiding Data	36
	Adding Color	36
	Charting Data	36
	Sorting Charts	37
	Inserting Text Labels	37
	Designing Reports	38
	Creating Smart Reports	39
	Sorting Columns	39
	Setting Up a Report Page	39
CHAPTER 3	Interactive Reporting Studio Basics	41
	Using the Interactive Reporting Workspace	42
	Main Menu	42
	Toolbars	42
	Section Title Bar	43
	Command Lines	43
	Data Layout	43
	Section Pane	43
	Catalog Pane	44
	Content Pane	44
	Status Bar	44
	Managing Documents	45
	Maintaining Documents	45
	Saving Documents	45
	Specifying Save Options	46
	Working with Data Sources	48
	Exporting Data	50
	Exporting to Microsoft Office HTML File Formats	56
	Formula Mappings	62
	Custom Formulas	67
	Exporting a Document as a Web Page	69
	Printing Documents	72
	Sending Documents	73
	Quitting an Interactive Reporting Studio Session	73
	Quitting the Interactive Reporting Web Client	74
	File Menu Command Reference	75

	Using Edit Commands	. 75
	Changing Workspace Views	. 76
	Inserting Sections and Breaks	. 77
	Formatting Text and Other Elements	. 78
	Formatting Numeric Data Types	. 79
	Changing Numeric Formatting	. 79
	Displaying Numbers in Scientific Notation	. 79
	=Working with Document Sections	. 81
	Understanding Document Sections	. 81
	Adding Sections	. 82
	Viewing Sections	. 82
	Moving Between Sections	. 83
	Duplicating Sections	. 83
	Renaming Sections	. 83
	Adding Headers and Footers to Sections	. 84
	Deleting Sections	. 85
	Setting Options	. 86
	Specifying Default Formats	. 86
	Selecting Program Options	. 88
	Document Properties	. 90
	Customizing Menus	. 92
	Tools Menu Command Reference	. 93
	Interactive Reporting Web Client Co-existence	. 94
	Locally Saved BQYs	. 94
	Interactive Reporting Web Client Mismatches	. 95
CHAPTER 4	Querying Relational Databases	. 97
	Query Section	. 98
	Using Data Models in the Query Section	. 98
	Building Queries	. 98
	Working with Items on the Request Line	. 99
	Processing Queries	100
	Query Processing Order	101
	Saving Queries	102
	Cancelling Queries	102
	Building Subqueries	103
	Regular Subqueries	103
	Correlated Subqueries	104
	Derived Tables	106
	Derived Tables Rules and Behavior	107
	Derived Tables and SQL	109
	Working with Query Section Data	111

	Processing Results to a Database Table	111
	Estimating Query Size	112
	Displaying Database Remarks	113
	Preaggregating Data Using Functions	113
	Appending Queries	116
	Using Local Results	117
	Using Stored Procedures	118
	Setting Query Options	119
	Importing Data Files	120
	Setting Data Type Properties	120
	Query Menu Command Reference	122
CHAPTER 5	Working with Query Results	123
	Results Section	124
	Understanding Data Types	124
	Enhancing Your Results Set	125
	Filtering Results	125
	Sorting Results Data	125
	Adding Computed Items to Results	125
	Applying Data Functions to Results	126
	Adding Grouping Columns	127
	Modifying Grouped Columns	128
	Adding Columns Automatically	128
	Breaking Out Dates	128
	Working with the Results Table	129
	Selecting Columns and Rows	129
	Deleting Columns	129
	Formatting Commands	129
	Saving Results Sets	120
	Error anti- a Darralt Cata	121
		151
	Results Menu Command Reference	132
CHAPTER 6	Working with Tables	133
	Table Section	134
	Creating Tables	134
	Tables as a Data Staging Area	134
	Manipulating Table Data	135
	Filtering Data in a Table	135
	Sorting a Table	136
	Adding Computed Items	137
	Adding Grouping Columns	137
	Modifying Grouping Columns	139
	Adding Date Groups	139
	Applying Data Functions to Tables	139
	Working with Table Components	140

	Selecting Columns and Rows	141
	Deleting Columns	
	Table Menu Command Reference	
CHAPTER 7	Querying Multidimensional Databases.	
	OLAPQuery Section	
	Defining OLAPQuery Options	
	General OLAPQuery Options	145
	Database-specific OLAPQuery Options	
	Building OLAP Queries	
	OLAPQuery Section Data Layout Rules	149
	OLAPQuery Member and Level Rules	150
	Refining OLAPQuery Data	151
	Specifying a Slicer	151
	Drilling Down	152
	Drilling Up	152
	Hybrid Analysis and Drilling (Analytic Services and DB2 only)	152
	Adding Computed Items	153
	Using OLAPQuery Functions	154
	Processing OLAP Queries	157
	Processing OLAP Queries Automatically	
	Working with an OLAPQuery Offline	
	Creating a OLAPResults Section Automatically	
	Applying Filters	
	Applying Member Selection Filters	
	Applying Measure Filters (Analytic Services)	
	Applying variable Filters	
	Changing Data Views	
	Suppressing Rows	
	Adding Totals	
	Showing OI AP Results as a Chart	105
	Earmantting OLA DOugray Itames	105
	Drilling Infough from a Multi-Dimensional Database to a Relational Database	
	Drilling Through	10/
	OLAP Menu Command Reference	
CHAPTER 8	Analyzing Data with Pivot Tables	171
	Pivot Section	
	Creating a Divet Table	
	Divoting Data	1/2
	Charting a Pivot Table	1/3
	Working with Divot Tables	1/3
	working with Pivot ladies	1/3

	Selecting Pivot Table Elements 174
	Moving Pivot Table Elements 174
	Changing Label Nesting Levels 174
	Sorting Pivot Tables
	Using Data Calculations in Pivot Tables
	Adding Totals and Subtotals 175
	Adding Cumulative Totals 176
	Using Data Functions 177
	Using True Computed Item Totals
	Using Surface Values in Data Functions 179
	Using Weighted Averages
	Adding Computed Items
	Pivot Section Aggregation and Surface Values
	Fact Detail Cells
	Pivot Fact Detail Cell Behavior
	Pivot Section Computed Items
	Computed Item Detail Cell Values
	Pivot Menu Command Reference
CHAPTER 9	Charting Data
	Chart Section
	Charting Basics
	Chart Terminology
	Understanding Chart Dimensions
	Using the Chart Data Layout
	Creating Charts
	Selecting a Chart Type
	Working with Two-dimensional Charts 206
	Using Pie Charts to Analyze Data 206
	Using Two-dimensional Bar Charts to Analyze Data 200
	Working with Multidimonoional Charts
	About the 3 D View 200
	About the 5-D view
	Understanding Clustered Per Charts
	Understanding Stacked Bar Charts
	Understanding Area Charts
	Understanding Pibbon Charts
	Understanding Line Charts 212
	Understanding Combination Charts 212
	Manipulating Chart Data 212
	Using Different Scales to Compare Polated Values
	Using Data Functions in Charts
	Adding Computed Items
	Northing L hart Itoma

	Creating Pivot Tables from Charts	. 217
	Drilling into Charts	. 218
	Hiding and Focusing on Charted Data	. 218
	Restoring Hidden Items	. 219
	Working with Chart Elements	. 219
	Selecting Chart Elements	. 219
	Displaying Axis Grid Lines	. 220
	Inserting Text	. 220
	Changing Chart Legends	. 220
	Chart Scrolling and Scaling	. 222
	Scrolling through Chart Data	. 222
	Rotating and Elevating Charts	. 223
	Smart Scaling	. 223
	Initial Display	. 224
	Auto Resizing Charts	. 226
	Fitting Charts to Screen	. 227
	Customizing Chart Properties	. 227
	Setting General Chart Properties	. 228
	Customizing Chart Patterns and Labels	. 232
	Customizing Chart Colors	. 234
	Chart Menu Command Reference	. 237
CHAPTER 10	Report Designer	. 239
CHAPTER 10	Report Designer	. 239
CHAPTER 10	Report Designer Report Section Report Section Elements	. 239 . 240 . 240
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar	. 239 . 240 . 240 . 241
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line	. 239 . 240 . 240 . 241 . 241
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report	. 239 . 240 . 240 . 241 . 241 . 241 . 242
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components	. 239 . 240 . 240 . 241 . 241 . 242 . 242 . 243
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options	. 239 . 240 . 240 . 241 . 241 . 242 . 243 . 244
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report	. 239 . 240 . 240 . 241 . 241 . 241 . 242 . 243 . 244 . 245
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups	. 239 . 240 . 240 . 241 . 241 . 242 . 243 . 243 . 244 . 245 . 245
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers	. 239 . 240 . 241 . 241 . 242 . 243 . 243 . 244 . 245 . 245 . 246
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 246 . 246 . 246
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks	. 239 . 240 . 241 . 241 . 242 . 243 . 243 . 244 . 245 . 245 . 246 . 246 . 247
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 246 . 246 . 247 . 247
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields	. 239 . 240 . 241 . 241 . 242 . 243 . 243 . 244 . 245 . 245 . 245 . 246 . 247 . 247 . 248
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 246 . 246 . 247 . 247 . 248 . 250
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields Working with a Report Page	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 246 . 246 . 247 . 246 . 247 . 248 . 250 . 251
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields Working with a Report Page Displaying Rulers	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 246 . 246 . 246 . 247 . 247 . 248 . 250 . 251 . 251
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields Working with a Report Page Displaying Rulers Using Grids	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 247 . 247 . 247 . 247 . 248 . 250 . 251 . 251 . 251
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields Working with a Report Page Displaying Rulers Using Grids Using Design Guides	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 246 . 246 . 246 . 246 . 247 . 248 . 250 . 251 . 251 . 251 . 251
CHAPTER 10	Report Designer Report Section Report Section Elements Report Section Toolbar Expression Line Creating a Custom Report Report Components Table Column Formatting Options Inserting Additional Tables in a Custom Report Adding Report Groups Inserting Report Headers and Footers Inserting Page Headers and Footers Inserting Page Breaks Adding Other Report Elements Working with Fields Modifying Filter Fields Working with a Report Page Displaying Rulers Using Grids Using Design Guides Setting Up a Report	. 239 . 240 . 241 . 241 . 242 . 243 . 244 . 245 . 245 . 245 . 246 . 247 . 246 . 247 . 247 . 248 . 250 . 251 . 251 . 251 . 252

	Specifying Page Size	252
	Specifying Page Margins	252
	Setting Up Page Columns	253
	Enhancing Report Data	254
	Sorting Report Items	254
	Adding Computed Items	255
	Applying Data Functions	256
	Applying Break Totals	256
	Using True Totals for Computed Items	257
	Hiding and Focusing on Reported Data	258
	Using Multiple Data Sources in a Report	259
	Creating Smart Reports	261
	Formatting Report Items	261
	Converting Detail Reports from Versions Earlier than 6.0	262
	Display Differences	262
	Conversion of Detail Report Categories	263
	Conversion of Data Area	263
	Conversion of Facts	263
	Conversion of Smart Reports	263
	Conversion of Graphic Objects	264
	Report Menu Command Reference	264
		0.67
CHAPIER 11	Using Filters	267
	About Filters	268
	Server versus Local Filter Processing	268
	Filter Line	269
	Filter Controls	270
	Setting Simple Filters	273
	Filtering Queries	274
	Removing a Query Filter	275
	Filtering Results	275
		275
	Setting Compound Filters	276
	Setting Variable Filters	277
	Customizing Filters Options	277
	Filtering Data in a Table Report	278
	Multiple Filters and the Meta Topic	279
	The Two-Tier Strategy	279
	Filtering Computed Items	280
CHAPTER 12	Working with Computed Items	283
	About Computed Items	200
	Computing New Data	204 284
	Computing New Data	204 284
		204

Computed Items and Data Functions	
Pivot Options	
Adding Computed Items	
Operators	
Using Functions	
Scalar Functions	
Teradata Version 3 OLAP Functions	
Functions for Returning the Day of the Week	
Common Computed Item Examples	
Math Functions	
Central Tendency	
Calculated Averages	
Percentile	
Rank	
Project Sales	
Scalar Function Examples	
Avg	
AvgNonNull	
ColMax	
ColMin	
Count	
CountDistinct	
CountNull	
CountNonNull	
Cume	
Sum	
Median	
Mode	
Percentile	
Rank	
RankAsc	
StdDev	
StdDevp	
Var	
Varp	
Trend Functions	
General Moving Average and <i>Moving Function</i> Functionality	
Simple Moving Averages	
Positioning of Moving Average Results - Trailing and Centered Averages	
weighted Moving Averages Europential Moving Averages	
Exponential Moving Averages	
Moving Dill	
Moving Median	

	Moving Sum	334
	Moving Minimum	335
	Direction Of Moving Function Calculation	336
CHAPTER 13	Applying Sorts	339
	Sorting Data	340
	Simple Sorts	340
	Sort Lines	340
	Complex Sorting	341
	Complex Sorting in the Query, Results, and Table Sections	341
	Complex Sorting in Chart, Pivot, and OLAPQuery Reports	342
	Glossary	345
	Index	355

Preface

Welcome to the *Hyperion System 9 BI+ Interactive Reporting User's Guide*. This preface discusses the following topics:

- "Purpose" on page xiv
- "Audience" on page xiv
- "Document Structure" on page xiv
- "Where to Find Documentation" on page xv
- "Help Menu Commands" on page xv
- "Conventions" on page xvi
- "Additional Support" on page xvii

Purpose

The *Hyperion System 9 BI+ Interactive Reporting* \mathbb{T} *User's Guide* provides information that you need to use Interactive Reporting Studio. It explains the user interface a features and options and contains the concepts, processes, procedures, formats, tasks, and examples that you need to use the software.

Audience

This book is for all levels of Interactive Reporting Studio[™] and Web Client[™] users, from those who need to simply retrieve and view data in a report format, to those who need to build queries and reports as well as analyze data.

Document Structure

This document contains the following information:

- Chapter 1, "Introducing Interactive Reporting Studio" introduces Interactive Reporting software tools and provides an overview of how anyone can use Interactive Reporting to access and analyze database information.
- Chapter 2, "Interactive Reporting Studio: A Tutorial" offers a tutorial on basic data analysis techniques and familiarizes you with the powerful capabilities and features of the Interactive Reporting application.
- Chapter 3, "Interactive Reporting Studio Basics" provides an overview of the Interactive Reporting and describes fundamental features and functions.
- Chapter 4, "Querying Relational Databases" explains how to use Interactive Reporting to connect to and query a relational database. It provides basic information about data models and the Hyperion System 9 BI +Interactive Reporting Repository, as well as how to build and process queries and subqueries.
- Chapter 5, "Working with Query Results" explains how to work with the results sets obtained from your relational database query or data import. It includes how to enhance your results set, as well as how to export your results to different file formats.
- Chapter 7, "Querying Multidimensional Databases" explains how to use Interactive Reporting to connect to and query a multidimensional database. It details how to build, refine, and process OLAP queries and how to apply filters.
- Chapter 6, "Working with Tables" explains how to use tables to organize your data.
- Chapter 8, "Analyzing Data with Pivot Tables" explains how to use pivot tables to quickly summarize or cross-tabulate large amounts of data.
- Chapter 9, "Charting Data" explains how to use the Interactive Reporting Studio charting features to perform interactive analysis of your data in a graphic format
- Chapter 10, "Report Designer" explains how to use the Report Designer, Interactive Reporting Studio dynamic analytical report writer, to create free-form, presentation-quality reports.

- Chapter 11, "Using Filters" explains how to use filters to refine your data and restrict the data you do not need for more intelligent analysis.
- Chapter 12, "Working with Computed Items" describes how to use calculations to compute new data items. Such calculations are important for supplementing the information already stored in the database
- Chapter 13, "Applying Sorts" discusses features that enable you to sort data in various Interactive Reporting sections, including sort lines, single sorts, and nested sorts.
- Glossary contains a list of key terms and their definitions.
- Index contains a list of Interactive Reporting terms and their page references.

Where to Find Documentation

All Interactive Reporting documentation is accessible from the following locations:

- The HTML Information Map is available from the Interactive Reporting Help menu for all operating systems; for products installed on Microsoft Windows systems, it is also available from the Start menu.
- Online help is available from within Interactive Reporting after you log on to the product, you can access online help by clicking the Help button or selecting Help from the menu bar.
- The Hyperion Download Center can be accessed from the Hyperion Solutions Web site.
- > To access documentation from the Hyperion Download Center:
- **1** Go to the Hyperion Solutions Web site and navigate to **Services > WorldWide Support > Download Center**.
- **Note:** Your Login ID for the Hyperion Download Center is your e-mail address. The Login ID and Password required for the Hyperion Download Center are different from the Login ID and Password required for Hyperion Support Online through Hyperion.com. If you are not sure whether you have a Hyperion Download Center account, follow the on-screen instructions.
 - 2 In the Login ID and Password text boxes, enter your e-mail address and password.
 - 3 In the **Language** list box, select the appropriate language and click **Login**.
 - 4 If you are a member on multiple Hyperion Solutions Download Center accounts, select the account that you want to use for the current session.
 - **5** To access documentation online, from the Product List, select the appropriate product and follow the on-screen instructions.

Help Menu Commands

Table i describes the commands that are available from the Help menu in Interactive Reporting.

Table i	Help	Menu	Commands
---------	------	------	----------

Command	Description	
Help on This Topic	Launches a help topic specific to the window or Web page.	
Contents	Launches the Interactive Reporting help.	
Information Map	Launches the Interactive Reporting Information Map, which provides the following assistance:	
	Online help in PDF and HTML format	
	Links to related resources to assist you in using Interactive Reporting	
Technical Support	Launches the Hyperion Technical Support site, where you submit defects and contact Technical Support.	
Hyperion Developer's Network	Launches the Hyperion Developer Network site, where you access information about known defects and best practices. This site also provides tools and information to assist you in getting starting using Hyperion products:	
	 Sample models A resource library containing FAOs tips and technical white papers 	
	 Demos and Webcasts demonstrating how Hyperion products are used 	
Hyperion.com	Launches Hyperion's corporate Web site, where you access a variety of information about Hyperion:	
	Office locations	
	• The product suites	
	Consulting and partner programs	
	Customer and education services and technical support	
About Interactive Reporting	Launches the About Interactive Reporting dialog box, which contains copyright and release information, along with version details.	

Conventions

The following table shows the conventions that are used in this document: [

Table ii	Conventions Used in This	s Document

Item	Meaning
>	Arrows indicate the beginning of procedures consisting of sequential steps or one-step procedures.
Brackets []	In examples, brackets indicate that the enclosed elements are optional.
Bold	Bold in procedural steps highlights user interface elements on which the user must perform actions.
CAPITAL LETTERS	Capital letters denote commands and various IDs. (Example: CLEARBLOCK command)

Item	Meaning	
Ctrl+0	Keystroke combinations shown with the plus sign (+) indicate that you should press the first key and hold it while you press the next key. Do not type the plus sign.	
Ctrl+Q, Shift+Q	For consecutive keystroke combinations, a comma indicates that you press the combinations consecutively.	
Example text	Courier font indicates that the example text is code or syntax.	
Courier italics	Courier italic text indicates a variable field in command syntax. Substitute a value in place of the variable shown in Courier italics.	
ARBORPATH	When you see the environment variable <i>ARBORPATH</i> in italics, substitute the value of ARBORPATH from your site.	
п, х	Italic <i>n</i> stands for a variable number; italic <i>x</i> can stand for a variable number or a letter. These variables are sometimes found in formulas.	
Ellipses ()	Ellipsis points indicate that text has been omitted from an example.	
Mouse orientation	This document provides examples and procedures using a right-handed mouse. If you use a left-handed mouse, adjust the procedures accordingly.	
Menu options	Options in menus are shown in the following format. Substitute the appropriate option names in the placeholders, as indicated.	
	Menu name > Menu command > Extended menu command	
	For example: 1. Select File > Desktop > Accounts.	

Table ii Conventions Used in This Document (Continued)

Additional Support

In addition to providing documentation and online help, Hyperion offers the following product information and support. For details on education, consulting, or support options, click the Services link at the Hyperion Solutions Web site.

Education Services

Hyperion offers instructor-led training, custom training, and e-Learning covering all Hyperion applications and technologies. Training is geared to administrators, end users, and information systems professionals.

Consulting Services

Experienced Hyperion consultants and partners implement software solutions tailored to clients' particular reporting, analysis, modeling, and planning requirements. Hyperion also offers specialized consulting packages, technical assessments, and integration solutions.

Technical Support

Hyperion provides enhanced telephone and electronic-based support to clients to resolve product issues quickly and accurately. This support is available for all Hyperion products at no additional cost to clients with current maintenance agreements.

Documentation Feedback

Hyperion strives to provide complete and accurate documentation. Your opinion on the documentation is of value, so please send your comments by going to http://www.hyperion.com/services/support_programs/doc_survey/index.cfm.

1

Introducing Interactive Reporting Studio

This sectionshows how anyone —both technical and nontechnical people—can use Interactive Reporting Studio to access and analyze database information.

This chapter introduces the conceptual background of business intelligence software tools and provides an overview of database concepts. It also explains the Interactive Reporting Studio approach to querying databases.

Interactive Reporting Studio is an all-in-one query, data analysis, and reporting tools for your Windows or UNIX system.

In This Chapter	Overview	. 22
	How Interactive Reporting Studio Works	. 23
	About Interactive Reporting Documents	. 23
	Turning Data into Information	. 27

Overview

Interactive Reporting Studio connects business users to data and give them a complete set of tools to support business decisions including ad hoc *client/server* querying, reporting, and analysis all in one application. Interactive Reporting Studio provides the following capabilities:

- Data extraction and analysis
- Reporting and distribution
- Platform development

The Interactive Reporting Studio is an all-in-one query, data analysis, and reporting tool. The interface is highly intuitive and provides an easy-to-navigate environment for data exploration and decision making. With a consistent design paradigm for query, pivot, charting, and reporting, users at any level move fluidly through cascading dashboards—finding answers fast. Trends and anomalies are automatically highlighted, and robust formatting tools enable users to easily build free-form, presentation-quality reports for broad-scale publishing across their organization.

Interactive Reporting Studio Features

Interactive Reporting Studio features include:

- Support for all industry-standard databases (see "Relational Databases" on page 23 and "Multidimensional Databases" on page 23 for a list of the specific databases supported).
- A point-and-click interface for intuitive custom query and report building.
- Support for Microsoft Windows and Motif (UNIX), with complete file compatibility between the platforms.
- Easy, nonprocedural navigation between query and reporting sections.
- An extensive online help system that provides assistance for features and document construction.
- A drag-and-drop Data Layout tool for developing reports and analyzing data.
- Interactive pivot reporting that lets you perform unrestricted drill-down analysis of different data relationships.
- Extensive formatting tools for creating compelling data presentations.
- An easy-to-use, interactive charting utility for graphically displaying and drilling-down into data.

How Interactive Reporting Studio Works

Interactive Reporting Studio enables you to access and analyze information stored in different company data sources. They connect you to data and supply a complete set of tools that enable you to build queries quickly and intuitively—by clicking icons and manipulating objects. Interactive Reporting Studio automatically builds a query to your specifications, send it to the database, and display the retrieved data as a table of results.

There are many types of data sources, but the most prevalent are relational databases and multidimensional databases.

Relational Databases

A *relational database* is a collection of data items organized as a set of formally described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables.

The definition of a relational database results in a table of *metadata* or formal descriptions of the tables, columns, domains, and constraints. Metadata is literally "data about data."

Multidimensional Databases

A *multidimensional database* is a data cube that provides multidimensional views of business data. Multidimensional databases are OLAP servers that enable you to easily and selectively extract and view data from different points of view. Multidimensional databases consider each data attribute as a separate dimension and allow you to create hierarchies within a dimension.

OLAP (On-Line Analytical Processing) designates a category of applications and technologies that allow the collection, storage, manipulation, and reproduction of multidimensional data, with the goal of analysis. OLAP provides for the fast analysis of multidimensional shared information.

About Interactive Reporting Documents

Interactive Reporting documents (.bqys) are files you create and use to retrieve information from a database, analyze the information, and build reports. Since Interactive Reporting Studio is an integrated query, analysis, and reporting tools, Interactive Reporting Studio documents have multiple sections, each of which governs one part of the query and reporting process. You create sections progressively as you query a database, retrieve results, and then generate reports.

Interactive Reporting documents (.bqy) can contain data from any number of relational databases queries, multidimensional database queries, and/or from imported data. Documents usually include one or more of the following items:

- A data model, which is a visual representation of actual database tables
- A query or multiple queries for retrieving a subset of data from the database

- *Join* options, including local joins between different data sets within a single document, local join filters, and optional join path generation
- A results set displayed in a table-style format
- Reports presenting customized hierarchical views of your data
- Multidimensional *pivot* tables that permit drill-down analysis of data results
- *Charts* that graphically display your query results and allow different angles of vision on the data.

All Interactive Reporting documents (.bqys) usually have at least one Query section and one Results section. From the Results section, you can create multiple Pivot, Chart, Table, and Report sections to analyze and present data. Developers can also create Dashboard sections, which provide an automated push-button interface to a document for use by other users across the enterprise.

Data Source Connections

For Interactive Reporting Studio users, the process of creating a new document and connecting to a database is simple. You select an Interactive Reporting database connection, or .oce file, for the database server you plan to use and enter your database password. You can select either a new or an existing Interactive Reporting database connection.

The way you choose an Interactive Reporting database connection depends on the data model or document with which you are working, and also on which Interactive Reporting Studio edition you are using:

- When a data model is present in the Query workspace, Interactive Reporting Studio automatically prompts you with the correct connection when your actions require a database connection. You need to be connected to a database when you download a data model and when you process your query to retrieve a data set. In addition, you must be connected to show values for a server filter, to use server functions to create a computed item, or to schedule a document.
- When you open an Interactive Reporting Studio to begin a work session (for example, by downloading a data model from an Hyperion System 9 BI + Repository, or creating a new data model), you must select the correct connection for the targeted database.

Interactive Reporting database connections retain all the information necessary to log onto a specific configuration of database and connection API software. In addition, an Interactive Reporting database connection retains DBMS-specific connection preferences as well as specifications for automatic access to metadata. This simplifies the connection process for company personnel by transparently handling host and configuration information. Each user can substitute his or her own database user name when using the Interactive Reporting database connection, which enforces security measures and privileges that are centralized at the database server.

Interactive Reporting database connections have significant advantages in network environments with many database users. One connection can be created for each database connection in the environment and shared with each end-user. Because passwords are not saved with the Interactive Reporting database connection, there is no danger that distribution will provide unauthorized access to any user who receives the wrong Interactive Reporting database connection file or acquires it from other sources.

Data Models

After connecting to a database, Interactive Reporting Studio presents subsets of the database contents in the Query section through custom views called data models, which are visual representations of actual database tables. You use a data model to interact with a database to create queries that specify which data to fetch from the database and retrieve for analysis.

Data models make the database more accessible because you can:

- Substitute descriptive names for arcane database table and column names.
- Create custom views of the data.
- Add computed fields for performing calculations on the retrieved data.

In addition to standard data models derived from database tables, Interactive Reporting Studio lets you create metatopics—virtual views independent of the actual database. You can use metatopics to standardize complex calculations and simplify views of the underlying data with intuitive topics customized for business needs.

Depending on their Interactive Reporting Studio tool, users can create their own data models or use data models provided by other users or through the centralized Hyperion System 9 BI + Repository, a catalogued storehouse of database views for querying.

You also optionally can provide a document that contains a master data model from which your users can then build one or more queries. This master data model allows your users to concentrate on the data they want, not how to set up the data access. Any existing data model can be promoted to a master data model.

Thus, you can offer users a raw look at the table schema, or you can hide the complexity by first creating one or more metatopics and then promoting the data model. Each time the user adds a new query, the Interactive Reporting Studio asks if the query should be linked to the master data model. Any linked queries inherit changes made to the master data model, but the query's data model is locked and cannot be modified. (Only the master data model can be changed.)

Queries

A *query* is a request for information from a database. Queries take the form of a command language that lets you select, insert, update, find out the location of data, and so forth.

The standard command language for getting information from and updating a relational database is Structured Query Language (SQL). SQL statements are used both for interactive queries for information from a relational database and for gathering data for reports.

Multidimensional databases also require a language that allows you to express multidimensional queries; however, to date, there is no standard. MDX (Multidimensional Expression Language) is used by Microsoft's OLE DB for OLAP API and OLAP Services. Hyperion System 9 BI + Analytic Services[™] uses MaxL (Multidimensional Access Language). MDSQL (Multidimensional Query Language) is yet another query language.

With Interactive Reporting Studio, you do not need to know SQL or any multidimensional query languages to create powerful database queries. You build queries by choosing the data you want to retrieve from a visual representation of the database.

Interactive Reporting Studio offers two query methods, each of which is displayed as a separate section within an Interactive Reporting Studio document:

- Query Displays the structure of the relational database as tables (or topics), which are used to create a data model—the visual representation of the database tables. An Interactive Reporting Studio document can have more than one data model.
- OLAPQuery Displays the structure of the multidimensional database as a hierarchical tree. OLAP queries are displayed in a form similar to a pivot table, except the data comes straight from the OLAP server.

A Interactive Reporting Studio document can contain one or more relational Query sections as well as one or more OLAPQuery sections. This allows users access to information in organizations that have both types of databases.

Analysis and Reporting

Once a query is processed and data results are returned to the desktop, you can use the Interactive Reporting Studio's powerful reporting and analysis tools to create custom views, cross-sections, and drill-downs to slice and dice data and view the multidimensional relationships it contains.

You may create as many different views of the data as you want, and you can display the information in any form and from any angle possible. At any time, you can reconnect to the server and update your reports and charts with fresh data from the database.

You can also use Interactive Reporting Studio to work autonomously with data after disconnecting from the server. Even without a database connection, you can continue to analyze data and produce reports. You can save results in the desired format for additional refinement in Interactive Reporting Studio, or you can export the data to other applications for further analysis.

Interactive Reporting Studio enable you to create a wide variety of reports, including:

• Tables – Columnar arrangements of data. Tables are used as building blocks in other reporting sections. You can apply filters to tables, add computed items, include subtotals and grand totals, as well as summary totals such as sum, count or average.

- Pivot tables Interactive tables that quickly summarize or cross-tabulate large amounts of data. You can rotate rows and columns to see different summaries of data or display the details for areas of interest. A pivot table summarized data by using a summary function that you specify, such as Sum, Count, or Average. You can include subtotals and grand totals automatically, or use your own formulas by adding computed items.
- Charts A visual display of information; fully interactive, three-dimensional views of data. Interactive Reporting Studio displays data from results sets as bars, lines, columns, pie slices, or other shapes in the chart. When you create a chart, the values from the worksheet are automatically represented in the chart. Charts are linked to the data they are created from and are updated when you change the data.
- Custom reports Using an Interactive Reporting Studio Report Designer, you can create free-form presentation-quality reports with graphic objects, predefined fields, band-style report data from multiple data sources and computed fields, charts, and pivots; Smart reports enable you to embed charts and pivot tables and show only the data that is relevant to the section in which they are placed.

Turning Data into Information

Data is meaningless unless it can be analyzed and interpreted. Analysis depends on consolidating and summarizing data through mathematical operations that reveal meaningful relationships, also called aggregation. The result is a summary of the data at a higher level, which summarizes and consolidates data from a lower level.

Aggregation is a critical feature of data analysis. Successful and rapid interpretation of data requires you to have some easy method of aggregating data and representing it for easy interpretation.

The Interactive Reporting Studio aggregation techniques are easy to master. With the simple drag-and-drop of a data item, you can reorganize your data. Remove an item or drill down into your data, and you disaggregate your data.

Interactive Reporting Studio provides a great deal of flexibility in how you choose to aggregate your data. One possibility is to aggregate your data at the time of your query, called server aggregation or preaggregation. In this case, the database server actually performs the aggregation for you.

An advantage of server aggregation is that since the database server returns less data, preaggregation reduces network traffic and takes less time. However, if you preaggregate data at the server, you might find that later when you want to drill down into the data that you cannot reach the data depths that you need. This is because you excluded the more detailed data by preaggregating. If you do not know your database tables, you could eliminate data that might be important for analysis. Also, preaggregation requires more server processing resources.

Alternatively, you can aggregate data on your desktop without involving the server. Interactive Reporting Studio automatically aggregates your data for you in report sections.

2

Interactive Reporting Studio: A Tutorial

This chapter offers a tutorial showing basic data analysis techniques. It also familiarizes you with the powerful capabilities and features of the Interactive Reporting Studio.

To use this tutorial, you must have access to Interactive Reporting Studio and their sample database and documents. A connection to your company database is *not* necessary.

In This Chapter	Starting the Interactive Reporting Studio	30
	Opening and Saving a Sample File	30
	Looking at a Simple Query	31
	Viewing Results	32
	Pivoting Data	34
	Charting Data	36
	Designing Reports	38

Starting the Interactive Reporting Studio

To start a Interactive Reporting Studio, start Start > Programs > Interactive Reporting Name. For example Start > Programs > Designer.

Whenever you start a Interactive Reporting Studio, you must create a new Interactive Reporting document or open an existing Interactive Reporting document.

Opening and Saving a Sample File

During the installation process, Interactive Reporting Studio install several sample files. This tutorial uses the document named Sample1.bqy to familiarize you with the many Interactive Reporting Studio features.

- To open Sample1.bqy:
- **1** In the Welcome dialog box, select **Recent Documents** from the Open Existing Documents area.
- 2 Click **Browse** and navigate to the Samples folder in the appropriate Interactive Reporting Studio folder (for example, the Designer folder or the Explorer folder).
- Note: If you cannot locate the sample files, please contact your Interactive Reporting Studio administrator.

3 Select **Sample1.bqy** and click **Open**.

The Interactive Reporting Studio workspace is displayed and shows the sample document with the Dashboard active.

4 Select File > Save Options > Save Query Results with Document.

The Save Query Results With Document dialog box opens.

Save Query Results With Document				
Query Results	Computed Columns	OK		
SalesDuery	RankedSales	Cancel		
	T	Help		

- 5 Make sure that the all of the items in the Query, Results and the Computed Columns lists are selected, and then click **OK**.
- 6 Select File > Save As to open the Save File dialog box.
- 7 Type a new name for the sample document (for example, *practice.bqy*) and click **Save**.

Looking at a Simple Query

The Query section is the foundation of an Interactive Reporting Studio document. It is the space where you build questions for the database.

The Query section in the sample document is labelled *SalesQuery*. It is a simple query constructed for a fictitious company named Books, Movies, and Video (BMV). BMV distributes books, movies, and videos to a number of retail stores. To predict trends and locate strengths and weaknesses in its distribution techniques, BMV warehouses extensive data.

> To view the SalesQuery section, select **SalesQuery** in the Sections pane.

Viewing Database Tables

In Sample1.bqy, there are four tables, also called *topics*, in the Content pane. These four topics represent tables in the BMV database. The database tables are listed in the Catalog pane, located to the left of the Content pane.

To view all the database tables:

1 In the Catalog pane, click the plus sign to the left of Tables.

If you are not connected to the sample database, the Hyperion Sample1.oce dialog box is displayed and prompts you for a user name and password.

An Interactive Reporting database connection file (.oce) is a file that enables you to connect to a database. The following icons on the Status bar indicate your connection status:



Connected – You must be connected to a database to work in the Query section.

∰₩₩

Disconnected – You do not need to be connected to a database for many Interactive Reporting Studio tasks.

For the sample database, you do not need to enter any information in this dialog box.

2 Click **OK** to connect to the database.

The Tables tree in the Catalog pane expands to show all of the tables in the database.

Adding Topics to a Query

To include data from a particular table in your query, drag and drop the table from the Catalog pane to the Content pane.

Each topic contains a list of topic items that represent fields or rows of data in the database. In Sample1.bqy, the topics included in the query are *Periods*, *Sales Fact*, *Products*, and *Stores*.

You build queries by adding topics from the Content pane to the Request line. You can drag and drop any topic item to the Request line. When you process a query, Interactive Reporting Studio return data for all the topic items present on the Request line. In the sample, several topic items from each table have already been dragged to the Request line (for example, Unit Sales, Amount Sales, Year, Quarter, and so on).

You can also add *filters* to the data, or specify columns by which to *sort* the data. You can apply filters and sorts in either the Query section or the Results section.

In the Query section, filters instruct the *database server* to filter unwanted information from the requested data. Sorts instruct the database server to retrieve data to your desktop in a particular order.

Viewing Results

Data returned from a query is displayed in the Results section. Each column of results corresponds to items on the Request Line in the Query section. Request items are listed in the Catalog pane.

> To view the Results section, select **SalesResults**. in the Sections pane:

Use the arrow buttons on the Section title bar to compare the items in the Query and Results sections.

Reordering Columns

To reorder columns in the Results section, drag one column to the left or right of another column.

You can also move column labels in the Data Layout.

Sorting Columns

- To sort columns of data:
- 1 If the Sort line does not show, click **Sort(0)** on the Section title bar.
- 2 Drag **Product Line** from the Catalog pane to the Sort line.
- **3** Click **Sort Now** to group items by Product Line.

Filtering Data

At times you may have more data in a column than you want. Use the Filter line to filter the data displayed in a column.

- ► To filter data:
- 1 If the filter line does not show, click filters(0) on the Section title bar.
- 2 Drag Amount Sales from the Catalog pane to the filter line. [Ctrl+L]

The Filter dialog box is displayed.

- 3 Select >=Greater or Equal from the filter drop-down list.
- 4 Click **Custom Values**, type **100000** in the field provided, and click **OK**.

All entries with sales amounts less than \$100,000 dollars are dropped from the Results section.

To remove a data filter, delete **Amount Sales** from the Filter line.

The data is redisplayed when you remove the filter.

Calculating Data

Interactive Reporting Studio can perform calculations on columns of numeric data.

- To sum up numeric data:
- 1 Select the Amount Sales column.

The Insert Grand Total dialog displays.

- 2 Select Results > Grand Total and click OK.
- 3 Scroll down to the bottom of the table to view the total amount of sales.

BMV total sales for 1999-2000 are \$132,881.

If the figure you see does not match \$132,881, you need to remove any filters imposed on the results. Delete any filters from the Filter line and view the total again. The sum automatically adjusts.

If you see ########, the number is too large to fit in the designated space.

To resize the column, select Format > Column > Auto-Size Width.

Pivoting Data

In Sample1.bqy, the SalesPivot section is a simple example of a pivot table. Pivot tables provide multiple angles on your data.

> To view the Pivot section, select **SalesPivot** in the Section pane:

The SalesPivot section is displayed.

Pivot tables allow you to quickly summarize data in the Results section and immediately see the relationships between different dimensions of your data. These reports *pivot* to provide fresh angles of vision on your data.

- > To create a pivot table:
- **1** Select **SalesQuery** in the Section pane.
- 2 Select Insert > New Pivot.
- 3 If the Data Layout is not already visible, click **Data Layout** on the Section title bar.

Drag one or more items from the Catalog pane into each of the Data Layout panes. Remove items from the Data Layout panes and add new ones. Use the Top Labels pane and Side Labels pane for text. Use the Facts pane for numeric values.

- ► To pivot views:
- 1 Click the *dimension tab* at the end of the row labels.
- 2 Drag the tab down and left to turn your row into a column.

The same data is displayed but with a different angle on the data.

3 Take the tab of the newly formed column and drag it so it becomes a row again.

Calculating Totals and Subtotals

To make effective use of data, you may need to generate totals or subtotals.

					A	
8			2001	2002		
			Unit Sales	Unit Sales	Г III	
Г	Americas	Books	1336072	227384		
Г		Music	613754	92642	1	
Γ		Videos	286195	37187		
		Total	2236021	357213		
	Asia Pacific	Books	788729	163952		
		Music	435276	81131		
		Videos	189036	40731		Subtotals
		Total	1413041	285814		
	Europe	Books	903138	142380		
		Music	502274	65673		
		Videos	217638	33453		
		Total	1623050	241506		
	Total		5272112	884533		Total
						Dimonsion
	Side Labels: I	Region + Produ	ict Line	▲ Top	o Labels: Year	Handles

- To calculate totals and subtotals (as displayed above):
- 1 Drag **Product Line** and **Region** to the Row pane in Data Layout.
- 2 Drag Year to the Column pane in Data Layout.
- **3** Drag **AmountSales** to the Facts pane in Data Layout.
- 4 Select the dimension handle for Region (click at the bottom of the Region column).
- 5 Select **Pivot > Add Totals**.

A row is added that shows the total number of product line sales for all regions.

6 Select the handle for Product Line and select **Pivot > Add Totals**.

A row is added that shows the subtotals (also know as break totals) for each product line by region.

Drilling Down

More data is available for analysis than is currently visible in your pivot table.

- To drill anywhere, select the Region column and select Pivot > Drill Anywhere > Country.
 A column is added to your pivot table that shows countries within region.
- To restore the original pivot table without the Unit Sales column, select the Country column and select Pivot > Drillup.

Hiding Data

You can temporarily hide data.

- > To hide an item, select a label such as the Americas label and select **Pivot > Hide Items**.
- > To focus on an item, select a label such as the Americas label and select **Pivot > Focus On Items**.
- > To restore your excluded items, select **Pivot > Show All Items**.

Adding Color

Use the Format toolbar to add color to emphasize aspects of your pivot table.

Note: If the Format toolbar is not visible, select View > Toolbars > Formatting.

- To add a line color:
- 1 Click the label, dimension handle, or column whose line color you want to change.
- 2 On the Format toolbar, open the Line Color list and select a color from the palette.
- ▶ To add a fill color:
- 1 Click the label, dimension handle, or column whose fill color you want to change.
- 2 On the Format toolbar, open the **Fill Color** list and select a color from the palette.
- ▶ To add a text color:
- 1 Click the label, dimension handle, or column whose text color you want to change.
- 2 On the Format toolbar, open the **Text Color** list and select a color from the palette.

Charting Data

Interactive Reporting Studio charting features make graphic analysis of data and powerful presentations simple. In Sample1.bqy, the *AllChart* section is a chart based on the data from the original query.

To view the Chart section, select Pivot > Show All Items. The AllChart section is displayed.

Interactive Reporting Studio rapidly converts data from one chart type to another.

To change the chart type and format, select a chart format from the Chart list. As you shift from one chart type to another, data may be shifted to different *axes*.

- To create a new chart:
- **1** In the Section pane, select **SalesQuery**.
- 2 Select Insert > New Chart.

Chart is displayed in the Section pane.

3 In the Section pane, double-click Chart.

The Section Label dialog box is displayed.

- 4 Delete Chart and type Unit Sales Region, and then click OK.
- 5 If the Data Layout is not already visible, click **Data Layout** on the Section title bar.

Numeric values (facts) are placed in the y pane in the Data Layout. Non-numeric data (dimensions) are placed in the x pane and z pane in the Data Layout.

- 6 Drag **Unit Sales** to the x pane in Data Layout.
- 7 Drag **Region** to the z pane in Data Layout.
- 8 Drag **Product Line** to the y pane in Data Layout.
- 9 Click a Legend box to change the distribution and patterns of colors.

Sorting Charts

It is often useful to order the bars of a chart sequentially.

- ► To sort your chart:
- 1 If the Sort line is not visible, click **Sort** on the Section title bar.

Sort provides drop-down menus to select sort criteria. Experiment sorting.

2 Click the Ascending or Descending sort icon on the Standard toolbar.

The data on the chart is rearranged in ascending or descending order.

Inserting Text Labels

You can insert a text label anywhere within the chart to further explain or emphasize a chart component.

- To insert a text label:
- **1** On the shortcut menu, click **Insert Text**.

The Set Inserted Text dialog box is displayed.

- 2 Type the text you want to insert and click **OK**.
- **3** Drag the text box to any position on the chart.

Designing Reports

The Report Designer is another way to analyze and present data and offers a great deal of formatting flexibility. You can embed either a pivot table or a chart directly in a report.

► To view a Report section, in the Section pane, select **RegionReport**.

The RegionReport section is displayed.

- To create a new tabular report:
- 1 Select Insert > New Report.

Blank columns are displayed in the Content pane.

2 Click **Groups** and **Table** on the Section title bar to see all panes in the Data Layout.

In the Report section, create reports by dragging items from the Catalog pane to the Groups Data Layout and the Table Data Layout.

The Groups Data Layout allows you to drag nonquantifiable items and create separate tables for each label in a report group. In other words, when you designate a results column to serve as a report group, you instruct Interactive Reporting Studio to organize data in repeating collections of records according to the Report group.

The Table Data Layout is divided into the Dimensions and Facts panes. The Facts pane allows you to drag quantifiable items to show quantity. Totals are generated automatically in the report body. The Dimensions pane allows you to drag nonquantifiable items to itemize the facts.

- 3 In the Catalog pane, double-click the **SalesQuery** folder, and then click the **plus sign** to the left of the SalesResults folder.
- 4 Drag Year and Quarter into the Dimensions pane in Table Data Layout.
- 5 Drag Amount Sales into the Facts pane in Table Data Layout.
- 6 Drag Territory to the Report Group 1 field of the Groups Data Layout.
- 7 Drag Country to the Report Group 2 field of the Groups Data Layout.

The new report has visible section boundaries. Results columns added to the Facts pane are automatically summed and the totals are displayed in the tabular report of the report table column.

Note: You can disable automatic totaling.
Creating Smart Reports

Smart reports allow you to embed charts and pivot tables into a report body. These reports show only the data that is relevant to the report section in which they are placed. For example, if the report is grouped by year and you insert a chart in the report body, the chart replicates automatically so that there is a chart for each year of data in the report. Each year's chart contains data specific to that year.

- To create a Smart report:
- 1 In the Catalog pane, click the **plus sign** to the left of the SalesQuery folder.
- 2 Drag a Chart icon (use either Unit Sales Region or AllChart) and position it just above the table in the report body.

Allow some white space between the embedded chart and the table.

In the newly-created Smart report, the chart changes to reflect the data specific to each country within a territory.

Sorting Columns

▶ To sort table columns, select a table column, anddrag it to the Sort line.

Multiple table columns can be added to the Sort line to create a nested sort.

Setting Up a Report Page

- To set up a report page:
- **1** Select **Report > Report Setup**.

The Report Page Setup dialog box is displayed.

- 2 On the Margin page, set margin sizes (top, bottom, left, and right) and click OK.
- **3** On the Column page, specify the **number of columns** on a page and the default column width and spacing, then click **OK**.

Page breaks can be inserted before and after a report body or before and after a Report Group label.

- To insert a page break, select a Report Group header (be careful not to select the Report Group label) and on the shortcut menu, select Page Break After or Page Break Before.
- ► To remove a page break, select a Report Group header with a page break applied, and on the shortcut menu, select **Page Break After** or **Page Break Before** to remove the check mark.

3

Interactive Reporting Studio Basics

This chapter provides an overview of the Interactive Reporting Studio workspace and describes fundamental features and functions. These features, available through the File, Edit, View, Insert, and Tools menus, help you manage documents and provide numerous ways to manipulate data in the query and results sections.

In This Chapter	Using the Interactive Reporting Workspace	. 42
	Managing Documents	. 45
	Changing Workspace Views	. 76
	Inserting Sections and Breaks	. 77
	Formatting Numeric Data Types	. 79
	Setting Options	. 86

Using the Interactive Reporting Workspace

When you start and create a new document or open an existing document, the workspace is displayed.

The workspace is a tri-pane window that provides the tools and the access you need to perform database operations. You can resize the window and customize the appearance of most of the elements in the workspace, and you can decide whether certain elements should be visible.

Toolbars, buttons, Data Layouts, and palettes provide quick access to most functions. All the toolbars, Data Layouts, and palettes are dockable; that is, they can be detached and dragged as floating windows to any location on the workspace. You can also hide these objects from view to reclaim screen space for report viewing.

Main Menu

The Main menu provides access to all menus and functions. In addition, certain menus change based on the active section. Every section has a unique menu whose label matches the name of the section. Interactive Reporting Studio also provide extensive shortcut menus.

Toolbars

Toolbars provide quick access to frequently used functions and features. Available toolbars include:

- Standard Toolbar
- Navigation Toolbar
- Formatting Toolbar
- Section Toolbar

Standard Toolbar

The Standard toolbar contains icons for commonly used operations.

Navigation Toolbar

The Navigation toolbar returns to the dashboard from another section when the Section catalog, Section title bar, toolbars, and menus are turned off. This toolbar is hidden by default. When activated, it includes the Back and Forward buttons and a dashboard Home button.

Formatting Toolbar

The Formatting toolbar provides text formatting, styling, and editing commands.

Section Toolbar

The Section toolbar provides commands for a specific document section.

Section Title Bar

The Section title bar displays the name of the active section (that is, the section in which you are currently working). The Section title bar includes navigation buttons, as well as buttons that correspond to the command lines and Data Layouts available in a particular section.

Command Lines

You use command lines to complete important operations during the query and reporting processes. The command lines that ar are determined by the active section. All command lines can be docked, resized or hidden. The command lines available in each section are:

- Query Section Request, Filter and Sort
- Results Section Filter and Sort
- **Report Section** Sort and Expression

Data Layout

Data Layouts are drag-and-drop templates used in the Pivot, Chart, Results, OLAPQuery, and Report sections. Each Data Layout pane corresponds to a specific layout element of the report. Data Layout allow you to easily view, plot, and manipulate the data in the Content pane.

> To display the Data Layout for a section, click **Data Layout** on the Section Titlebar.

Section Pane

The Section pane lists the sections available in the current document. Possible sections include:

- Dashboard A document front-end that can be customized, which makes it easy for developers to build and deploy analytic applications and for end users to access information.
- **Report** A dynamic and analytical designer that provides the ability to easily develop a complete range of reports. Use the Report Designer's visual layout capabilities to drag and drop columns, expression, charts, logos, bitmaps, and other items to quickly design and customize your reports.
- Query The foundation of the document, the Query section enables you to connect to a relational database and download a prebuilt query or data model, or build a new data model and create your own query.
- **Results** Created when you process a query or import data. retrieves data to your desktop and displays it in the Results sections as columns in a table.

- **Pivot** An interactive table that quickly summarizes, or cross-tabulates, large amounts of data. You can rotate its rows and columns to see different summaries of the source data, or display the details for areas of interest.
- Chart A fully interactive, two- or three-dimensional view of your data that provides powerful ways to visually analyze your data.
- Table A single-dimension report that displays your data in columns. Tables are often used as building blocks in other sections.
- OLAPQuery A query section specifically designed for connecting to multidimensional databases.
- OLAPResults Results section for an OLAPQuery.

Catalog Pane

The Catalog pane contains the objects you use to build contents.

For example in the Pivot section, the Catalog pane contains Results columns. In the Query section, the Catalog pane contains the tables available in your database. In the Report Designer section, the catalog pane contains the Results, Pivot, and Chart sections in addition to available graphic and field elements

To use an object, drag it from the Catalog pane to the Content pane or to the Data Layout.

Content Pane

The Content pane displays a data model or provides a view of the content from a Query, Pivot, Chart, or Dashboard section.

Status Bar

The Status bar, located below the Content pane, displays information about the current database connection, document, and document section.



Connected – You must be connected to a database to work in the Query section.



Disconnected – You do not need to be connected to a database for many Interactive Reporting Studio tasks.

Managing Documents

Managing documents involves:

- Maintaining Documents
- Saving Documents
- Specifying Save Options
- Exporting Data
- Printing Documents

Maintaining Documents

Review the following sections for information on:

- Opening an Existing Document
- Closing a Document

Opening an Existing Document

- > To open an existing Interactive Reporting Studio document:
- Select .File > Open, or click the Open Document icon on the Standard Toolbar. The Open File dialog box is displayed.
- 2 Select the file that you want to open and click **OK**.

Closing a Document

Note: Document shutdown scripts are only executed when a document is closed.

- To close a document, select **File > Close**.
- **1** Click Close on the browser.

You are prompted to save changes to your document.

Saving Documents

When you save a document, you save the current formatting and layout of all sections in the document. You can also save the Results section data for work offline.

Note: The Save and Save As commands in Interactive Reporting Studio, Interactive Reporting Web Client, and Workspace do not execute the document shutdown scripts when saving the document. These scripts only execute when a document is closed.

- > To save a document, click the **Save** button on the standard toolbar, or select **File > Save**.
- To save the document under a different name:
- 1 Select File > Save As.

The Save File dialog box is displayed.

2 Type a name for the document in the File Name field and click Save.

Specifying Save Options

► To select specific save options, select **File > Save Options** and choose the desired option.

Save options include:

- Save Query Results With Document
- Compress Document
- Password Protect Document

Save Query Results With Document

This option saves the results of a query and any computed columns (from the results) with a document and allows you to analyze and generate reports using the results set without being connected to the database.

Compress Document

The Compress Document option condenses the number of bytes in a document and saves the document in a compressed file format. It enables you to quickly transmit the document and saves valuable storage space. Compression concentrates the number of bytes by removing empty data fields and unnecessary information. automatically decompresses the file when you reopen it.

Tip: To configure so that it compresses all documents when saving, use the Compress All Documents option. See "General Options" on page 88 for more information.

Password Protect Document

Use the Password Protect Document option to restrict access to a document and ensure the confidentiality of sensitive documents. This option requires users to input a correct password for document access.

- To assign a password to a document:
- 1 Select File > Save Options > Password Protect Document.

The Password Protect Document dialog box is displayed.

Password Protect Document			
Password to open Password:	OK		
Verify Password:	Cancel		

2 In the **Password** field, type the password you want to assign to this document.

Passwords can contain up to 38 alphanumeric characters and are case sensitive.

3 In the Verify Password field, retype the password and click OK.

If the password and the verification do not match, an error message is displayed.

Caution! Keep a list of each document and the specific password needed to access it in a secure place.

Password Protect Designer Mode

Use the Password Protect Designer Mode option to do the following:

• Restrict access to a document script and the scripting environment.

This ensures the integrity and confidentiality of sensitive scripts in a Dashboard section and when you are working with document scripts.

• Enable encryption for startup, shutdown, document, and Dashboard scripts associated with Interactive Reporting documents.

You must supply a valid to encrypt scripts. (Encrypted scripts are not functional in releases prior to 6.6.3 or 8.1.) You can still password protect a document without encrypting any scripts associated with it.

You can define a default encyption state on the General Options tab of the Designer Options dialog box. When you create a new BQY, the default encryption state will match the one on the Designer Options dialog box. Despite the default setting, scripts are only encrypted if the BQY has a valid Design mode password.

- To password protect a script:
- **1** Select File > Save Options > Password Protect Designer Mode.

The Password Protect Designer Mode dialog box is displayed.

2 In the **Password** field, enter the password required to open the document script.

Passwords can contain up to 38 alphanumeric characters and are case sensitive.

3 In the **Verify Password** field, retype the password and click **OK**.

If the password and the verification do not match, an error message is displayed.

- > To open a password protected script:
- **1** Do one of the following:
 - Select File > Document Scripts.
 - Click the Design/Run icon in the Dashboard section run mode.

The Unlock Design Mode dialog box is displayed.

Unlock Design Mode	
Enter password to unlock design m	ode:
	OK
	Cancel

- 2 Enter the password required to open the script and click **OK**.
- > To encrypt a script:
- 1 In the **Password** field, enter the design mode password required to open the script.

Passwords can contain up to 38 alphanumeric characters and are case sensitive.

2 In the Verify Password field, retype the password and click OK.

If the password and the verification do not match, an error message is displayed.

3 Select the Encrypt Dashboard scripts for new document.

Working with Data Sources

You can use a prebuilt data model or you can import files from other applications and use the data as the source for your document. As you work with data sources, you can use the following commands:Use the following commands to import data into the Web client:

- Import Data File
- Import SQL

Import Data File

Use the Import Data File command to use data stored in Microsoft Excel, tab-delimited, or comma-delimited file formats. The data is displayed in the Results section. You can then use the imported data to build reports and perform data analysis.

- > To import a data file:
- **1** Select File > Import Data File > Data File.

The Import File dialog box is displayed.

2 Navigate to the location of the file you want to import.

Select a file type from the Files of Type drop-down list to make the file easier to find. Select from:

- Tab delimited (*.txt)
- Comma Delimited (*.csv)

- Excel (*.xls) (the Import feature supports style and font information and Unicode strings [Excel 2000, Excel 2003 file formats]).
- All files (*.*)
- **Note:** If an Excel (.xls) file to be imported is not in table format, you receive the following error message: "Warning: Data you are trying to import is not in a tabular format. Do you still want to import it?" At this point, you can either reformat the data in the Excel file to a table format, or continue the import. If you continue the format, the data does not map in the BQY.

3 Select the file you want to import and click OK.

The data from the imported file is displayed as a table in the Results section.

Import SQL

The Import SQL command allows you to take a complete SQL statement from a text file, import it into an existing query, and retrieve the data set from the database server. Use this feature to take advantage of SQL statements you have already written.

Before importing SQL files, make sure that the following conditions are true:

- The SQL file you want to import begins with a SELECT statement.
- The Query section of your document is active.
- The connection to the database is active.
- The Content pane does not contain any tables.

You also need to know the number of columns to display in the Results section.

After you import the SQL file into the Query section:

- You cannot edit it.
- You cannot drag items from the table to the Request line.
- You cannot use the custom SQL feature.
- You cannot display its properties.

However, you can specify a user-friendly name for the Request item and identify its data type.

- ► To import SQL files:
- 1 Select File > Import Data File > SQL.

The Import File dialog box is displayed.

- 2 Navigate to the location of the file you want to import.
- **3** Select the file you want to import and click **OK**.

prompts you for the number of data columns. The number that is by default in the dialog box is an estimate.

4 Type the number of columns and click **OK**.

inserts the SQL statement directly into the content, nested between the header and footer "Imported SQL Statement." If the statement is larger than the visible Content pane, use the scroll keys to view it.

Exporting Data

allows you to export data to other file formats for use with applications. Review the following sections for information on exporting data:

- Defining Export Properties for Dynamic HTML in the Results and Table Sections
- Defining Export Properties for Dynamic HTML in the Pivot and OLAP Section
- Defining Export Properties for Dynamic HTML in the Chart Section
- Defining Export Properties for Dynamic HTML in the Report Section
- Defining Export Properties for Static HTML in the Results and Table Sections
- Defining Export Properties for Static HTML in the Pivot and OLAP Sections
- Defining Export Properties for Static HTML in the Report Section
- Exporting a Section
- Exporting to Microsoft Office HTML File Formats
- Exporting a Document as a Web Page
- Browser and HTML Restrictions and Limitations
- Using the Export to HTML Wizard
- Exporting SQL
- Exporting a Query Log

Defining Export Properties for Dynamic HTML in the Results and Table Sections

Before exporting data to an HTML file to use in the Hyperion[®] System[™] 9 BI+[™] Workspace use the Export Properties dialog box to specify export properties for data.

The Export Properties dialog box is separated into two discrete sections for exporting either to dynamic or static HTML. Common export features are shown at the top of the dialog.

The options in this section allow you to override any default pregeneration settings defined for the document. Before any new setting takes effect, you must publish (or republish) the BQY file. Any dynamic HTML changes you make either through the Export Properties dialog box or through the Object Model are not inherited until you republish the BQY file.

Static HTML by definition has pregenerated data and is stored locally.

- > To specify export properties:
- 1 Select Format > Export Properties.

The Export Properties dialog box is displayed.

2 Specify the number of vertical rows to include on an HTML page before starting a new page (file) by checking the **Vertical Page Break** checkbox and entering the number of rows.

The default setting is 100 rows per HTML page. (Select Pixels to specify the number of pixels instead of rows).

To export to one file or to have no page break, leave the Vertical Page Break checkbox blank.

3 In the Dynamic HTML section, select the desired dynamic export properties.

Specify to pregenerate the HTML for All Pages. Enabling pregeneration for all pages can place high demands on the Hyperion System 9 BI + Repository, especially if the section is large. (Select **First and Last _ pages** to specify a range of pages to pregenerate).

- 4 For data exported to tab-delimited text files, select the **Export Without Quotes** check box to exclude double quotation marks around real column/cell values in the exported files.
- 5 Click OK.

Defining Export Properties for Dynamic HTML in the Pivot and OLAP Section

Before exporting data to an HTML file to use in the Hyperion System 9 BI + Workspace, use the Export Properties dialog box to specify export properties for data.

The Export Properties dialog box is separated into two discrete sections for exporting either to dynamic or static HTML. Common export features are shown at the top of the dialog.

The options in this section allow you to override any default pregeneration settings. Before any new setting takes effect, you must publish (or republish) the BQY file. Any dynamic HTML changes you make either through the Export Properties dialog box or through the Object Model are not inherited until you republish the BQY file.

Static HTML by definition has pregenerated data and is stored locally.

- > To specify export properties:
- **1** Select Format > Export Properties.

The Export Properties dialog box is displayed.

2 Specify the number of vertical rows to include on an HTML page before starting a new page (file) by checking the **Vertical Page Break** checkbox and entering the number of rows.

The default setting is 100 rows per HTML page. (Select Pixels to specify the number of pixels instead of rows).

To export to one file or to have no page break, leave the Vertical Page Break checkbox blank.

3 Specify the number of horizontal columns to include on an HTML page before starting a new page (file) by checking the **Horizontal Page Break** checkbox and entering the number of rows.

The default setting is 100 columns per HTML page. (Select Pixels to specify the number of pixels instead of columns).

To export to one file or to have no page break, leave the Horizontal Page Break checkbox blank.

4 In the Dynamic HTML section, select the desired dynamic export properties.

Specify to pregenerate the HTML for All Pages. Enabling pregeneration for all pages can place high demands on the Repository, especially if the section is large. (Select **First and Last _ pages** to specify a range of pages to pregenerate).

- 5 For data exported to tab-delimited text files, select the **Export Without Quotes** check box to exclude double guotation marks around real column/cell values in the exported files.
- 6 Click OK.

Defining Export Properties for Dynamic HTML in the Chart Section

Before exporting a chart section to an HTML file to use in the Hyperion System 9 BI + Workspace, use the Export Properties dialog box to specify export properties for data. Note that any dynamic HTML changes you make either through the Export Properties dialog box or through the Object Model are not inherited until you republish the BQY file.

> To specify export properties:

1 Select Format > Export Properties.

The Export Properties dialog box is displayed.

2 To match the maximum number of bars displayed (X and Z directions) for HTML renderings (static or Hyperion System 9 BI + Workspace) with the corresponding values on the Label Axis tab of the Chart Properties dialog (that is, use the Designer settings for the chart), check the Sync with Chart Properties field.

To manually set the maximum number of bars to display per view, uncheck the Sync with Chart Properties field. When the X-Axis Maximum Bars Displayed and Z-Axis Maximum Bars displayed are active, you can set the maximum number of bars to display for both the X axis and the Z axis. The horizontal scrollbar controls scrolling of the X axis, and the Vertical scrollbar controls scrolling of the Z axis.

3 Select the desired dynamic export properties and click OK.

Specify whether to pregenerate the HTML for All Pages. Enabling pregeneration for all pages can place high demands on the Repository, especially if the section is large. (Select **First and Last _ pages** to specify a range of pages to pregenerate).

Specify whether to pregenerate the HTML for All Views. Enabling pregeneration for all views can place high demands on the Repository, especially if the section is large. (Select **First and Last _ views** to specify a range of views to pregenerate).

Specify whether to show Chart Boundaries. The chart boundary consists of the rectangular parameters, measured in pixels, that surround the actual chart, legend and label(s).

Specify the height of the vertical boundary and the width of the horizontal boundary in pixels.

Defining Export Properties for Dynamic HTML in the Report Section

Before exporting a chart section to an HTML file to use in the Hyperion System 9 BI + Workspace, use the Export Properties dialog box to specify export properties for data. Note that any dynamic HTML changes you make either through the Export Properties dialog box or through the Object Model are not inherited until you republish the BQY file.

- To specify export properties:
- **1** Select Format > Export Properties.

The Export Properties dialog box displayed.

2 Select the desired dynamic export properties and click **OK**.

Specify to pregenerate the HTML for All Pages. Enabling pregeneration for all pages can place high demands on the Hyperion Repository, especially if the section is large. (Select **First and Last _ pages** to specify a range of pages to pregenerate).

Defining Export Properties for Static HTML in the Results and Table Sections

Before exporting data to static HTML, use the Export Properties dialog box to specify export properties for data in the Results and Table sections.

- > To specify export properties:
- **1** Select Format > Export Properties.

The Export Properties dialog box is displayed.

2 Specify the number of vertical rows to include on an HTML page before starting a new page (file) by checking the **Vertical Page Break** checkbox and entering the number of rows.

The default setting is 100 rows per HTML page. (Select Pixels to specify the number of pixels instead of rows).

To export to one file, or to have no page break, leave the Vertical Page Break checkbox blank.

3 Select the desired export properties.

Check the Export with Style Sheet (CSS) option to create a Cascading Style Sheet file separate from the HTML file. If this field is unchecked, the style sheet information is embedded in the HTML file itself.

4 Select the Export MS Excel formulas check box to export Interactive Reporting computed items to Excel. To export a raw table (without formulas), leave the check box blank.

By default the Export MS Excel formulas feature is enabled. For more information about including formulas in a section to be exported to Excel, see "Formula Mappings" on page 62.

5 For data exported to tab-delimited text files, select the **Export Without Quotes** check box to exclude double quotation marks around real column/cell values in the exported files and click **OK**.

Defining Export Properties for Static HTML in the Pivot and OLAP Sections

Before exporting data to static HTML, use the Export Properties dialog box to specify export properties for data in the Pivot section.

- > To specify export properties:
- **1** Select Format > Export Properties.

The Export Properties dialog box is displayed.

2 Specify the number of vertical rows to include on an HTML page before starting a new page (file) by checking the **Vertical Page Break** checkbox and entering the number of rows.

The default setting is 100 rows per HTML page. (Select Pixels to specify the number of pixels instead of rows). To export to one file, or to have no page break, leave the Vertical Page Break checkbox blank.

3 Specify the number of horizontal columns to include on an HTML page before starting a new page (file) by checking the **Horizontal Page Break** checkbox and entering the number of rows.

The default setting is 100 rows per HTML page. (Select Pixels to specify the number of pixels instead of rows). To export to one file, or to have no page break, leave the Horizontal Page Break checkbox blank.

4 Select the desired export properties.

Check the Export with Style Sheet (CSS) option to create a Cascading Style Sheet file separate from the HTML file. If this field is unchecked, the style sheet information is embedded in the HTML file itself.

- 5 For data exported to tab-delimited text files, select the **Export Without Quotes** check box to exclude double quotation marks around real column/cell values in the exported files.
- 6 Click OK.

Defining Export Properties for Static HTML in the Report Section

Before exporting data to static HTML, use the Export Properties dialog box to specify export properties for data in the Report section.

- > To specify export properties:
- **1** Select Format > Export Properties.

The Export Properties dialog box displayed.

2 Select the desired export properties.

Check the Export with Style Sheet (CSS) option to create a Cascading Style Sheet file separate from the HTML file. If this field is unchecked, the style sheet information is embedded in the HTML file itself.

3 Click OK.

Exporting a Section

After processing a query, you can export the data contents of the Results, Pivot, Chart, Table, or Report sections for use in other applications. If you export from a Results section, the data is raw and unaggregated. If you export from a report section, the data is already aggregated.

- > To export the current section to a file:
- **1** Select File > Export > Section.

The Export Section dialog box is displayed.

- 2 Specify the location where you want to save the file.
- **3** Type a name for the section to be exported in the **File Name** field.
- 4 Select a file format from the Save As Type drop-down list. Select from:
 - Microsoft Office 2000 HTML (*.html)
 - Microsoft Office Web Archive (*.mhtml)
 - Excel (*.xls)
 - Lotus 1-2-3 (*.wks)
 - Text (Tab delimited) (*.txt)
 - Text (Comma Delimited) (*.csv)
 - HTML (*.htm)
 - PDF (*.pdf)

The available export file formats change depending on which section you export.

Note: Select Format > Export Properties to set properties for files exported to text or HTML.

Exporting to Microsoft Office HTML File Formats

You can export and deploy a section on an intranet or internet, and work back and forth between the HTML file and Microsoft Office 2000 Excel. This is achieved by using the Microsoft Office 2000 HTML (*.html) or Microsoft Office Web Archive (*.mhtml) file formats. These file formats preserve and recognize formatting attributes, formulas and Visual Basic for Applications macros (VBA) created in the original section.

This feature uses Microsoft Office XML (extended markup language) tags in addition to the HTML. When the exported section is opened in Microsoft Office (specifically Excel), the formatting and formula from the original section are preserved and recognized. You can modify the file again and save it as an HTML file without the loss of any HMTL code. If you modify the file and save it in Office Excel file format, the HTML tags are not retained.

A section exported to a Microsoft Office 2000 HTML (.html) file format has the main web page saved in one folder and all the graphics and other related information (such as VBA macros) in another.

Microsoft Office 2000 Web Archive (.mhtml or .mhtm) refers to MS HTML, which is a standard for including objects in the same file as the HTML code (for example .gif or.jpeg files). Objects are encoded using the MIME HTML Internet standard. You might use this file format if you plan to email the HTML as a single file.

Note: Any changes made to the section through the HTML Wizard are not recognized when it is exported to Microsoft Office 2000 HTML.

Types of HTML File Formats

Interactive Reporting supports three HTML file formats to which a section can be exported:

- Microsoft Office 2000 HTML (*.html)
- Microsoft Office Web Archive (*.mhtml)
- Standard HTML (*.html)

The export features are supported in the following sections:

- Table and Results
- Chart
- Pivot
- Report

The Dashboard section cannot be exported.

Comparison of HTML File Format Types

The following table shows which export attributes are supported by the HTML file format.

Section	Office HTML Export to Office Excel	Standard HTML Export to Excel
Results/Table	Formatting is preserved. Formulas are supported. (See Formula Mappings).	Formatting is not preserved. Formulas are not supported.
Pivot	Export preserves formatting. Cell merging is supported.	Export does preserve formatting. Export does not support formulas. Cell margins are not supported.
Chart	The Chart section is written to two worksheets: the first worksheet contains the Chart data in a table, and the second worksheet contains the Chart image.	The Chart data is written to a simple table.
	the same way as it is aggregated in the original Chart Section.	
	Tip: When a Chart section has been exported to Excel, you can use the data in the table as a data source and launch the Excel Chart Wizard to create the actual Chart. The rendered Chart looks the same as the one created in Designer.	

 Table 1
 Comparison of HTML File Formats

Section	Office HTML Export to Office Excel	Standard HTML Export to Excel	
Report	The Report data is written to a simple table.	The Report data is written to a simple table.	
	If a report has only one table or only one pivot (no matter how many pictures and/or text labels), then column widths are preserved. Otherwise column width are not maintained. Column width, in this case, will be made sufficient to fit data from any table.		
	Formatting is preserved.		
	Report elements including: text, labels, images, tables, pivots, and charts are placed one after another vertically. A vertical gap of a certain fixed height will exist between two elements. If elements overlap vertically by a small value these elements are with shown no gap between them. This is to accommodate any design that includes stacked elements. The gap is one row high.		
	Report headers/footers will be output.		
	Graphical elements such as lines, circles are not supported.		
	Overlapping tables are not supported and are computed consecutively.		
	Page headers/footers are not supported.		
Dashboard	Not supported except as an image in the export feature	Not supported	

Table 1 Comparison of HTML File Formats (Continued)

Exporting a Section to Microsoft Office Excel Worksheets

Each exported section to Office HTML creates one worksheet in the Excel workbook even when there are multiple sections. Exported sections of the same BQY do not reference each other in the Excel worksheet. For example if a Table section is created from a Results section, both exported sections are displayed as independent worksheets in Excel.

An Internet Explorer browser can show more than 65, 536 rows, and an Excel spreadsheet has 256 columns and a maximum of 65,536 rows. If an exported section has more rows than the Excel row maximum, all rows are exported to the source rows available, but only 65, 536 rows are processed for Excel to process any formulas. In this case, Excel truncates the exported data in the section.

Number and Date Formats

Number and date formats in Interactive Reporting sections are exported based on the following predicates:

- Any text within BI format strings is enclosed double-quotes (" ").
- Interactive Reporting fills the number formats for positive, negative and zero numbers to achieve compatibility with Excel's construction. Excel number format definition consists of the above number formats, delimited with a semi-colon.
- Date/time formats are exported as is except for the am/pm format, which receives a special format.
- Thousands and decimal separators are converted to Excel special symbols based on the default BI locale format. When the Interactive Reporting number format is evaluated, the format is converted so that the decimal separator becomes "." and thousand separator becomes "," in the target Excel format. Excel treats these special symbols as locale independent separators and replaces them with actual separators from user's locale on the run-time.
- A value in an Excel cell has two aspects: the value formatted for displaying and the original value. Similarly when a value is placed in a HTML table, a special attribute x:num="<actual value>" is used to preserve the original value. The pre-formatted value is placed in a cell of the HTML table, and it can be shown in the web browser. Formulas operate with the original values. The x:str attribute is used as a global designator in a html in order to tell Excel that all the values which do not have x:num attribute specified should be treated as strings. This is useful when there is mixed string and numeric data in a string type column.

The third component associated with a HTML table cell is a style class. It is used to specify the custom numeric format for a cell value. Date and time values are represented as numbers. These values and saved in the x:num attribute, and the style class definition specifies the format for date.

Styles

Exported sections support styles (CSS) that have been applied to the section including: font name and size, and bold/italic properties.

The Interactive Reporting overline and double overline properties are not supported in Excel, because Excel has no equivalent styles.

The simple overline style is supported in a browser: the double overline is not. In addition, the overline applies only to the text, not the whole cell as in Designer.

The Spotlighter feature is not supported in Excel. The appearance of a cell to which the spotlighter has been applied, retains the same look as in the exported spreadsheet, but without the real auto-formatting. Excel has an equivalent feature called Conditional Formatting. This feature can use a formula returning a Boolean result as a condition.

Colors

Excel supports a fifty-six color palette of which sixteen are used exclusively for Charts, and forty for cells in a spreadsheet. Colors can be defined by using RGB notation, and shared among all worksheets in a workbook.

All colors displayed in the sections being exported are saved as a custom palette and saved in the HTML file (in a hidden xml section). The color palette that is used in Excel by default, is used as a basis for creating a custom palette. The colors that match Excel's retain their positions. Unused positions are filled by default Excel colors.

Excel automatically determines which index in the color palette to use for a particular color occurring in CSS.

If the total number of colors is more than forty, Excel determines the color placement.

Text Wrapping

If text wrapping is disabled in Designer and the section is exported, Excel sizes the column so that all values within a row are displayed, but the words that do not fit into the cell are wrapped to the next line and therefore are not visible. In the browser, the text is truncated.

Chart as Image

A worksheet containing a chart image shows no cell grid.

Charts that take up more than single image (this is when chart scrolling is enabled) are placed on a single worksheet, and page breaks are provided. Each image is printed on its own page.

Headers and Footers

Headers and footers are converted into appropriate Excel equivalents, and are displayed in a printed document. In addition, the format template is converted to the Excel format. Multiple headers/footers are supported; but they are converted into a single multi-line header or footer.

Headers and footers are not displayed when the exported section is opened in a web browser.

Printing Improvements

Results/Table and Pivot sections are exported so that the table headings and pivot top/side labels are marked. These marks allow headings and labels to be duplicated on each page when they are printed from Excel.

Suppress Duplicates for Results or Table

The Suppress Duplicate feature hides duplicating values in a selected column, but duplicates are included in calculations. The value can be seen and changed only by clicking a cell in Excel. Typing any new value will not change the hidden status of the cell unless the user changes the cell format.

Formula Generation

An exported BQY table section can have the following kinds of JavaScript expressions:

- Grand/Break Total calculation (for example, Sum(Units, Breaks) + 10)
- Grand/Break Total labels (for example, "Total "+ ToChar(Store))
- Computed Item expressions (for example, Units*2 + 10)

When exporting a table with expressions, the following steps are performed:

- Values that are calculated using formulas are exported to HTML, and available to be shown in a web browser.
- Excel formulas are generated for each cell in the table and written to HTML. This file can be opened then in Excel and all the values are re-calculated. An end-user can change values in the cells and results will be re-calculated again.

JavaScript Expressions

JavaScript expressions used with the Export to HTML feature can refer to columns in the table and invoke computed items. They can also refer to any JavaScript class like Math and String. Those computed items which do not have direct equivalent among the built-in Excel functions are simulated by custom VB functions. For information about how Interactive Reporting computed items map to Excel's built-in function, see "Formula Mappings" on page 62.

If the expression refers to a JavaScript class not supported by the Export feature, then #NAME? is displayed in the Excel cell (and as a valid value in the web browser) and the seer office html tag attribute is used. In addition, a comment is to the cell explaining the error. In Excel, this comment is displayed near the cell where error occurred. In browser, the comment is displayed when the mouse pointer is hovered over the cell with a small red-colored corner.

Mapping JavaScript to Excel

Interactive Reporting uses a translator to distinguish range type which are used to substitute column reference in a computed item expression. There are three types of ranges:

- Continuous range which corresponds to the whole column (for example, Units)
- Aggregate range this is a continuous range which gets broken by Break Totals (for example Units_Agg). The only purpose of this range is to be a data source for aggregate functions (hence the name). The translator uses Units in all kind of expressions that refer to the column, but it uses Units_Agg range for aggregate functions, for example, SUM(Units_Agg).
- Continuous sub-range used in break total computation.
- Individual cell references (for example, A1, B5)

The table below outlines how the JavaScript expression operations are mapped to Excel.

JavaScript	MS Excel
*, /, <=, =>, -, + (arithmetical plus operator)	Uses the same symbols.
%	MOD() built-in function
!=	<>
	=
+ (concatenation operator)	&
Exp1 ? Exp2 : Exp3 (ternary operator)	IF(Exp1, Exp2, Exp3)
Comma operator (e.g. exp1, exp2,, expN)	Replaced with the last expression. For example, (exp1, exp2,, expN) -> replace with expN in resulting Excel formula. Expressions should be of the same type (expression or string)
!	NOT() built-in function
&&	AND() built-in function
	OR() built-in function
Character literal (e.g. 'a')	Converts to string literal (for example, "a")
Hexadecimal and octal number (e.g. 0x10, 010)	Converts to a decimal number (for example, 16, 8)
Logical constants (true and false)	Converts to Excel's TRUE and FALSE
Expressions with 'null'	Expressions with 'null' have limited support. Column values can be compared with null. Computed Items should not produce null values.
nt r a b and xHH in string literals. Example: "a na"	Replaced with CHAR(x) in output string.For example: "a" & CHAR(10) & "a"

The JavaScript' '+' operator can be applied to strings and denotes a string concatenation. Excel's string concatenation operator is '&'. Interactive Reporting distinguishes between arithmetic '+' and string '+' by tracking the type of arguments in expressions. If one of the arguments is a string literal, then a concatenation operator is used. The translator tracks return type of Object Model functions and columns, so it can infer final expression type. There are three supported types: numeric, string and date.

Formula Mappings

Most of the computed items that comprise computed item expressions can be mapped directly to Excel functions. Certain functions have different numbers of arguments and/or argument order and require additional processing.

The following table shows which Interactive Reporting computed items map to Excel formulas.

Interactive Reporting Computed Item	Microsoft Excel Function	Implementation	Aggregation
Decode	H_Decode	VBA	
Nvl	Limited support: Nvl(column, expr).	Excel	
AddMonth	H_AddMonth	VBA	
DayOfMonth	DAY	Excel	
LastDay	H_LastDay	VBA	
MonthsBetween	H_MonthsBetween	VBA	
NextDay	H_NextDay	VBA	
Sysdate	NOW	Excel	
ToChar(value, format) ToChar(column)	Limited Support: Excel's built-in TEXT(val, format) function is used and the format argument is translated from the BI format into Excel's on the formula generation time. Consequently, the format should always be a constant string literal. When the argument for ToChar is a column reference, then converts to TEXT(column, format), where format is one of real/int/date_format depending on the column type	Excel	
ToDate(x)	(DATEVALUE(x) + TIMEVALUE(x))	Excel	
ToMonth	H_ToMonth	VBA	
ToQtr	H_ToQtr	VBA	
ToYear	YEAR	Excel	
Abs	ABS	Excel	
Atn	ATAN	Excel	
Ceil	H_Ceil	VBA	
Cos	COS	Excel	
Cosh	COSH	Excel	

 Table 2
 Computed Items to Excel Function Comparison of HTML File Formats

Interactive Reporting Computed Item	Microsoft Excel Function	Implementation	Aggregation
Ехр	EXP	Excel	
Floor	H_Floor	Excel	
Ln	LN	Excel	
Log	LOG	Excel	
Log10	LOG10	Excel	
Мах	МАХ	Excel	
Min	MIN	Excel	
Mod	H_MOD	VBA	
Power	POWER	Excel	
Round	ROUND	Excel	
Sign	SIGN	Excel	
Sin	SIN	Excel	
Sinh	SINH	Excel	
Tan	TAN	Excel	
Tanh	TANH	Excel	
Trunc	TRUNC	Excel	
Avg	H_Avg	VBA	~
AvgNonNull	AVERAGE	Excel	~
Chr	H_Chr	VBA	
ColMax	H_ColMax	VBA	~
ColMin	H_ColMin	VBA	~
Count	H_Count	VBA	~
CountDistinct	H_CountDistinct. Limited Support: Can be used in Break/Grand Totals ONLY (performance considerations as it uses heavy computations).	VBA/Excel	V
CountNonNull	COUNTA	Excel	V
CountNull	H_CountNull	VBA	~

					() ()
lable 2 C	computed Items to	Excel Function Co	mparison of HIML	File Formats	(Continued)

Interactive Reporting Computed Item	Microsoft Excel Function	Implementation	Aggregation
Cume	Computed Items with expressions that have exactly one call to Cume() and nothing else. Example: "Cume(Units)"	Excel/VBA	Aggregation and cross references in rows
Next	Function call is converted to appropriate cell reference. Limited support: Cannot be used in Break or Grand Total expression.	Excel/VBA	Cross references in rows
Prior	Function call is converted to appropriate cell reference. Limited support: Cannot be used in Break or Grand Total expression.	Excel	Cross references in rows
Sum	Use Excel's SUM() for non- string columns. String columns exception: H_SumStr() will be used here. Can be used in Break and Grand Totals only. (performance considerations as it uses heavy computations).	Excel/VBA	~
Median	MEDIAN	Excel	~
Mode	H_Mode Mode maps to Excel's H_Mode because Excel's MODE does not support complex ranges. Mode can only be used in Break and Grand Totals. There may be performance considerations associated with this computed item because it uses heavy computations.	VBA	r
Percentile	H_Percentile	VBA	~

Table 2 Computed Items to Excel Function Comparison of HTML File Formats (Continued)

Interactive Reporting Computed Item	Microsoft Excel Function	Implementation	Aggregation	
Rank(column)	H-Rank Rank maps to Excel's H_Rank function. Rank can only be used in Break/Grand Totals expressions. There may be performance considerations associated with this computed item because it uses heavy computations.	VBA	~	
RankAsc(column)	H-RankAsc This function can only be used in Break andGrand Totals expressions. There may be performance considerations associated with this computed item because it uses heavy computations.	VBA	V	
StdDev	H-StdDev	VBAI	~	
StdDevp	H_StdDevp	VBA	~	
Var	Var	VBA	~	
Varp	Varp	VBA	~	
Ascii	CODE	Excel		
Concat	CONCATENATE	Excel		
Initcap	H_Initcap	VBA		
Instr	H_Instr	VBA		
Length	LEN	Excel		
Lower	LOWER	Excel		
Ltrim	H_Ltrim	VBA		
Replace	H_Replace	VBA		
Rtrim	H_Rtrim	VBA		
Substr	MID	Excel		
Translate	H_Translate	VBA		
Upper	UPPER	Excel		
MovingAvg	H_MovingAvg	VBA	Cross references in rows	
MovingDiff	H_MovingDiff	VBA	Cross references in rows	

Table 2	Computed Items to Excel Function	tion Comparison of HTML	File Formats (Continued)
10.010 =			

Interactive Reporting Computed Item	Microsoft Excel Function	Implementation	Aggregation
MovingMax	H_MovingMax	VBA	Cross references in rows
MovingMed	H_MovingMed	VBA	Cross references in rows
MovingMin	H_MovingMin	VBA	Cross references in rows
MovingSum	H_MovingSum	VBA	Cross references in rows
All aggregate functions with two and three arguments: Ex. Sum(col, break_col, break_value)	When two-argument functions are used in Break/Grand Total expressions and the second argument is 'Breaks', most of the functions are converted to their built-in Excel equivalents. Otherwise, the function is substituted with the appropriate H_xxx() VBA function.	Excel/VBA	

 Table 2
 Computed Items to Excel Function Comparison of HTML File Formats (Continued)

Custom Formulas

Interactive Reporting has a number of computed items that do not map directly to an equivalent Excel function. This section describes the methods the Export feature uses to construct a mapping from the Interactive Reportingcomputed item to an Excel function.

Adding VBA functions to the Exported Excel File

Interactive Reporting helper functions implemented in Visual Basic are saved to the exported file. In this instance, the Excel file becomes self-contained. It can be passed to any user 'as is' and does not require installing additional components such as Excel add-ins. All the functions are implemented within this file

Since VBA functions are treated in Excel as macros, Interactive Reporting signs them digitally in order to prevent a 'security warning' dialog in Excel if the macro security level is Medium or High. When the macros are signed, the end user is able to observe certification information and choose to trust the sources. Once the certificate is marked as trusted, all the subsequent openings of exported excel files proceed silently.

Unsupported Formulas

Excel has a limitation for formula content length. This limit is 1,024 characters.

In Interactive Reporting, a null value is absent of data, but does not equal zero. Excel does not recognize null values, but has a similar concept. Cells that are absent of data are represented by blank cells. There are also built-in aggregation functions analogs which understand this concept

Custom Cell Ranges

Named cell ranges are generated to construct Excel formulas for computed items in spreadsheet. For example, if there is a Table section with Break Totals, then the following types of named ranges are generated:

• A continuous cell range that includes the whole column.

For example: Units = Results!\$C\$2:\$C\$998. This range can be used in all kin of formula expressions, but cannot be passed to aggregate functions like SUM.

• A compound or broken cell range that includes all the cells with actual data excluding those occupied by Total information.

For example: Units_Agg = Results!\$C\$2:\$C\$4,Results!\$C\$6:\$C\$34,Results!\$C\$36:\$C\$164. This range can be passed to aggregate functions, which is reflected in its the name.

• A cell range used in break totals calculations.

For example: Sr_32_2=Results!\$C\$2:\$C\$4

Below is a sample spreadsheet generated from the Results section. The column 'Computed' of the section contained Units*2 expression, the column 'Computed2' contained SUM(Units) expression. The generated Excel formulas may be seen in the cells.

	A	В	С	D	E
1	Store	Item Name	Units	Computed	Computed2
2	Bayshore Electronics	1024 MB Drive	45	=Units*2	=SUM(Units_Agg)
3	Bayshore Electronics	EZ Fax Modem-I	30	60	21265
4	Bayshore Electronics	EZ Fax Modem-x	45	90	21265
5	Bayshore Electronics	One Button Mouse-	30	60	21265
6	Bayshore Electronics	The Quad	35	70	21265
7	Bayshore Electronics	One Button	30	60	21265
8	Bayshore Electronics	One Button Mouse-	40	80	21265
9	Bayshore Electronics	One Button Mouse-	35	70	21265
10	Total Bayshore Electronics		475	=SUM(Sr_13_2)	

For the Excel user, the Auto Outlining feature can be used to create row groups from the Break Totals that Interactive Reporting creates. Excel analyzes the formulas and creates the row groups and outline automatically. The user can then expand/collapse individual groups of rows. Named cell ranges created in Interactive Reporting can be used in the in-sheet formulas and in an external Excel file (worksheet linking). For example, an end user can create a new Excel file and put a formula to a cell referring to a data column of the workbook file exported from Interactive Reporting: =SUM(BIExport.xls!Results_Units_Agg), where Results_Units_Agg is an auto-generated range that denotes the Units column of the Results table in the BIExport.xls file.

Worksheet Names and Range Names Generation

Worksheet names are based on the BQY section names. Before exporting a section, consider the following limitations that apply to worksheet names:

- The name of section to be exported should not be longer than thirty-one characters in the name
- The symbols / \ ? * are not allowed and are removed.
- Duplicate worksheet names are appended with a number to ensure the uniqueness of each worksheet.
- A worksheet name can include space symbols. Leaving space symbols in worksheet names improves user experience. When such a worksheet name is used in formulas, it should be surrounded with single quotation marks.

Generated range names that correspond to BQY table columns are based on a respective column name. The following restrictions apply to generated range names:

- The worksheet name is added to the beginning of the range name to ensure its uniqueness within the whole workbook.
- Spaces within the name are replaced by underscores.
- The symbols $\sim !@#$ %% $&^()_+ |{}: <>? -= [];, / are not allowed and are removed.$
- Duplicate range name are appended with a number to ensure the uniqueness of each range name.should be provided.
- Columns must have appropriate identifiers in JavaScript, for instance, "Order Date" column can be referred to as "Order_Date" in a computed item expression. This identifier is used as the main part of a generated range name.
- Excel has a limitation on the length of the expression used as a named range. The total length of the of references (not including worksheet names) can not be more than 255 characters. When a named range is generated, its content is broken into smaller chunks, each comprising a smaller named range. The final range is a union of these smaller ranges.

Exporting a Document as a Web Page

Use the Export Document As Web Page command to publish the contents of a document as a Web page. You can select which sections of the current document to include in the export set.

Documents sections are exported in the same order as they is displayed in the document. Exportable document sections include: Results, Table, Pivot,

Chart, OLAPQuery, and reports created with the Report Designer. Imported sections that have been added to the document, such as text files and Excel files, can also be exported. Data models and the Query, and Dashboard sections are not exportable. Neither are empty or blank sections. After these files are posted to a Web site, you can access the files individually or use the main HTML file to view the frameset, which lists the sections in the report. The exported Web pages include navigation buttons so you can scroll back and forth between pages, or jump to the beginning or end of a document. In addition, the current page number and total number of pages are included in the report.

- **Note:** The Export As HTML and HTML Wizard options in prior versions of Interactive Reporting remain unchanged. However, we recommend that you use the Export As Web Page option.
 - > To export the document as a Web page:
 - **1** Select File > Export > Document As Web Page.

The Export Document As Web Page dialog box is displayed.

2 Select the sections of the document that you want to export and click **OK**.

To select all sections, click Select All. To clear all selected sections, click Deselect All.

The Save As Type dialog box is displayed.

- 3 Navigate to the location where you want to save your Web page.
- **Tip:** Create a separate folder to store the files created by this export option.
 - 4 Type a name for the exported Web page in the File Name field, or accept the default name, for example Sample1.htm.

The Save As Type field is set to *.htm by default. All of the HTML files that the export process creates begin with the name specified in the File Name field, for example, SamplelChart.htm, SamplelPivot.htm, and so on. The HTML file with the exact same name as the name specified in the File Name field is the main HTML page, and it contains the frameset and links to all of the other pages, for example Sample1.htm.

5 Click Save.

To view the exported selection, open your Web browser, select File > Open, browse to the location of the exported files, and open the *main* HTML page.

Saving an Excel file to HTML format

If you plan to import an HTML file that was created from Excel, note the following restrictions:

- Prior to saving an Excel file to HTML format, turn off Excel's password protect feature.
- Formulas are saved as sheet references in Excel. When an Excel workbook is saved to HTML workbook, formulas linked to a cell in the HTML workbook do not operate.
- Custom view and scenarios can only be used in Excel.
- Add-ins and templates are only available in Excel.

Browser and HTML Restrictions and Limitations

Browser and HTML restrictions may affect how graphics and formatting are displayed on Web pages created by exporting document sections. The known restrictions and limitations are:

- Diagonal lines, ovals, round rectangles, and dotted or dashed lines do not export to the Web page. Overline or double-overline text is displayed as regular text.
- The Netscape browser shows data formatted with bold Arial 8 pt. as regular Arial 8 pt.
- Data formatted with Arial 14 pt. is displayed smaller in Interactive Reporting Studio and Netscape than in Microsoft Word and Internet Explorer.
- Border properties (including color properties) are supported in Microsoft Internet Explorer, but not in Netscape. Consequently, border properties for the Results, Table, and Pivot sections are not displayed on Web pages opened with Netscape. If you embed a Results, Table, or Pivot section in a report, Netscape does recognizes the border properties.
- Raised and sunken borders are displayed as regular borders.
- Lines and rectangles are not displayed in reports in UNIX browsers.
- The right border of a table embedded in a report sometimes is displayed thicker in Netscape.
- The Picture Tile property is not supported by HTML.
- Word-wrapped fields contain hard-coded leading spaces for left padding. HTML permits the browser to implement word-wrap, but eliminates extraneous space. In addition, a browser word-wrap feature does not break a word in the middle of a word regardless of its length.
- When empty table cells are displayed with a certain font, it is because Interactive Reporting Studio inserts a single blank into each empty cell. HTML requires a single value in each empty cell.
- HTML does not recognize vertical text.

Using the Export to HTML Wizard

The Export To HTML Wizard helps you create Web pages from existing charts, reports, and pivot tables.

- **Note:** We suggest you use the Export Document As Web Page command (see "Exporting a Document as a Web Page" on page 69) rather than the HTML Wizard.
 - To use the Export to HTML Wizard, select **File > Export > HTML Wizard**.

Exporting SQL

Use the Export SQL command to export the SQL statement for your query. The file is saved in an SQL format.

- ➤ To export SQL:
- 1 Select File > Export > SQL.

The Export SQL File dialog box is displayed.

2 Specify the file name and location and click Save.

Exporting a Query Log

When you process a query, translates your request into SQL or a multidimensional database query statement and forwards it to the database server.

- > To save the contents of the SQL log to a text file:
- 1 Select File > Export > Query Log.

The Export Query Log dialog box is displayed.

2 Specify the file name and location, and then click Save.

Printing Documents

Interactive Reporting StudioWeb client printing functions are available for most document sections. You can specify the page setup for your printer, print directly to a printer, or preview a print job onscreen.

Page Setup

Use the Page Setup command to specify the default printer for theInteractive Reporting StudioWeb client document and to define default page properties for that printer.

> To define page setup information, select **File > Page Setup**.

Print Preview

Use the Print Preview command to view onscreen a representation of the printed version of a finished report. The Print Preview command is available for all sections except the Dashboard. In the Report Designer section, the page view of the report is a direct representation of the printed report.

To preview a section before sending it to a printer, select File > Print Preview.

A preview of the current section is displayed in the Content pane, and a Print Preview menu is added to the Main menu. Use the commands on the Print Preview menu to navigate through the preview pages and to specify starting page numbers.

Print

Use the Print command to print the information displayed in the Content pane of most sections.

To print the information in the document, select File > Print.

Sending Documents

You can attach documents to your electronic mail (email) if you have a MAPI-compliant email system such as Microsoft Mail or Microsoft Exchange.

- To email a document:
- 1 Select File > Send.

The Mail Document window is displayed.

2 Select the name of the document recipient.

You can send the document with or without the results. Sending a BQY document with the results enables the recipient to do further analysis.

A document sent without the results contains snapshots of the Chart, Pivot, and other reporting sections, but not of reports created using the Report Designer. No further analysis is possible.

3 Select whether to copy other recipients or include additional remarks, then click Send It.

Quitting an Interactive Reporting Studio Session

To end your Interactive Reporting Studio session:

Select File > Exit.

If documents remain open, you are prompted to save changes to each document before it shuts down.

Quitting the Interactive Reporting Web Client

> To end your work session in the Web client:

If documents remain open, the Web client prompts you to save changes to each document before closing it.
File Menu Command Reference

Table 3 provides a quick reference to the commands available on the File menu and list any related shortcut menus. The commands listed are those available in Query and Results sections.

Command	Description	Keyboard Shortcut
Save	Saves the active document	[Ctrl+S]
Save As	Opens the Save As dialog box.	
Save Options	Allows you to specify save options for the active document.	
Import Data File	Imports data stored in Microsoft Excel, tab-delimited, or comma-delimited file formats to the Results section for analysis and reporting.	
Export	Exports data to the selected format.	
Page Setup	Opens the Print Setup dialog box.	
Print Preview	Toggles Print Preview mode and the Print Preview menu. Displays a preview of the current section in the Content pane.	
Print	Prints the information displayed in the Content pane.	[Ctrl+P]

 Table 3
 File Menu Commands

Using Edit Commands

The Edit menu contains standard editing commands. It also contains commands that allow you to work with document sections. (See "Formatting Numeric Data Types" on page 79 for detailed information on working with document sections.)

Table 4 provides a quick reference to the commands available on the Edit menu and lists any related shortcuts. The commands listed are those available in the Query and Results sections.

Command	Description	Keyboard Shortcut	Shortcut Menu
Undo	Reverses the last command issued.	[Ctrl+Z]	
Redo	Re-applies the actions or commands on which you have used the Undo command.	[Ctrl+Y]	
Cut	Cuts the selected item.	[Ctrl+X]	~
Сору	Copies the selected item.	[Ctrl+C]	
Paste	Paste the last copied item.	[Ctrl+V]	

 Table 4
 Edit Menu Commands

Command	Description	Keyboard Shortcut	Shortcut Menu
Clear	Clears entry fields in dialog boxes.		
Select All	Select all items or elements, depending on the location of the cursor.	[Ctrl+A]	
Delete Section	Opens the Delete Section dialog box.		~
Rename Section	Opens the Section Label dialog box.		~
Duplicate Section	Duplicates the selected section and adds the copy to the Section pane with a sequenced number appended to the section name.		V

Changing Workspace Views

The View menu allows you to toggle the display of interface elements, such as panes, toolbars, and so on. It also provides commands for working with sections. (See "Formatting Numeric Data Types" on page 79 for detailed information on working with document sections.)

Table 5 provides a quick reference to the commands available on the View menu and list any related shortcuts. The commands listed are those available in the Query and Results sections.

Command	Description	Shortcut Menu
Section/Catalog	Toggles the display of the Section and Catalog panes.	
Section Title Bar	Toggles the display of the Section title bar.	
Toolbars	Toggles the display of the Standard, Format, and Section toolbars.	
Status Bar	Toggles the display of the Status bar.	
Console Window	Opens the Console window, which is used to display error messages and alert values generated by the JavaScript interpreter.	
Go To Section	Navigates to the section selected from the list of sections in the current document.	
Hide Section	Hides the active section.	
Unhide Section	Opens the Unhide Section dialog box which lists currently hidden sections.	
Query Log	Opens the Query Log dialog box, which displays the command language statement for the active query.	
Zoom	Changes the display of the active section to the selected zoom setting.	
Hide Request Item	Hides the selected Request item from view.	~

Table 5 View Menu Commands

Table	5	View	Menu	Commands	(Continued)
-------	---	------	------	----------	-------------

Command	Description	Shortcut Menu
Unhide Request Item	Opens the Unhide Columns dialog box, which lists currently hidden Request items.	~
SQL	Opens the Custom SQL window where you can edit the SQL statement for a query.	
Properties	Opens the Properties dialog box for the selected topic or topic item.	✓

Inserting Sections and Breaks

Use the commands on the Insert menu to insert new sections in your Interactive Reporting Studio document. You can also insert page headers and footers for use when printing certain sections.

Table 6 provides a quick reference to the commands available on the Insert menu and list any related shortcut menus. The commands listed are those available in the Query and Results sections.

Command	Description
New Query	Inserts a new Query section.
New OLAP Query	Inserts a new OLAPQuery section.
New Table	Inserts a new Table section.
New Pivot	Inserts a new Pivot section.
New Chart	Inserts a new Chart section.
New Report	Inserts a new Report section.
New Dashboard	Inserts a new Dashboard section.
Page Header	Inserts a page header that is used when the section is printed.
Page Footer	Inserts a page footer that is used when the section is printed.

 Table 6
 Insert Menu Commands

Note: See "Formatting Numeric Data Types" on page 79 for detailed information on adding document sections and customizing the headers and footers in the document sections.

Formatting Text and Other Elements

Use the commands on the Format menu to change the formatting properties of text, numbers, borders, rows, columns, exported documents, and so on. Most of these commands can also be found on the Formatting toolbar (see "Formatting Toolbar" on page 42).

Table 7 provides a quick reference to the commands available on the Format menu and lists any related shortcut menus. The commands listed are those available in the Query, Results, and OLAPQuery sections.

Command	Description	Keyboard Shortcut	Shortcut Menu
Font	Opens the Font page of the Properties dialog box.		~
Style	Select between Plain, Bold, Italics, Underline, Overline, Double Underline.		
Number	Opens the Number page of the Properties dialog box.		~
Justify	Select between Left, Center, Right and Top, Middle, Bottom.		V
Column	By default, Results columns are evenly sized without regard to the length of data values. Numeric data that does not fit is replaced with pound signs (#).		
	To manually resize a column, drag the right edge of the column to a new position.		
Row	Resizes all rows to the standard row height.		
	To resize all rows in a table, drag the bottom edge of a row to a new position. All rows in the table are resized.		
Column Titles	Toggles the display of column titles.		
Row Numbers	Toggles the display of row numbers. Row numbers are printed on reports, but are not copied to the clipboard or exported to a file.		
Text Wrap	Wraps text in a column.		V
Suppress Duplicates	Suppresses duplicate values in a column. Use this feature if you want to display only the first instance of a duplicate value when individual database records include redundant information.		r
Grid Lines	Opens the Gridlines page of the Properties dialog box.		
Border and Background	Opens the Border and Background page of the Properties dialog box.		
Spotlighter	Opens the Spotlight dialog box. Use to spotlight important values.		
Export Properties	Opens the Export Properties dialog box. Use to set the number of rows that should be included on an HTML page before the data breaks to another page, and to export data that does not contain any quote to a tab-delimited text file.		

Table	7	Format	Menu	Commands
Iavic		iunat	INICIIU	Commanus

Tip: To automatically size all columns so that the column width fits its contents, press [Ctrl+A] [Ctrl+E].

Note: See "Formatting Numeric Data Types" on page 79 for detailed information on working with document sections.

Formatting Numeric Data Types

This section discusses formatting numeric data types in the following areas:

- Changing Numeric Formatting
- =Working with Document Sections

Changing Numeric Formatting

You can change the formatting properties of numeric data types (real and integer) in the following ways:

- Select an object, right-click to access the Number shortcut menu, and select a numeric formatting option from the menu.
- Use the buttons on the right-hand side of the Formatting toolbar.

Displaying Numbers in Scientific Notation

enables you to display numbers in scientific notation. The default scientific notation format is:

0.00E+000

The scientific notation format is displayed using the appropriate decimal separator for the current local. (The above example uses a period as the decimal separator.)

Table 8 describes the acceptable variations on the default scientific notation format:

Variation	Description
The number of decimal positions after the decimal separator.	For example, 0.00000E+000 displays six digits of precision.
#	Used after the decimal point to suppress zeros. For example, the value .000179 with the format 0.00000E-00 displays as 1.79000E-04. With the format 0.0####E-00, the same value displays as 1.79E-04.
	With the format 0.0####E-00, the same value displays as 1.79E-04.
Eore	
E0 E+0 E-0 (and the lower case equivalents)	If the plus sign is used, the exponential component always displays with a sign (plus or minus). If the minus sign or no sign is used, the sign displays only when negative.

Table 8 Variations on the Default Scientific Notation Format

Variation	Description
E+0 to E+000	The number of zeros following the E or e, with or without a sign character, is the minimum number of digits to display for the exponent. For example, the value 179 displays as 1.79E2 with the format 0.00E0 and as 1.79e002 with the format 0.00e000.
	If the minimum number of digits is not adequate to represent the magnitude of the number, additional digits are added as required. For example, the value 1,789,000,000,000 with the format 0.00E0 displays as 1.79E12.
Leading + or - sign	A leading minus sign (or no sign character) displays a sign only if the precision of the number is negative. The appropriate sign always displays if the plus character leads the string.

 Table 8
 Variations on the Default Scientific Notation Format (Continued)

Note: You cannot apply scientific notation to non-numeric data types. If you try to do so, the formatting remains the same.

Numeric formatting buttons on the Formatting toolbar are shown below:

Button	Description
₩ •	Currency – Applies currency formatting to the selected numeric object(s). The currency formatting applied is the first currency type for the selected locale specified in the BQFORMAT.INI file. To display all the available currency types for the current local, click the arrow to the right of the Currency button.
	Note : The Currency button is ignored if the selected object is not of a numeric type (real or integer).
% •	Percentage – Applies percentage formatting to the selected numeric object(s). The percentage formatting applied is the first percentage format for the selected locale specified in the BQFORMAT.INI file. To display all the available percentage formats for the current local, click the arrow to the right of the Percentage button.
	Note : The Percentage button is ignored if the selected object is not of a numeric type (real or integer).
3	Comma Formatting – Toggles the presence of the thousands-separator character on or off. If on, comma formatting inserts the thousands-separator character into the current format string for the selected object. The thousands-separator character is inserted based on current locale settings.
	Note: Comma formatting does not affect numbers formatted with scientific notation.

 Table 9
 Numeric Formatting Options

Table 9 Numeric Formatting Options (Continued)

Button	Description
↓ .0 .00	Decimal Increase – Increases the number of displayed digits after the decimal point (or its locale-specific equivalent).
	For example, if the current format for the field specifies three decimal digits, such as \$#.000, pressing Decimal Increase modifies the format to \$#.0000.
	Pressing Decimal Increase when the number of decimal positions is zero adds the decimal separator character in addition to the one decimal digit. For example, if the format is ##%, pressing Decimal Increase changes the format to ##.0%.
	Note : The Decimal Increase button is ignored if the selected object is not of a numeric type (real or integer).
.00 →.0	Decimal Decrease – Decreases the number of displayed digits after the decimal point (or its locale-specific equivalent).
	For example, if the current format for the field specifies three decimal digits, such as \$#.000, pressing Decimal Decrease modifies the format to \$#.00.
	If the number of decimal positions goes from one to zero when pressing Decimal Decrease, the decimal separator is no longer part of the format. Pressing Decimal Decrease when the number of decimal positions is already at zero results in no action.
	Note : The Decimal Decrease button is ignored if the selected object is not of a numeric type (real or integer).

-Working with Document Sections

Working with document sections involves:

- Understanding Document Sections
- Adding Sections
- Viewing Sections
- Moving Between Sections
- Duplicating Sections
- Renaming Sections
- Adding Headers and Footers to Sections
- Deleting Sections

Understanding Document Sections

Documents are divided into multiple sections, each of which governs one step of the query and reporting procedure. You create sections progressively as you query a database, retrieve results, and generate reports.

A document usually includes Query and Results sections. From the Results section, you can create multiple Pivot, Chart, Table and Report sections to analyze and present data. You can also create Dashboard sections, which provide an automated push-button interface to a document.

Each section occupies an independent window and performs distinct operations. You can move back and forth between sections at any time to rebuild your query or alter your result data. You can also position sections side-by-side in multiple windows.

To see a graphical representation of a document section, click the desired section name in the Section pane.

Figure 1 shows some example document sections displayed in the Section pane.



×



Sections

🌃 Dashboard

Adding Sections

> To insert a new section in a document, select Insert > New Section.

For example, to insert a new Chart, select **Insert** > **Chart**, to insert a new Table, select **Insert** > **Table**.

inserts the new section and adds a new section label to the Section pane. The section label is based on the type of section added. A sequence number is added to the section label if a section with the same name already exists.

Viewing Sections

You can hide sections to simplify your view of the workspace. This allows you to concentrate on only those sections in which you want to work.

- To hide a section:
- 1 In the Section pane, select the section that you want to hide.
- 2 Select View > Hide Section.
- To view a hidden section:
- 1 Select View > Unhide Section.

The Unhide Sections dialog box is displayed.

2 Select the hidden section that you want to view and click OK.

The section is displayed in the Section pane.

Moving Between Sections

Although each section occupies an independent window and performs distinct operations, you can move back and forth between sections at any time to rebuild your query or alter your results data. You can also position sections side-by-side in multiple windows.

You can easily navigate between sections to work on queries, results, and reports.

To move between sections, use one of the following options:

- Select the desired section in the Section pane.
- Click the arrow keys on the Section title bar.
- Select View > Go To Section > Section.

Duplicating Sections

You can copy Query, OLAPQuery, Chart, Pivot and Dashboard sections if the Duplicatable feature is selected.

- To make a section duplicatable, select the section label in the Section pane and select Edit > Duplicatable.
- To duplicate a section, select the section label in the Section pane and select Edit > Duplicate Section.

Interactive Reporting duplicates the section and adds a new section label to the Section pane. The new section label is based on the original section label, but a sequence number is appended to the label. For example, if you duplicate a section named SalesChart three times, the Section pane would show: SalesChart, SalesChart2, SalesChart3, and SalesChart4.

Renaming Sections

The first section that you create is given the default section name, for example, Results. When you insert new sections of the same type as those that already exist, they are numbered sequentially, for example, Results2. To assign sections different or unique names based on your application, use the Rename command.

- > To rename a section:
- **1** In the Section pane, select the section label.
- 2 Select Edit > Rename.

You can also click **Rename Section** on the shortcut menu, or double-click the desired section. The Section Label dialog box is displayed.

Section Label	
Label:	OK
Query	Cancel

3 Type a new name in the Label field and click **OK**.

Adding Headers and Footers to Sections

You can add custom headers and footers and page numbers to your printed section.

- To add a header or footer:
- **1** Select File > Print Preview.

A preview of the current section is displayed in the Content pane.

2 Select Insert > Page Header (or Page Footer).

The Edit Header (or Edit Footer) dialog box is displayed.



3 Enter the desired text or use the buttons in the dialog box to add current date, time, file name, page, page total, or limit values and click **OK**.

The Interactive Reporting StudioThe Web client adds the new header or footer to your report. You can change the font properties and alignment of headers and footers, but you cannot add color.

Editing a Header or Footer

To edit a header or footer, double-click the header or footer you want to edit, make any desired changes, and then click **OK**.

Deleting Sections

You can delete a section, but do so with care. Some sections are dependent on other sections. Deleting one section could also delete one or more sections that you did not want to delete. Note that you cannot restore a deleted section.

- ► To delete a section:
- **1** In the Section pane, select the section label.
- 2 Select Edit > Delete Section.

You can also select the section and click Delete Section on the shortcut menu.

The Remove Section dialog box is displayed.

3 Click Remove.

Setting Options

The Tools menu provides commands that allow you to manage various Interactive Reporting Studio options, such as job processing options, connections, default formats, and program options.

Review the following sections for information on:

- Specifying Default Formats
- Selecting Program Options

Note: See "Tools Menu Command Reference" on page 93 for an overview of all the Tools menu commands.

Specifying Default Formats

This section explains how to set up and change the way fonts, styles, numbers, currency values, and dates are displayed within the workspace. Default formats that you can set include:

- Default Fonts and Styles
- Default Number Formats

Default Fonts and Styles

You can control the way fonts are displayed in every section (except Results and Dashboard) by applying default font and formatting styles to each section element.

> To change default fonts and text settings:

1 Select Tools > Options > Default Formats.

The Default Fonts And Styles dialog box is displayed, with tabs that contain specific font settings for each section.

Default Fonts and Styles		
Default Fonts and Styles Query Table Pivot OL Settings: Table Catalog Pivot Pivot OL Topic Structure View Request Line Items Finite Items Pivinary Key Pirimary Key Foreign Key Indexed Column Topic Name	AP Report Chart Numbers Font Name: Arial T Size: 3 T B I U	OK Cancel Defaults Help
Query Log Custom SQL Log Sample Sample Text		

2 Click the tab for the section whose fonts and styles you want to change.

3 Make the desired default font, size, style, and alignment settings and click **OK**.

To restore the default settings, click Defaults.

Default Number Formats

You can change the way numbers, currency values, and dates are displayed through out the Interactive Reporting Studio, or you can create new custom formats. Use the Numbers tab of the Default Fonts and Styles dialog box to specify default settings for number formats.

Table 10 lists the fields available on the Numbers tab.

Field	Description
Select a Formatting Locale	Sets the locale or country associated with the default format that you want to use. The locale that you select determines the available number, date, and currency formats.
Date	Sets the default date format.
Timestamp	Sets the default time and date format.
Time	Sets the default time format.
Month (For "Add Date Groups")	Sets the default month format for the month used in Add Date Groups.
Null	Sets the default format for null values.
	Null values are empty values for which no data exists. Null values are not equal to zero.
Real	Sets the default format for real values.
Integer	Sets the default format for integer values.

 Table 10
 Default Number Formats

Table 11 lists the numeric field options and definitions along with examples.

Option	Description	Examples
0	Integer placeholder or zero value. If a number has an integer value in this position relative to the decimal point, the integer is displayed. Otherwise, a zero is displayed.	Apply 0 to show 123 Apply 0.00 to show 123.45
#	Integer placeholder. If a number has an integer value in this position relative to the decimal point, the integer is displayed. Otherwise, nothing is displayed.	Apply #,##0 to show 1,234
()	Formats with parenthesis options display negative values in parentheses. Otherwise, negative values display with a minus sign.	Apply (#,##0) to show (1,234)
;	A semicolon operates as a separator between two number formats. The semicolon separates a positive integer and a negative integer.	Apply #, ##0;(#,##0) to show 1, 234 or apply (1, 234) for a negative number

 Table 11
 Number Field Descriptions (Continued)

Option	Description	Examples
\$ %	Adds the respective character to numeric values in the same position relative to the decimal point.	Apply \$#,##0.00 to show \$1,234.56 Apply 0% to show 3%
m d yy	Displays month, day, and year in respective positions for date-coded information.	Apply mm dd yy to show 05 07 99
- /	Adds the respective character to date-coded values in the same position relative to variables.	Apply mm/dd/yy to show 06/23/99
HH MM SS	Displays hour, minute, and second in respective positions for date-coded information.	Apply HH:MM:SS to show 17:45:10
: AM PM	Adds the respective character to time-coded values in the same position relative to variables.	Apply HH:MM:SS AM to show 17:45:10 AM

- To change a default number format:
- **1** Select Tools > Options > Default Formats.
- 2 Select the Numbers tab.
- 3 Make the desired selections and click OK.

Selecting Program Options

This section explains how to set up default file locations, enable and disable specific operating functions, administer document features, define drill-down paths, and set OLAP options. Review the following sections for information on:

- General Options
- File Locations
- OLAP Options

General Options

Use the General tab to globally enable or disable specific operating functions.

- To select general options:
- **1** Select Tools > Options > Program Options.
- 2 Select the **General** tab in the Options dialog box.
- **3** Define the desired options and click OK.
 - **Reset Print Properties** Retains the print settings with each section of the document, instead of inheriting the current default print settings.

- Use Microsoft Windows Colors and Styles Specifies the colors and styles scheme used in the Interactive Reporting Studio, or the Interactive Reporting Web Client. The selection in this field updates s a UseSystemColors registry key setting in either the Designer or plug in folder of the registry. Enable this field o use legacy colors and styles based on Microsoft Windows. Disable this field to use Workspace colors and styles. Before the setting in this field can take effect, the application must be closed and restarted.
- Compress All Documents Specifies that the Interactive Reporting Studio save all documents in compressed file format. This reverses the default setting, which saves documents without compression. If enabled, you can override this privilege and save documents without compression by choosing File > Save As and changing the Save As type.
- Create New Documents Compressed Specifies that the Interactive Reporting Studio only compress new documents.
- Always Prompt For Owner Name Requires Interactive Reporting Studio to prompt for an owner name of job scheduling repository tables whenever you schedule a document. Enable this feature if you schedule documents to more than one repository.
- When A Two Digit Year Is Entered, Interpret As A Year Between By default, if you enter a date and type only two digits for the year, the Interactive Reporting Studio handles the dates as follows:
 - Two-digit years entered from 00 up to and including 29 are assigned to the 21st century (2000 to 2029). For example, if you enter 3/12/18, the Interactive Reporting Studio accepts the date as March 12, 2018.
 - Two-digit years entered from 30 up to and including 99 are assigned to the 20th century (1930 to 1999). For example, if you enter 3/12/96, the Interactive Reporting Studio accepts the date as March 12, 1996.

You can change the default century to which a two-digit year is assigned by using the datehandling boxes. These boxes require a range of dates within a 99-year time period. Changes to a date format are applied globally, but do not affect dates previously formatted.

For example, if you want the two-digit year 25 to be assigned to the twentieth century instead of the twenty-first century use the arrow keys to scroll to the year *19*99. The date in the corresponding read-only date-handling box is automatically changed to 1900.

Tip: Whenever possible, enter the year as four digits; that is, type 2001 instead of 01.

File Locations

Use the File Locations tab to specify the default locations for Interactive Reporting Studio documents and other necessary files.

- > To specify default file locations:
- **1** Select Tools > Options > Program Options.

The Options dialog box is displayed.

2 Select the File Locations tab.

- 3 Enter the desired options and click **OK**.
 - Documents Directory The default directory where you want to save Interactive Reporting Studio documents when the Save File dialog box is displayed. Interactive Reporting Studio documents are saved in the default directory with a .bqy extension.
 - Connections Directory A directory that contains the Interactive Reporting connection files used to connect to databases. The default Connections directory is C:\hyperion\BIPlus\data\Open Catalog Extensions.
 - **Default Connection** The Interactive Reporting connection used when no connection is specified, such as when you click the connection icon in a new document file.
 - **Preferred Repository Connection** The repository connection file you want the user to see in the Open Repository Connection drop-down list.
 - HTML Template Directory The directory of HTML templates used with the HTML Export Wizard.

OLAP Options

Use the OLAP tab to haveInteractive Reporting automatically create a Results section when you click **Process** to process your OLAPQuery

- To select OLAP options:
- **1** Select Tools > Options > Program Options.

The Options dialog box is displayed.

- 2 Click the OLAP tab.
- **3** Define the desired options and click **OK**.

Auto-Generate Results Section When Processing an OLAP Query – Interactive Reporting automatically creates a Results section for any future OLAPQuery section when that OLAPQuery section is first processed.

Document Properties

The Document Properties tabs are used to describe information about an Interactive Reporting document, such as in which application the document was created and also the server address associated with a document. Document properties consist of the General tab and the Server tab.

General Document Properties

Use the General Document Properties dialog to display the creation and modification attributes of a document.

The following Created display fields for the document are shown on the General Document Properties dialog:

- **Document Path** Displays the path name to which the document has been saved.
- **Application Name** Displays the application name in which the document was created. Application names include:
 - Interactive Reporting Studio
 - o Interactive Reporting Web Client
 - Hyperion System 9 BI + Workspace
 - o Interactive Reporting Job
 - o SmartView
- **Application Version** Displays the release version in which the document was created.
- **Date** Displays the date on which the document was created.

The following Modified display fields for the document are shown on the General Document Properties dialog:

- **Document Path** Displays the path name to which the modified document was saved.
- **Application Name** Displays the application name in which the document was last modified. Application names include:
 - o Interactive Reporting Studio
 - o Interactive Reporting Web Client
 - Hyperion System 9 BI + Workspace
 - Interactive Reporting Job
 - o SmartView
- Application Version Displays the release version in which the document was last modified.
- **Date** Displays the date on which the document was last modified.

Server Document Properties

Use the Server Document Properties dialog to specify the server address associated with an Interactive Reporting document (.bqy).

To specify a server address for a document, enter the URL address in the Server Address field and click **OK**.

The URL address must be in the following format: http://<server>/workspace/dataaccess.

Customizing Menus

You can use JavaScript to customize Interactive Reporting Studio menus. Add scripted menu items to the menu bar to:

- Run commonly used scripts
- Launch separate applications
- Export sections to a different file format with a single click
- **Note:** Since version 6.0, JavaScript is used as the script-editing tool instead of the Interactive Reporting scripting language. Script written prior to version 6.0 is still recognized, but is enclosed in a wrapper and called with a JavaScript command.
 - To add a custom menu:

1 Select **Tools > Customize**.

The Customize dialog box is displayed.

Customize	
Menu Name	
&Custom	Add
Menu Items	Edit
<u>_</u>	Remove
	Separator
	▲ ▼
OK Cancel Help	J

- 2 Type the name of your custom menu in the Menu Name field.
- 3 Click Add to add a new menu item to the Menu Items list.

The Properties dialog box is displayed and shows the Text page.

Properties	
🗖 Query 🗖 Results 🗖 Pivot	
🗖 Chart 🗖 Report 🗖 EIS	
	l
	l
	l
UK Cancel Help	

4 Type a **name** for the menu item.

Select one or more check boxes to indicate the sections where the menu item should be displayed.

5 Click the **Script** tab to display the Script page.

Enter script commands to be run when the item is chosen from the menu. If you plan to deploy the menu item to a group of users, make sure that commands which reference external applications or files use universal paths.

6 Click **OK** to return to the Customize dialog box.

Add separator lines and move menu items as needed to complete the final menu.

- 7 Click **OK** when you are finished to close the Customize dialog box.
- 8 To provide the same functionality for distributed users, copy the preference file which supports this feature to users' machines.
 - For Windows, the bqtools.ini file is located in the Windows directory.
 - For UNIX, the .bqtools.ini file is located in the user home directory

Tools Menu Command Reference

Table 12 provides a quick reference to the commands available on the Tools menu and list any related shortcuts. The commands listed are those available in the query and results sections.

Command	Description	Keyboard Shortcut
Process Query	Processes the current query, all queries in the document, or a customized selection of queries.	
Connection	Allows you to select, log on, log out, modify, or create a connection.	
Connections Manager	Opens the Connections Manager dialog box.	[F11]
Save Connection	Saves a connection file with a Interactive Reporting Studio document.	

Table 12 Tools Menu Commands

Table 12	Tools Menu	Commands	(Continued)	(Continued)
----------	------------	----------	-------------	-------------

Command	Description	Keyboard Shortcut
Administer Repository	Opens the Administer Repository dialog box, where you can modify object descriptions or groups.	
View Job List	Shows run time and status details for each job submitted to a job repository.	
Options	Allows you to set default formats or program options.	
Customize	Opens the Customize dialog box where you can add customized menus or menu items.	

Interactive Reporting Web Client Co-existence

Interactive Reporting Web Client users can now work with different versions of the Interactive Reporting on the same client machine. To enable this feature, new Interactive Reporting Web Client installations include multiple versions of the application. Users can work with all servers older than Release 8.3 + server through Release 8.5. The feature was added because the Interactive Reporting installation does not have a backward compatibility mechanism and only supports the8.5 version of the Interactive Reporting Web Client. The correct version that gets used is determined at run-time when the BQY is opened. For example, if a user opens n Interactive Reporting document (.bqy) in a Release 8.3 server, the 8.3 version of the Interactive Reporting Web Client is launched to serve the request.

Two Interactive Reporting Web Client versions are installed in this release: an 8.3 version and the current 8.5 release. Each Interactive Reporting Web Client plug-in entity is separated on the disk and in the registry. Each version is installed in a separate folder under <IE Install Directory>\PLUGINS\Hyperion\<Insight Series Version> or under <Firefox Install Directory>\Plugins\Hyperion\<Insight Series Version>. A mechanism has been implemented to find and start a particular Interactive Reporting Web Client executable based on its version. The Interactive Reporting Web Client Launcher that provides 6x/8x co-existence has been enhanced for 8.3/8.5 and future co-existence. There is a single launcher in the system associated with the BQY mime-type and file extension.

Locally Saved BQYs

If a user is working with a locally saved Interactive Reporting document (.bqy) and saves it to disk from the Interactive Reporting Web Client, the file is appended with version information. If the user opens the file again, the appropriate version of the Interactive Reporting Web Client is launched. After this, the Interactive Reporting Web Client checks the server version to connect to the server. If the server is newer than the current Interactive Reporting Web Client, the user is prompted to upgrade to the current version. In this case, the older version accesses

the URL of the server, launches the zero administration page, and performs the upgrade. The Interactive Reporting Web Client is opened from the disk using file name in the upgraded version.

If a BQY was created in previous versions of the Interactive Reporting Web Client or in the Designer, the user is prompted for the version to use. The user can select from a list of versions of the Interactive Reporting Web Client, including the 6.x version. If the user selects a 6.x version, then no version record is attached to the document because there is no command on the Tools menu to open the document in another version of the Interactive Reporting Web Client.

After the Interactive Reporting Web Client is launched, the user can change the Interactive Reporting Web Client designated to open the document at any time by selecting the following options from the Tools menu:

- Change Web Client version to open this file
- Make this Web Client default for locally saved BQY files.

Interactive Reporting Web Client Mismatches

If there is a mismatch between the Interactive Reporting Web Client series required by the server and the user selects a compatible version, a file version record is saved and the document is reloaded.

Required Interactive Reporting Web Client series information is only provided by the 8.5 version servers. If this information is unavailable, it is assumed that the user is connected to either an 8.3 or 6.x server (which is detected by specific cookies and the absence of later version server cookies). Otherwise, it is assumed the user is connected to an 8.3 server. If the user opens a document in the Interactive Reporting Web Client in a greater version number than is required by the server (for example, a document from a 6.x server is opened in an 8.3 server), the Save on Exit prompt is suppressed when the document is reloaded to prevent the user from upgrading accidentally to the newer series.

Querying Relational Databases

This chapter explains how to use Interactive Reporting Studio to connect to and query a relational database.

In This Chapter	Query Section	98
	Building Queries	98
	Processing Queries	100
	Saving Queries	102
	Cancelling Queries.	102
	Building Subqueries	103
	Derived Tables	106
	Working with Query Section Data	111

Query Section

The Query section is the foundation of any Interactive Reporting Studio document and is automatically created (along with its complementary Results section) whenever you open a new document. It provides an intuitive interface to a relational database server*database server*.

The Query section is the Interactive Reporting interface with databases. In the Query Contents pane, you can examine the overall informational contents of the database, and even look underneath to verify the actual data values.

The Query section contains a Data Model possibly a query. Some Data Models are locked, and only allow you to process the existing query; others allow you to make modifications to the query or build a new one. You can also apply filters, compute and modify items, and process your request for information.

An Interactive Reporting document can contain multiple Query sections. These query sections can access a wide range of data sources (relational databases, OLAP servers, imported data sets, and local joins). Each Query section has its own Results section and can be associated with the same database or different databases (that is, the connection file or data model used is independently defined in each query).

Using Data Models in the Query Section

Relational queries use data models to view the server database tables and create queries. Depending on your access privileges, you can download a prebuilt data model to the Query section, or build a data model and create your own query.

When you connect to a database, the tables in the database are displayed in the Catalog pane of the Query section. A data model is a visual representation of these tables. You use a data model to create queries that specify which data to fetch from the database and retrieve for analysis.

To view the database tables in the Table catalog, select DataModel > Table Catalog.

The Table catalog in the Catalog pane expands to show all of the tables in the database. If you are not connected to a database, Interactive Reporting Studio prompt you for your user name and password.

To create a data model, in the Catalog pane, select a table from the Table catalog, and select Query > Add Request Item(s).

Building Queries

You build queries by adding topics from the data model in the Content pane to the Request line.

- To build a relational query:
- 1 Click the **Request** button on the Section title bar to display the Request line.
- 2 Complete one of the following actions:

- Drag an item from the Content pane to the Request line.
- Select an item in the Content pane and select Query > Add Request Item(s).

To add an entire column from a table to the Request line, select the table header. You also can select more than one of the same item (to create duplicate items).

If you add more items than the Request line can display, use the arrow buttons at the right of the Request line to scroll through the requested items, or resize the Request line to display multiple rows of request columns.

Working with Items on the Request Line

As you build your query, you can reorder, remove, or hide items on the Request line. This allows you to change the way in which the query processes and displays.

Reordering Request Items

You can move Request items to reorder them for viewing results.

To reorder items on the Request line, select the item to be moved and drag it to a new location on the Request line.

Removing Request Items

You can remove items from the Request line to exclude the data from your query or Results set.

- To remove an item from the Request line, select the desired item and complete one of the following actions:
 - Click the Delete button on the standard toolbar.
 - Click **Remove** on the shortcut menu.
 - Press the Delete key.

If you have not yet processed the query, Interactive Reporting Studio removes the item from the Request line. If you have previously processed the query, you are informed that your results no longer match the items requested.

Caution! Remove items with caution as a computed item or report may draw data from the item you delete.

Hiding Request Items

You can hide items that are displayed on the Request line. This allows you to incorporate data in the results set without displaying it. Hidden request items cannot be referenced for computations.

- > To hide a request item, complete one of the following actions:
 - Select the item and click Hide on the shortcut menu.
 - Select the item and choose View > Hide Request Items.
- To show a hidden request item:
- **1** Complete one of the following actions:
 - Click in the Request line and click Unhide on the shortcut menu.
 - Select View > Unhide Request Items.

The Unhide Columns dialog box is displayed.

2 Select the items you want to view and click **OK**.

When you have identified the items you want to include in your query, you can perform a number of other operations before processing the query. You can add filters or computed items to the Request line, or you can use a Request line item to specify a sort order.

For more information about filters, see Chapter 11, "Using Filters." For more information about computed items, see Chapter 12, "Working with Computed Items." For more information about sorts, see Chapter 13, "Applying Sorts."

Processing Queries

After you build your query and apply filters, computations, sorts, and any other adjustments to further refine your request, you need to process it. Processing your query may take a few moments if your query is complex or if the data in linked report sections needs to be refreshed.

When you process your query, the data is retrieved to the Results section in tabular form. You can reprocess your query at any time and in any section to refresh the data. You can also return to the Query section from any other section at any time to alter the query and reprocess it.

- > To process a query, use one of the following options:
 - Click the Process button on the standard toolbar. (Click the right-arrow to select a process option.)
 - Select Tools > Process Query and select the desired process option.

Since a document can contain multiple queries, there are three processing options on the Process drop-down list:

- **Process Current** Processes the current object. In some cases more than one query may be processed, for example, if a report references results sets from multiple queries. Process Current is the default selection when using the toolbar button.
- **Process All** Processes all the queries in the document. By default, queries are processed in the order in which they are displayed in the Section catalog. For example, in a document with three queries, Query1, Query2, and Query3, the queries are executed in that order when you select Process All. ("Query Processing Order" on page 101).
- **Process Custom** Opens the Process Custom dialog box so that you can indicate which queries to process by selecting a query's check box.

The query is sent to the database and retrieved data is displayed in the Results section. While the data is being retrieved, the Status bar displays a dynamic row count indicating rate and progress of server data processing and network transfer.

Query Processing Order

If there are multiple queries in a document, you can determine the order in which they get processed, and select which queries to include or exclude from the processing stream.

This feature is particularly useful when you need to use "fresh" data during a local join operation, or you want to ensure that your local results are populated with current data from a source query not affected by several earlier tables without duplicating the processing of some queries.

Query Processing Order is available for the Query section, OLAPQuery section and any imported data files sections that can be processed in an Interactive Reporting document (.bqy) file when you use the "Process All" feature.

- **Note:** Query processing order settings are saved with the document. For a temporary processing order, do not save the document or set the processing order to what is normal for the given document.
 - > To specify a query processing order:
 - 1 Select Tools > Process Query > Processing Order.

The Query Processing Order dialog box is displayed.

The first time the Query Processing dialog box is displayed after the new document is created, all query sections belonging to the document are displayed in the order which they are displayed in the Section Catalog (top down). Query sections added after the existing sections have been arranged, are displayed at the end of the list in the order in which they were added when the dialog is reopened.

Query Processing Order Select a query and use the Up/Down buttons to arrange the query processing order	
Select a query and use the Up/Down buttons to arrange the query processing order	
* Query Query2 * ImportedFile.csv * Duery3 Up Down	
Double-click on an item to add/remove it from the processing order. (Only queries with an * are processed during Process All actions.)	
OK Cancel Help	

- 2 Select a query section and then move it up or down in the processing order using the arrow keys to the right.
- **3** Double click a query section to remove it from the processing order or add it back and click **OK**..

Only queries marked with an asterisk (*) are processed during "Process All".

4 Select Tools > Process Query > All.

Saving Queries

After you process a query, your data is available until you close the document. Saving your document saves the current formatting and layout of all sections in the document.

To save your query, select File > Save.

For a complete discussion of save options, see "Saving Documents" on page 45 and "Saving Results Sets" on page 129.

Cancelling Queries

To cancel a query, both Interactive Reporting and the database must communicate properly for reasons other than running a query.

> To cancel a query, press and hold [Alt+End] until the query is cancelled.

The database connection used to connect to the database must use an Asynchronous API. This is the default configuration, but it can be turned off on some databases. If the Asynchronous API s disabled, the database cannot detect a new request until the query has finished processing.

The database must pause every so often even when processing a query. As a default configuration, many databases are not set up to pause, which could result in the database being too busy to hear the cancel request until the query has finished processing.

Building Subqueries

You can use subqueries to filter your data. A subquery answers a specific question or provides specific information within the context of a main query, also called a "parent" statement. The database evaluates the entire query by first analyzing the subquery. The parent statement then filters its rows based on the rows retrieved by the subquery. Interactive Reporting Studio uses two types of subqueries: regular and correlated.

Review the following topics for information on:

Regular Subqueries

Correlated Subqueries

Regular Subqueries

A regular subquery executes the inner and outer queries once and returns the values from the inner query to the outer query. For example, you might need to find out who sold more than the average of all sales representatives in April. You first use a subquery to define what was the average sales amount in April. This information is supplied to the parent query, which then determines which representatives exceeded the average of all sales in April.

- To build a regular subquery:
- 1 Select an existing parent query or build a parent query.
- 2 Use one of the following options to select an item on which you want to set a filter:
 - Double-click the topic item
 - Drag the item from the Content pane to the filter line
 - Right-click and item and select filter on the shortcut menu.

The filter dialog box is displayed.

3 Click Advanced.

The Advanced button toggles the Create Subquery button.

Limit:					
Name:			OK		
	🗖 Include Nulls		Cancel		
🗖 Not	= Equal	•	Ignore		
\times \checkmark			Help		
Show Values		A			
Custom Values					
Custom SQL					
Select All		-	· · · · · · · · · · · · · · · · · · ·		
Hemove	I		Advanced		
EL oaded Values !	Settings				
C Load From F	-ile	Cha	nge File		
Control Load From E)atabase				
Create SubQuer	y				Click have to greate a
Create a subque current guery as	ary with the its parent.	Create Su	ibQuery		subquery.
	•				

4 Click Create Subquery.

A subquery indicator shows that you are working with a subquery and not the parent query, even though the parent query's data models are displayed in the Content pane (which now has a gray background).

The Subquery section is considered a child of the parent query and is subordinate to the parent query. That is, it is dependent on the parent query and does not have its own default Query or Results sections. Subqueries can be nested within other subqueries, in which case the first subquery becomes the parent query to the subquery nested inside it.

5 Build the query.

Only one item can be on the Request line, but you can add server and local filters, set a filter inside a subquery as a variable, and define data functions and computations as needed.

6 In the Section pane, click the parent query for the subquery.

The parent query section is redisplayed.

7 Click Process to process the parent query and subquery.

Correlated Subqueries

A correlated subquery is related to a regular subquery in that it uses an inner query to feed result values to the outer query. A correlated subquery executes the outer query multiple times, once for each row returned by the inner query; it is processed by joining a column in the subquery to a column in the parent query.

For example, suppose you had to identify which sales representatives had more sales in the current month than they did in the previous month. The correlated subquery is executed for each row of sales information in the parent query to first determine what were the sales for each representative in the previous month. This data, in turn, is compared to sales for each representative in the current month, and only those representatives whose sales in the current month were greater that their previous month's sales are returned.

- > To build a correlated subquery:
- **1** Select an existing parent query or build a parent query.
- 2 Use one of the following options to select an item on which you want to set a filter:
 - Double-click the topic item
 - Drag the item from the Content pane to the Filter line
 - Right-click and item and select Filter on the shortcut menu.

The Filter dialog box is displayed.

3 Click Advanced.

The Advanced button toggles the Create Subquery button.

4 Click Create Subquery.

A subquery indicator ($\overline{\mathbf{m}}$) shows that you are working with a subquery and not the parent query, even though the parent query's data models are displayed in the Content pane (which now has a gray background).

The Subquery section is considered a "child" of the parent query and is subordinate to the parent query. That is, it is dependent on the parent query and does not include its own default query or results section.

Subqueries can be nested within other subqueries, in which case, the first subquery becomes the parent query to the subquery nested inside it.

A subquery is correlated based on a join from a column in the subquery to a column in the parent query.

5 Build the query.

Only one item can be on the Request line, but you can add server and local filters, set a filter inside a subquery as a variable, and define data functions and computations as needed.

- 6 If the parent query you want to work with is not displayed in the Catalog pane, select **Show Queries** from the shortcut menu in the Section pane.
- 7 Drag the parent query into the Content pane.

The parent query is displayed as a blank topic in the Content pane.

8 Drag the topic item you want to correlate by into the blank parent query topic.

The Select Correlation Column dialog box is displayed and shows the tables of the parent query.

- 9 Use the + and signs to navigate through the structure of the directory tree.
- **10** Select the column in the parent to which you want to join the subquery topic item and click **OK**.

The topic item is added to the Filter line and a join line is drawn.

11 In the Section pane, click the parent query section.

The parent query section is redisplayed.

12 Click Process to process the entire query.

The topic item added in the subquery shows the label *sub* next to the topic item name.

Derived Tables

A "derived table" is essentially a statement-local temporary table created by means of a subquery in the FROM clause of a SQL SELECT statement. It exists only in memory and behaves like a standard view or table.

For example, assume an Oracle 9i database table is called "state_table" and has the following row values in the "state" column.

State	
CA	
CA	
CA	
FL	

If you used the following inner SELECT statement which includes a derived table to evaluate the "state_table", you could return the count and percentage of each state. The SQL has been written for Oracle 9i.

```
select state, count(state) as State_Count,
(count(state)/derived_table.tot_state_count) as State_Percentage
from state_table,
(select count(state) tot_state_count from state) derived_table
group by state, derived_table.tot_state_count;
```

The results of the query is displayed below:

State	State_Count	State_Percentage
СА	3	0.75
FL	1	0.25

Derived tables are useful when you need to generate aggregates in a table that would otherwise contain dimension type data and join the resulting aggregate with detail level facts in another table.

Additionally, the aggregate values in the derived tables can be used in the outer query's WHERE clause (i.e., "where salary > average_salary", average_salary has been derived by the subquery). These tables can also optimize a query's performance in some circumstances, such as minimizing sorting when some DISTINCT values are needed from some tables but not all. Finally, it might eliminate the need in some cases, to build "local results" queries. This feature enables users to access this type of SQL construct in an easy-to-build manner.

Derived Tables Rules and Behavior

The rules and behavior of a derived table include:

- A Derived Table cannot be "iconized".
- A Derived Table only has two speed menu options: "Add Selected Items" and "Remove".
- When you modify a Query section that is used as a Derived Table, note the following:
 - Derived tables are available for connections to Teradata, DB2, Oracle, and Microsoft SQL Server. If you use the feature when you build a query, an active connection is required if the server code is ODBC or OLEDB so that the "smart ODBC" can determine the database type.
 - Interactive Reporting Studio Designer, Explorer versions and the Interactive Reporting Web Client all can use this feature. Scripts in documents opened with the Hyperion System 9 BI + Workspace can also use this feature.
 - If you remove all items from the Request line, the topic are displayed empty (there are no columns) in the section(s) where they are used as a derived query.
 - If the columns referenced by the "deriving" Query section are removed from the Request line, the effect is the same as if a "sync with database" had found the column removed.
 - If you rename items in a Query section used as a derived table, you must ensure that the new name is a valid name for the database in use. These names are used during the SQL generation, and are quoted if "quoted identifiers" are selected in the database connection.
 - o Items from "Local Results" collection cannot be added to the Query.
 - \circ ~ The icon used for a derived table in the Table Catalog tree is shown on the left.
 - Both the referencing query and the derived table query must use the same data source even if one or the other is not connected. For example, assume Query1 is built from DataModel1, and Query2 is a query section not derived from a DataModel section. That is, you must be sure that both Query1 and Query2 can connect to the same data source name (such as an ODBC data source name or an Oracle TNS name), and they share the same API and server codes. The data source name alone may not be sufficient since two names can be identical over two different APIs. However, it is not required that the connections use the same database credentials. You do have to ensure that the referenced tables are accessible from whatever logon is used in all selected Query sections.
 - Interactive Reporting Web Client users need at least a Query role to use derived tables in an existing data model, or data model role to add or remove them from the data model.
- > To build a query that uses a "derived table"
- **1** Build the query which will use the "derived table" and process the query.
- 2 Insert a new query by choosing **New Query** on the Insert menu.

A derivable query can be built for a relational database.

3 Build the "Derived Table" query by clicking anywhere in the Catalog pane and selecting **Derivable Queries** on the shortcut menu.

The "Derivable Queries" option is not displayed if no existing query sections can be used in the current query.

(The Table Catalog below shows the Derivable Queries tree expanded and the Tables tree contracted. Local Results only are displayed when a user has requested them from the speed menu.)



The only queries that are displayed in this list are those that:

- have the same connection information as the current Query section
- have at least one item on the Request line
- do not use local joins
- do not also contain derived table topics (that is, derived tables are not nestable)

The topic name, when a derived table is added to the workspace, is the same as the Query section name, and the column names are the same as the names of the items on the Request line in the Query section being added with the exception of any (data) function component. In cases where the function is displayed on the Request line, then the resulting topic item name would be the same as the name that is displayed in Results after processing the query. The following diagram illustrates how the Request line for Query2 are displayed as a topic in another Query section:

4 Create a manual join by dragging an item from at least one topic to another, including to/from the added derivable query sections.

Once the derived table becomes a topic, items from it can be added to the Request, Filter, or Sort lines of the containing query. It can be referenced in computed item dialogs, and can be used in custom GROUP BY logic.

If you use the Show Values feature when setting a filter on an item in the "derived table", the SQL that would be used is the same as if you set a filter on the same column in the source Query section.

5 Process the query using the derived table.

You can process the query by way of "Process" command when viewing the query section (or one of its dependent sections) or by checking it in the Process Custom dialog.

The "Process All" command process the query containing the derived table. It also processes the query section from which the derived table is derived, unless it explicitly removed using the Query Processing Order dialog (which is recommended to avoid duplication of processing at the database.)

Additions to the Request line of the source Query are reflected in the topic item list of the referencing Query section the next time it is displayed.

Once a Query section has been added to another Query or Data Model workspace, changes to the source query section will check for dependencies and warn you of any discrepancies that might occur. For example, you would receive a warning if you removed an item from the source query's Request line which is used someplace else (Request, Filter, Sort, etc.) in a Query that is deriving a table from the source query.

Derived Tables and SQL

Review the following sections for information on:

- Custom SQL
- Processing a Query that Contains another Query Section

Custom SQL

If t you open the Custom SQL window in a query that has a derived table topic, then the SQL for the derived table is shown in the custom SQL window as part of the overall query. At this point, the SQL is locked; changes in the source query section for the derived table are not reflected in this SQL until such time as the user presses the Reset button. Further more, when you process the query with the Custom SQL window open, it is executed "as is".

Processing a Query that Contains another Query Section

When you process a Query section that contains another Query section, it forms SQL that uses what are called "derived tables", which are essentially subquery statements in a FROM clause. The SQL generation phase behaves as follows:

- Generates the SQL for the Query section being used as a topic in the current Query section being processed, including resolving any variable filters. Any Sort line items in the source Query section are ignored in generating the SQL. This generation only occurs if the contents of the Query as a topic is referenced by the currently processing Query section, following the rules for join path generation specified for the currently processing section.
- The resulting SQL is added to the currently processing Query section if required, in the FROM clause, surrounded by parentheses, as with any subquery.

- The entire subquery is given a table alias. The alias will follow the pattern for any table involved in a query and will thus be of the form "ALn". As with subquery filters, any aliases used in the subquery itself will be distinct from the alias names used in the currently processing Query section's table references. The alias will be used thereafter to refer to columns in the "derived table", as is normal for SQL generation based on physical tables.
- Following the table alias name, "ALn", a parenthesized list of column aliases is added. This list is identical to the list of names of the topic items in the Query topic. (For Oracle, the column aliases are displayed in the select list of the subquery instead of in a separate parenthesized list, for example, SELECT AL1.STORE AS store....)
- Before submitting the resulting SQL to the database, a check is performed to ensure that the query is properly joined. If it is not, the standard behavior, as specified by the database connection preference "Allow Non-Joined Queries", is taken.
- Following submission of the SQL, any references to the column alias names on the Request line of the currently processing Query section are available to build the dependent Results section.
- The minimum requirement is that if the "derived table" is referenced by the currently processing query, then its SQL is generated as if that section were processed alone, with the exceptions noted above. At initial implementation or in a future release, the SQL would include only those columns actually needed by reference in the currently processing Query. This would reduce the number of items in the SELECT list of the subquery, with a corresponding reduction in the number of column alias names.

For example, assume the following Query section is to be used as the source for a "derived table" in another Query section:

Sections	x	Request Store Id City State UCASE(Computed)
Query	<u>^</u>	Limit 🔺 Store Type
Results		Locked Data Model : DataModel
Query2		🛄 Pow Customers 🔽
Results2	L	Store Id Store
🗓 🛛 DataModel	L	Buyer = Store Type
	L	Street
		State
		∠ip Phone

In this example, the computed item's definition is

"UCASE (Pcw_Customers.Buyer)". The filter on 'Store Type' is set to 'Discount'.

The Query section using "Query" as a derived table might appear as follows:

In the previous diagram, the filter on City is set to 'Los Angeles'. The SQL generated when the above query is processed, per the minimum requirements, would be:
```
SELECT AL2.STATE, SUM(AL1.UNITS) FROM PCW_SALES AL1,
(SELECT AL3.STORE_ID, AL3.CITY, AL3.STATE, UCASE(AL3.BUYER)
FROM PCW_CUSTOMERS AL3
WHERE AL3.STORE_TYPE = `Discount') AL2(STORE_ID, CITY, STATE, COMPUTED)
WHERE AL2.STORE_ID = AL1.STORE_ID AND AL2.CITY = `Los Angeles'
GROUP BY AL2.STATE
```

Working with Query Section Data

You can further enhance the data requested in the Query section by working with the data. Working with Query Section Data involves:

- Processing Results to a Database Table
- Estimating Query Size
- Displaying Database Remarks
- Preaggregating Data Using Functions
- Appending Queries
- Using Local Results
- Using Stored Procedures
- Setting Query Options

You can accomplish these tasks by using the commands on the Query menu.

Note: See "Query Menu Command Reference" on page 122 for a complete list of the commands available on the Query menu.

Processing Results to a Database Table

Instead of retrieving data to the Results section, you can instruct an Interactive Reporting Studio to create a table in the database to store your results set. Items on the Request line become the column headings of the new table, and you can append new columns to the table and query it as needed.

- **Note:** The connection file and database to which you are connected determine whether you can use this feature. You must have *Create* and *Insert* privileges on the database to process results to a database table.
 - To process results to a database table:
 - **1** Select Query > Process Results To Table.

The Process Results To Database Table dialog box is displayed.

- 2 Specify the information requested.
 - Table Name Name of the new table you want to create or the name of an existing table to which you want to append columns.

You can create tables under your own owner name or under different databases or owners. If you do not have the correct privileges or do not specify an alternate location, the table is created under your own owner name. Use the format *DATABASE.OWNER.TABLENAME* to specify alternate names.

- Create Table Creates a new table in which Request items form columns.
- Append To Table Appends Request items as new columns in an existing table.
- Grant Access To Enables either everyone or specific users to access the new tables. Type PUBLIC or specific user IDs (separated by commas) for each user who should have access to the table. If not selected, access to the table is filtered to your own user ID.

3 Click OK.

The table is created or modified under the specified database and owner name.

- > To verify that the query was processed and the results saved as a database table:
- **1** Select **DataModel > Table Catalog**.

[F9]

The Table Catalog expands in the Catalog pane.

2 Select Refresh on the shortcut menu.

The table that you created is displayed in the list of database tables.

Interactive Reporting Studio tracks tables that you have created under your database user name and stores a list of these tables in the bqtbls5.ini file.

- > To delete tables you created using the Process To Database Table feature:
- **1** Select Query > Process Results To Table.

The Process Results to Database Table dialog box is displayed. Tables created under your user ID are displayed in the Tables Created By list.

2 Select a table from the list and click **Delete**.

Estimating Query Size

Queries that sift through and retrieve enormous amounts of data can take a long time to process, and may consume unnecessary system and server resources. If you suspect these factors exist, you may want to size your query before you process it.

The Estimate Size feature queries the database to see how many records your query will retrieve. You can use this feature to test a questionable query or to decide whether to prevent or postpone processing a large results set.

> To estimate the size of a query, select **Query > Estimate Query Size**.

queries the database and counts the number of records to retrieve if the query is processed. This process may take a while for server-intensive queries.

Displaying Database Remarks

Database remarks provide detailed contextual information about a table or column. Remarks may describe the origin, derivation, or details about data model topics and items, which can help you identify and select the information you need. Database remarks often exist as metadata when you map data in a data warehouse project or if you use a CASE tool to manage your database.

- > To display database remarks, use one of the following options:
 - Select **Query >** Show Remarks.
 - Click Show Remarks on the shortcut menu.

Preaggregating Data Using Functions

Depending on how you plan to view your data, you can select to preaggregate data at the database server. Preaggregation (also called server aggregation) is a querying strategy that uses functions to summarize data as it is retrieved from the database. Instead of returning a lineitem list of every row that meets the criteria on your Request line, you can order the database to group related information. This results in one row representing the combined (aggregate) value of each distinct group.

You use data functions (provided by your RDBMS) to preaggregate data in a query. When a data function is applied to a Request item, the data related to that item is aggregated when the query is processed. If you need both summary data and increasing levels of detail breakdown in your reporting or analysis, do not preaggregate the data. Report sections will automatically provide an aggregated summary view, and component levels of detail data can be reached using drill-down tools.

If your data set is potentially very large, or incorporates very discrete levels of transactional data that do not apply to your analysis, it may be best to preaggregate the data at the server to return a more manageable data set. Preaggregating data in your query assumes that you have a clear idea of the data you want to look at and a good conceptual understanding of relational databases. If you are unsure that you want to preaggregate, process the query without applying data functions. If you find that it would be better to preaggregate, you can always return to the Query section and apply data functions to the query.

Use data functions to preaggregate data as it is retrieved from the database. Table 13 lists the prebuilt data functions that you can apply to items in the Request line.

Table 13 Query Section Data Functions

Data Function	Description
None	Returns unaggregated values as stored in the database. This is the default option in the Query section.
Sum	Returns sum of underlying values.
Average	Returns average of underlying values.
Minimum	Returns lowest of underlying values.
Maximum	Returns highest of underlying values.
Count	Returns number of underlying values.
Count Distinct	Returns the number of distinct values in a column. This function is not supported by all database servers.
Standard Deviation	Returns standard deviation of values. This function is not supported by all database servers.
Variance	Returns variance of values. This function is available through Oracle servers only.
Weight	Use for computing weighted items in pivot tables.

When using data functions, remember that with the exception of counts, data functions are applied almost entirely to numeric data items and the results are computed with respect to dimensional, nonnumeric items on the Request line, such as name and date items as in the following examples.

Example 1 Query 1 includes only items State and Units_Delivered. The data function Sum is applied and the data returned consists of one row for each state with an aggregate sum for that state in the Units column.

Request			
Request Region SUM(Amount Sales)		Region	Amount Sales
	1	Americas	\$68,755,83
	2	Asia Pacific	\$29,556,01
	3	Europe	\$34,569,83

Example 2 In Query 2, the item Fiscal_Year is added to Query 1, breaking out rows for each state/fiscal year combination with Units totaled on a per state, per year basis.

Request		×
Request	Fiscal Year Region SUM(Amount Sales)	

	Fiscal Year	Region	Amount Sales
1	1999	Americas	\$11,411,978
2	1999	Asia Pacific	\$4,250,710
3	1999	Europe	\$5,066,636
4	2000	Americas	\$57,343,857
5	2000	Asia Pacific	\$25,305,302
6	2000	Europe	\$29,503,197

Example 3 In Query 3, the Product_Line Name has been included and the data function is changed to Average. The number of rows increased, with data summarized as an average per state, per year, per product line.

Request Fiscal Year Region Product Line AVG(Amount Sales)				
			1	
	Fiscal Year	Region	Product Line	Amount Sales
1	1999	Americas	Books	\$11,717
2	1999	Americas	Music	\$6,073
3	1999	Americas	Videos	\$8,007
4	1999	Asia Pacific	Books	\$6,883
5	1999	Asia Pacific	Music	\$4,755
6	1999	Asia Pacific	Videos	\$6,294
7	1999	Europe	Books	\$6,469
8	1999	Europe	Music	\$4,051
9	1999	Europe	Videos	\$5,542
10	2000	Americas	Books	\$16,215
11	2000	Americas	Music	\$7,334
12	2000	Americas	Videos	\$11,285
13	2000	Asia Pacific	Books	\$9,546
14	2000	Asia Pacific	Music	\$6,342
15	2000	Asia Pacific	Videos	\$7,688
16	2000	Europe	Books	\$8,735
17	2000	Europe	Music	\$5,657
18	2000	Europe	Videos	\$6,982

- To apply a data function in the Query section:
- **1** Select an item on the Request line.
- 2 Select Query > Data Function and select the desired function. (See Table 13 for a list of the available data functions.)

The item is renamed to reflect the data function you selected. For example, SUM(Units) or AVG (Amount_Sold).

When the Query is processed, the data is returned from the server in aggregate form.

- > To remove a data function in the Query section:
- **1** Select the item on the Request line whose data function you want to remove.
- 2 Select Query > Data Function > None.

Appending Queries

When you need to view and merge multiple queries in a combined Results set, there are four query operators that allow you to merge two or more separate queries. The operators and their functions are:

- Union Union All distinct rows selected by either query are retrieved. No duplicate rows are retrieved.
- (1) Union All Union All All rows selected by either query, including duplicate rows, are retrieved.
- Intersection Intersection All distinct rows selected by both queries are retrieved.
 - Except All distinct rows selected by the first query but not the second query, are retrieved. (Oracle database servers refer to the Except operator as "Minus.")
 - **Note:** If your database supports the Intersection and Except operators, but they are not available in the Operator dropdown list, check the Allow SQL-92 Advanced Set Operations connection preference.

The rules governing the use of these operators are:

- The number of columns in the Select clause in both queries must be equal.
- The data type returned in the columns in both queries must match.

For example, if Column 1 in the first query is a date, then Column 1 in the second query must also be a date.

Note: Items on the Union line can be repositioned to see the results of different intersections.

- To append a query:
- **1** Verify data types and associated column(s).

This ensures that you know how to merge data in the second query.

2 Build the Request line.

Add server and local filters, data functions, and computations to the query as needed.

3 Select **Query > Append Query**.

An Operator drop-down list and a second query tab is added below the Request, Filter, and Sort lines. The drop-down list shows whether the queries are linked by way of a union, a union all, an intersection, or an except.

- 4 Build the second query.
- 5 To merge multiple queries, select the operator you want to use from the Operator drop-down list.
- 6 Click Process.

You can have the Interactive Reporting generate automatically the join path required by the context of the query by using the automatic join path feature. This feature eliminates the need for you to predefine any join paths, since the Interactive Reporting determines the paths. When multiple paths are available, you are prompted for which one to use.

Using Local Results

Local results are a snapshot of a Results section shown in topic format. They are used to add the results of one query to another in an Interactive Reporting document (.bqy).

To use local results, click anywhere in the Catalog pane and select Local Results on the shortcut menu.

A Table catalog named Local Results is added to the Catalog pane.

Limitations of Local Results

Since local results are maintained on the desktop and not by the database server, there are limitations when using Local Results. The following functions are *not* available when using Local Results:

- Filters, computed items, data functions, or query properties to further analyze the dataset
- The following Query Options menu governors (disabled if only local results topics make up the Query):
 - o Returning unique rows
 - o Row filter
 - o Time filter
 - o Auto-process
 - o Custom Group by
- Query filters on Local Results Topic Items
- More than one Filter Local Join
- Filter Local Joins used with Local Joins
- Meta topics
- Access or change properties for Local Results Topic Items
- Append Query features of Unions or Intersections with Local Results Topic Items
- Process to table a query

Note: A query based on local result topics will not perform as well as the equivalent database query.

Processing Order

When using *process all*, the query producing the results may be processed twice if the query using its results are listed first in the section catalog. It is also possible for the query using the local results to use *stale* data if it was saved with results, and the query that produced them is reprocessed. To prevent this from happening, you can defined the order in which queries are processed. For more information, see Query Processing Order.

Using Stored Procedures

Stored procedures are precompiled, complex queries that are executed on a database server and maintained by a database administrator. Stored procedures execute very quickly and are usually created to accomplish tasks that SQL cannot do alone. Interactive Reporting treats stored procedures as locked standard queries and does not allow you to modify the procedures.

You can use Interactive Reporting to process stored procedures through Open Client or ODBC, collect the results, and generate reports as you would with a standard query. Stored procedures can be loaded from your desktop and appear as a query object in the Content pane

- **Note: ODBC only.** Interactive Reporting supports stored procedures that return results. This support is contingent on the driver and database. The driver and database must support the required ODBC calls, including SQL Procedures to retrieve a list of available procedures and SQLProcedureColumns which identify the parameters required to execute the procedure. For Oracle, results are recognized in ODBC by specifying reference cursor parameters when the procedure is created. The database connection must specify the database as "ODBC" rather than "Oracle" to work properly.
- Note: The ODBC driver must recognize the ODBC syntax for calling procedures: {call <procedure name> (parameter list) }. If the procedure has no parameters, the parentheses surrounding the parameter list are optional. Interactive Reporting Studio does not insert empty parentheses in the call to execute the procedure. In addition, the driver must accept literal values for any specified parameter. Drivers that require parameter markers, for which values are provided when the procedure is executed, are not currently supported.

> To open a stored procedure:

1 Select **Query > Stored Procedures**.

The Stored Procedures dialog box appears.

2 Select the database owner name that contains the stored procedure.

Any stored procedure to which you have been granted access is displayed in the Stored Procedures list.

3 Select a stored procedure from the list and click **Load**.

The stored procedure appears as an icon in the Content pane. No items appear on the Request line until the stored procedure is processed.

- > To process a stored procedure:
- 1 Click Process.

If the stored procedure calls for user input, a dialog box appears and prompts you with up to 10 entry fields. If more than 10 arguments are required, successive dialog boxes appear.

2 If an argument dialog box is displayed, enter appropriate values as arguments to the stored procedure.

The arguments supplied are similar to variable filters. If necessary, see your database administrator for clarification on the arguments needed to process a particular stored procedure.

3 If the stored procedure queries the database, the database server returns data to the Results section and the adds items to the Request line.

Setting Query Options

When working with very large or unfamiliar databases, you may occasionally process a query that takes a long time to run or returns more data than is manageable. To prevent problems under these conditions, set query options before processing.

- > To set query options:
- **1** Select Query > Query Options.

The Query Properties dialog box is displayed.

Query Properties		
Process Return Unique Rows Return First 100 Rows Time Limit 1.00 Minutes Auto-Process	Cancel Help	
Custom Group By Define		

- 2 Select the desired restrictions for the current query and then click OK.
 - **Return Unique Rows** Eliminates duplicate rows from the data set retrieved by the query. It is not a regression. If the data source for a query is a local results table, even though this feature is enabled, the Unique Rows filter is not enforced.
 - Return First <u>Rows</u> Filters the number of database rows retrieved to the number entered.
 - Time Filter <u>Minutes</u> Filters the amount of time the query is allowed to run to the number entered. Seconds are entered as a decimal number. Time filters work for asynchronous database connections and cancel at the earliest opportunity for nonasynchronous connections.
 - Auto Process Specifies the current query as a Standard Query to be processed automatically on download from the Repository (Designer only).
 - Custom Group By Customizes the Group By criteria used to compute aggregate Request items, with selected items not factored into the grouping. This feature is available only when a data function is placed on a Request item.

Importing Data Files

An alternate method of "querying" is to import data from a file. Interactive Reporting can import files in Microsoft Excel (.xls), and in both comma (.csv) and tab (.txt) delimited text formats.

When you import data, the content of a file is delivered to the desktop data cache and displayed in the Results section. You can use imported data as you would the results of a query to build reports and perform data analysis.

- > To import a data file:
- **1** Select File > Import Data File > Data File.

The Import File dialog box is displayed.

2 Navigate to the location of the file that you want to import.

On a Windows operating system, select the correct type of the import file type menu to make the file easier to find.

3 Select the file that you want to import and click OK.

The data from the imported file is displayed in columnar Results Format.

Setting Data Type Properties

You may want to confirm or change the data type of an item to preserve the precision of a mixed data type computation, or to change the way a data item is handled for example interpreting numbers as strings).

Attention to data types is most important when computing items in the Query section. In this case the database server performs the computation, and the Interactive Reporting may receive the computed item with an unanticipated data type. To ensure that server correctly handles data computations, you should set the data type when performing mixed-data type computations.

Local calculations (Results, Pivot) are handled internally, and adjustment between 16- and 32bit integers, for example, can be handled safely using the automatic or number data type specification.

- > To set data type properties for a request item:
- **1** Select a request item and choose Properties from the shortcut menu.

The Item Properties dialog box is displayed.

- 2 Click Options and selected a datatype from the Datatype drop-down list.
- 3 Click OK.

The following table describes the supported data types:

Table 14 Supported Data Types

Data Type	Specification
Automatic	A data type is determined automatically given the data type of the reference items and the computations performed.
Blob	Binary large object which is truncated at 64,000 bytes.
Byte	Variable data type of length determined by a single byte of computer storage. Bytes can store numeric values from 0 to 255, or a single text character.
Date	Calendar date in server default format (typically mm/dd/yy).
Integer (16-bit)	Retains a 16-bit value (2 bytes). A 16-bit integer stores integer values from 0 to 16,777,216, and signed integers between +8,388,608 and -8,388,608.
Integer (32-bit)	Retains a 32-bit value (4 bytes). A 32-bit integer stores integer values from 0 to 4,294,967,296, and signed integers between +2,147,483,648 and -2,147,483,647.
Long Text	Character data (long text) exceeding 255 bytes (use the string data type for text strings up to 255 characters). The maximum long text retrieved is 4000; characters anything greater than that is silently truncated.
Packed Real	Real numbers packed for use with EDA middleware. The results in the Interactive Reporting Studio are the same as real numbers.
Real	Decimal numbers up to 5 positions right of the decimal.
String	Text strings to a maximum length of 256 characters.
	Interactive Reporting Studio supports very large character data type column fields, such as Microsoft SQL Server text data type files, and IBM's DB2 Character Large Object (CLOB) data types.
	The DB2 CLOB data type is SQL99-compliant and is used to store very large variable-length character strings (up to 10MB). Its length property is displayed with a default value of 4000 characters; however, this may be updated to any value between 1 and 9999 by an Interactive Reporting Studio user. The length property determines the maximum length of the CLOB data string that can be displayed after a database query. CLOB column fields with data that is greater that 9999 bytes in length are truncated after the 9999th character.
Time	Time in format set by user preference.
TimeStamp	Date/time combination in format set by user preference.

Query Menu Command Reference

Table 15 provides a quick reference to the commands available on the Query menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	Shortcut Menu
Process Results To Table	Allows you to create a table in the database to store your results set.		
Estimate Query Size	Queries the database to see how many records your query will retrieve.		
Show Remarks	Displays any remarks recorded about a topic or topic item.	[Ctrl+I]	~
Add Request Item(s)	Adds the selected topic item to the Request line.		~
Add Filter(s)	Allows you to create a filter for the selected topic item.	[Ctrl+L]	~
Add Sort(s)	Adds the selected topic item to the Sort line.		
Add Computed Item	Allows you to add a new data item derived from server-side calculations performed on an existing topic item.		~
Data Functions	Applies a prebuilt data function to the selected Request item.		~
Variable Filter	Designates the selected filter item as variable, which causes Interactive Reporting Studio to prompt the user for the filter values when the query is processed.		~
Customize Filter	Allows you to control access to the features on the Filter dialog box.		~
Append Query	Allows you to combine two or more queries in one Results set.		
Stored Procedures	Loads a stored procedure and displays it as a query object in the Content pane.		
Query Options	Opens the Query Properties dialog box where you can specify options for rows returned, time filters, and so on.		

Table 15 Query Menu Command	Table 15	Query N	Menu	Commands
-----------------------------	----------	---------	------	----------

Working with Query Results

This chapter explains how to work with the results sets obtained from your relational database query or data import.

In This Chapter	Results Section	124
	Enhancing Your Results Set	125
	Saving Results Sets	129
	Exporting Result Sets	131
	Results Menu Command Reference	132

5

Results Section

When you process a query or import data, Interactive Reporting retrieves data to your desktop and displays it in the Results section. Although the query may have accessed several different database tables, the results set is displayed as a single table. Each requested item is displayed as a column in the table and each database record is a row. The Status bar shows the date and time the results set for a particular Results section was last processed (or imported).

Use the Results section to:

- Verify that your query returned the correct information.
- Refine and extend the data set by applying filter conditions or create new computed or grouped items.
- Sort or use text and column formatting features to enhance the appearance of data results.
- Add summary totals or subtotals and compute them with data functions.
- Print or export the retrieved data to other applications.

All reports, including tables, pivots, charts, and those created using the Report Designer, are based on the data that is retrieved to the Results section.

Understanding Data Types

To effectively work with the data in the Results section, you need to understand how Interactive Reporting handles data. Certain functions can only be used on certain types of data.

The Results section formats data in table format. A table is either a fact table or a dimension table. A table is a fact table if it contains at least one fact column. A dimension table contains only dimension columns.

A fact is a quantifiable entity, such as a value or unit of measure. Facts are the numeric values in a relational database that are available for analysis.

A dimension is a descriptive item, such as a name or label.

Enhancing Your Results Set

Before generating reports or exporting the data set, verify that the Results section contains all the information you need. You might want to redesign your query and process it again so that the data set is more manageable.

As you begin to refine your querying technique, you will increasingly use more sophisticated filters and computed items and will understand when to use aggregation in the Query section. Complex queries can be somewhat difficult for new users. When using data aggregation features, it is best to check the data set before creating reports.

You can return to the Query section to modify your query and refresh the data set if necessary, but you can also apply many of the same refinements locally in the Results section, including filters, sorts, and computations.

Enhancing your results set involves:

- Filtering Results
- Sorting Results Data
- Adding Computed Items to Results
- Applying Data Functions to Results
- Adding Grouping Columns
- Adding Columns Automatically
- Breaking Out Dates

Filtering Results

Local filters applied in the Results section enable you to temporarily screen out portions of data for reporting purposes, without eliminating then from the data set.

Local filters are discussed in "Server versus Local Filter Processing" on page 268.

Sorting Results Data

Use the sort buttons to quickly sort a Results column or report item locally on your desktop. You can apply sequenced, nested sort conditions to Request items in the Results section.

For information about sorts, see Chapter 13, "Applying Sorts."

Adding Computed Items to Results

You can rank and provide statistics for the values represented as totals or subtotals in your Results section. The Add Computed Item command enables you to build equations to compute totals, or to apply functions to existing values. Computations are performed on the desktop by a and involve only the data in your Results set. Therefore, you can only create new computed items – you cannot modify original data items that were retrieved from the database. In the Results section, reference items are filtered to the items that is displayed on the Request line of the original query. Also, the scalar functions used to compute items are provided at the desktop level rather than the RDBMS.

For more information about computed items, see Chapter 12, "Working with Computed Items."

Applying Data Functions to Results

In the Results section, you can only use a *data function* for totals and subtotals. The other values cannot be recalculated without redoing the query. Data functions return to the underlying values and recalculate the value according to the type of function specified.

You can apply a break (subtotal), grand, or custom total to any column. Table 17 lists the data functions that you can use with break totals and grand totals.

Function	Description
Sum	Returns sum of underlying values.
Average	Returns average of underlying values.
Minimum	Returns lowest of underlying values.
Maximum	Returns highest of underlying values.
Count	Returns number of underlying values.
Other	Allows you to create a custom function using JavaScript.

Table 17 Break Total and Grand Total Data Functions

Inserting Column Totals

To calculate a column total, select the column to be totaled and click the summation icon on the Standard toolbar.

The adds a row labeled Total to the bottom of the table and display the total as the last entry in the selected column.

Inserting Grand Totals

- > To apply a grand total to a column using a data function:
- 1 Select a column and choose **Results > Grand Total**.

The Insert Grand Total dialog box is displayed.

- 2 Select a data function from the Grand Total Function drop-down list.
- 3 Select one or more columns to be totalled from the Add Grand Total To list and click OK.

The total and any subtotals in the column are computed to reflect the new data function.

Inserting Break Totals

- To apply a break total (subtotal), select a column and choose **Results > Break Total**. The Insert Break Total dialog box is displayed.
- **1** Select a **break column** from the At Every Break drop-down list.
- 2 Select the data function you want to apply from the Break Total Function drop-down list.
- 3 Select one or more columns on which to display the break total and click OK.

Adding Grouping Columns

Grouping columns, like computed items, create new data in your results set by grouping data from an already existing column. You can use grouping columns to consolidate nonnumeric data values into more general group values and map the group values to a new column in the data set.

Grouping columns are new items added to the Results section and are available for use in report sections.

For example, your company sales database may contain the items: State, Sales Region, and Country, which allow you to aggregate data on different levels in reports. However, suppose you are looking to track sales by subregion, or you want to see data for one state versus an average for all other states combined. You can do this by grouping states together to create a Subregion item or other custom dimension.

- To add a grouping column:
- **1** Select a column as a base for your grouped column.
- 2 Select Results > Add Grouping Column.

The Grouped Column dialog box is displayed. Use the column values to build the grouping categories for the new item.

- **3** Type a name for the new column in the **Column Name** field.
- 4 Create custom group values and link them to values in the base column.
 - Click New Groups to create groups and add them to the Groups list.
 - Select a group in the Groups list; then, select items from the Available Values list and use the arrows to add them to the Items In Group list for the selected group.
 - Remove selected values from a group by using the arrow to move them back to the Available Items list.
 - Double-click a group name to modify it.
 - Specify options for ungrouped values as follows:
 - Null Leaves the values ungrouped and unaggregated.
 - **Default** Allows you to specify a default name to assign to all ungrouped values.
 - Individual Group Assigns each ungrouped values the name originally assigned to it.

5 When the grouping definitions are complete, click **OK**.

The new grouping column is added to the Request line and to the Content pane.

Modifying Grouped Columns

You can modify a grouping column to change the group structure.

To modify a grouped column, select the grouped column and choose Results > Modify Column.

Adding Columns Automatically

By default, retrieves data to your desktop and displays it in the Results section as columns. You can manually add request items by having the return an empty Results set. This allows you to add columns as you need them.

This feature allows you to display only the rows with which you want to work. You can still sort or create filters using columns not displayed in the Results section.

To toggle AutoAdd columns, select Results > AutoAdd Columns.

If the AutoAdd Columns feature is selected, all requested items are displayed in columns.

If the AutoAdd Columns feature is not selected, no columns are returned to the Results section and you have to manually add requested items.

Breaking Out Dates

Use date breakout columns to separate date-typed columns into Year, Quarter, and Month items. The new items are automatically derived using date functions available to computed items.

For example, when you add date groups for an item Order Date, the item is broken into constituent date items. A new Year item is created as an integer, Qtr as a string, and Month as a new date.

- To break out date items:
- **1** In the Content pane, select a date-type **column**.
- 2 Select Results > Add Date Group.
- **Note:** This feature automatically sets the display format of the new Month item to mmm so that the data sorts correctly. Quarters are based on the calendar year beginning 1/1.

Working with the Results Table

Interactive Reporting Studio offer a number of options for working with table components (that is, columns and rows) in the Results section. These commands are found on the Format and Results menus. Many of these commands also have corresponding toolbar icons and are available on the shortcut menu.

Selecting Columns and Rows

- To select a column, click anywhere inside the column.
- To select a row, click the row header (row number).

Deleting Columns

- To delete a selected column from the Results table (and Data Layout), select Results > Remove. If an item is removed from the Content pane, it is completely removed from the Data Layout and the data set.
- **Caution!** Remove items with caution as computed items and other report sections may draw data values from the deleted item.

Formatting Commands

You can use the commands available on the Format menu to change the appearance of fonts, backgrounds, borders, color, row heights, and column widths. For more information on formatting options, see "Formatting Text and Other Elements" on page 78. For detailed information on adding document sections and customizing the headers and footers in the document sections, see "Formatting Numeric Data Types" on page 79.

Saving Results Sets

After you process a query, your data is available until you close the document. Saving your document saves the current formatting and layout of all sections in the document.

You have the option to save the results set with the document. You also have the option to save any computed column expressions as a *snapshot*. Your decision in this selection depends largely on how you want to use the information in the document, and on what information needs to be recalculated. Saving results with your query allows you to analyze and generate reports *without being connected to the database*. Results are saved for an individual query or for multiple queries for which results have been generated. You also can specify whether to save any computed columns in the results set as a snapshot with the document.

Note: If you intend to work with a document that includes a Report Designer section, you *must* save results with the document. If you do not save results with the document, the Report Designer section is not available.

Saving your results set makes sense if you cannot connect to a database, for example, when traveling or working remotely, or if you are scheduling or forwarding documents for someone else's use.

> To save results with your document:

1 Select File > Save Options > Save Query Results With Document.

The Save Query With Results Document dialog is displayed showing all of the query sections contained in your document.

2 Select the check box for the query results you want to save and decide whether to save computed columns as snapshots, then click **OK**.

The query results and snapshots for computed columns that you selected are automatically saved the next time you save the document.

Computed values saved as snapshots are not recalculated when the document is opened. Not even dynamic expressions (for example, values that reference the sysdate function such as date or time) are recalculated. They are recalculated only when the query is reprocessed. Documents that are saved with computed columns as snapshots tend to be larger in size than documents that do not contain shapshots, but they take less time to open.

If you want to automatically recalculate the values of computed columns when a document is opened, do *not* select the corresponding results section in the Computed Columns list. The document file may take longer to open, especially if the results set contains a large number of computed columns or uses complex formulas in the definitions, since all computed values are recalculated in the Results section and in any other section that references the Results section. On the other hand, documents that do not contain snapshots tend to be smaller in size than documents that contain snapshots of computed columns.

 Table 18 lists the selection options and effects for saving query results and snapshots of computed columns with documents.

Table 18 Effects of Save Query Results with Document Options

Save Query Results	Save Computed Columns (as Snapshot)	What Happens
X	x	Results are saved with the document and computed columns are saved as a snapshot. Computed columns are <i>not</i> recalculated when the document is opened. Values are recalculated only when the query is reprocessed.
X	©	Results are saved with the document but computed columns are <i>not</i> saved as a snapshot. Computed columns are recalculated when the document is opened.
®	X	Neither Results nor computed columns are saved with the document.
®	0	Neither Results nor computed columns are saved with the document.

Note: You cannot save computed columns as snapshots unless you first save the corresponding query results.

Exporting Result Sets

After processing a query, you can export the data contents of the Results section for use in other applications. There are several ways to export, the most common being into file formats such as Excel or Lotus.

Interactive Reporting also exports too HTML format, making it easy to distribute data to many corporate intranets or Web sites. Exported Results section data is raw and unaggregated. If you export from a report section, the data is drawn from the desktop datacube and is preaggregated. Scripts created by the JavaScript engine can be saved to a text file.Export options are discussed in detail in "Exporting Data" on page 50. This section also covers export properties such as whether to use page breaks in HTML files, or whether to include double quotation marks in tab-delimited text files.

Results Menu Command Reference

Table 19 provides a quick reference to the commands available on the Results menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	Shortcut Menu
Filter	Opens the Filter dialog box.	[Ctrl+L]	~
Sort Ascending	Sorts the selected column values in ascending order (alphabetical or numeric).		~
Sort Descending	Sorts the selected column values in descending order (alphabetical or numeric).		~
Add Computed Item	Opens the Insert Computed Item dialog box.		~
Add Grouping Column	Opens the Grouped Column dialog box. Use to merge dimension labels into new groupings and aggregate the associated data.		~
Add Date Groups	Separates date-type items into year, quarter, and month items.		
Modify Column	Use to modify a computed column or a group column.	[Ctrl+M]	
Remove	Removes the selected column (or Data Layout item).	[Del]	~
Break Total	Opens the Insert Break Total dialog box.		~
Grand Total	Opens the Insert Grand Total dialog box.		~
Hide Column	Hides the selected column from view.		~
Unhide Column	Opens the Unhide Column dialog box.		~
AutoAdd Columns	Automatically adds columns in the Content pane for all requested items.		
	If not selected, no columns are displayed in the Content pane. Turn this option off to manually add columns for requested items.		

able 19	Results Menu Comman	ds
able 19	Results Menu Comman	(

Working with Tables

This chapter explains how to use tables to organize your data.

6

In This Chapter	Table Section. 13	4
	Creating Tables	4
	Tables as a Data Staging Area 13	4
	Manipulating Table Data	5
	Working with Table Components	0
	Table Menu Command Reference 14	1

Table Section

A *table* is a columnar arrangement of data. Tables are used as building blocks in other reporting sections. A *Table section* functions is used in much the same way as the *Results section*. All of the commands that are available on the Results menu are also available on the Table menu. However, you cannot apply complex filters or aggregate data in the Table section.

The data in a Table section is derived from the section that is active when you insert a new table. When you insert a Table section from the Results section, the new Table Section is attached to the Results section. This means any changes to the results set are propagated immediately to the Table section.

You can also create additional tables, Pivot tables, Charts, and other reports form a dataset of a Table section just as you would from the Results section. Filters, computed columns, *grouping columns*, and other actions that modify the dataset of an active section set carry over to all tables and reporting sections built from that section.

Creating Tables

Use the Table Data Layout to construct, plot, and manage data.

- > To create a table based on the Results section data:
- 1 From the Results section, select **Insert > New Table**.

If the Table Data Layout is not already displayed, click Data Layout on the Section title bar.

2 Drag Results items from the Catalog pane to the Table Data Layout.

The populates automatically the table columns.

Tables as a Data Staging Area

A *computed item* in the Results section becomes a mere data element when added to a Table section. Thus, the Table section can become an intermediate calculation staging area. This ability comes in handy in applications where you wish to place filters on computed items.

For example, suppose you wanted a report of the top ten producers in your organization. It is easy to add a computed item to your Results section that is based on the Rank function. (This function lets you find out each producer's rank relative to the others.) However, when you try to filter the results based on that ranking, informs you that filters cannot be placed on aggregate items.

To get the results you want, you need to insert a new Table section that is based on your Results section. Then you add all the relevant Results items, including the computed Rank field. Once Rank is a column in the table, it is no longer a computed item. It is a regular number on which you can now place a filter. Since the Table section is based on the Results section, your Top 10 report is updated automatically each time you run the query.

Manipulating Table Data

The Table menu provides a number of commands that enable you to manipulate the data in the Table section. Review the following sections for information on:

- Filtering Data in a Table
- Sorting a Table
- Adding Computed Items
- Adding Grouping Columns
- Adding Date Groups
- Applying Data Functions to Tables

Filtering Data in a Table

Filtering data in a table filters the data displayed in the table columns. You can apply *filters* in the Table section in addition to any filters set in the originating section. Filters set in the Table section are automatically propagated to any other reports that inherit their data set from the table. You can apply only one filter per column.

To return data to the display and make it available for reporting, delete or suspend the filter.

- > To filter data in a table:
- 1 Double-click the column in which you want to place a filter, or click the column heading and select Table > Filter.[Ctrl+L]

The Filter dialog box is displayed.

- 2 Select an arithmetic or logical operator from the drop-down list box.
- **3** Define the potential filter values by selecting one of the following options:
 - Show Values Shows column values associated with the item.
 - Custom Values Supplies an empty field for inputting custom values. Select the check mark to add a value to the list.
- 4 In the Values list, select the values to include in the filter definition.

Individually select values or click **Select All** and deselect the values that you do not want to include.

5 When the values are highlighted in the values pane, click **OK**.

The filter is applied to the column and the column name is added to the Filter line.

To remove a filter in a table, select the filter item that you want to remove and choose Table > Remove.

- > To remove a filter in a table, select the filter and do one of the following:
 - Click the **Delete** key.
 - Select Table > Remove.
 - Click **Remove** on the shortcut menu.
- > To remove all filters in a table, click **Filter** on the Filter line and select **Table > Remove**.
- > To remove all filters in a table, click Filter on the Filter line and do one of the following:
 - Click the **Delete** key.
 - Select Table > Remove.
 - Click **Remove** on the shortcut menu.

Sorting a Table

You can sort the rows in a table by one or more columns in ascending or descending order. You can also apply sequenced, nested sorts to columns in the Table section.

- To sort a column, select the column you want to sort and click the Sort button on the Standard toolbar or select Table > Sort Ascending or Sort Descending.
- > To apply sort conditions using the Sort line:
- **1** Click **Sort** on the Section Title bar to display the Sort line.
- 2 Drag Results items from the Catalog pane to the Sort line.

You can add items to the Sort line that are not in the Data Layout.

3 Establish a final sort sequence by reordering sort items.

Items are sorted left to right on the sort item. To reorder the sequence, drag each item to its new position.

4 Double-click specific sort items to toggle ascending and descending sort orders.

Ascending is the default sort order.

5 Click **Sort Now** on the Sort line.

Adding Computed Items

You can rank and provide statistics for the values represented in the totals or subtotals. The Add Computed Item feature enables you to build equations to compute totals, or to apply functions to existing values. Computed items are like normal data items and can be included in reports or reused to compute other data. For example, you can modify an *Amount Sold* item by building an equation around it, multiplying by a *Unit Price* item, and renaming the resulting item *Revenue*. You can apply a scalar function such as Cume to *Amount Sold* and return each individual value as a cumulative running total, or simply multiply *Amount Sold* by the local tax rate to find the tax owed on each sale.

The Computed Item dialog box is used to build a computed item expression. The computed item expression is a value, variable, logic statement, or equation that instructs the how to perform a computation.

- > To create a computed item:
- 1 Select Table > Add Computed Item.

The Computed Item dialog box is displayed.

2 In the Name field, type a name that describes the computation.

The default name is Computed, which is numbered sequentially if there is more than one. If you assign a name to a computed item that is identical to an existing scalar function name, Interactive Reporting numbers the name starting with the number 2.

3 Define the new data item by building an expression in the Definition text box.

Use the operator buttons to insert arithmetic and logical operators at the insertion point.

- Click **Reference** to display the Reference dialog box, and select Request items to place in the equation.
- Click Functions to apply scalar functions using the Functions dialog box.

You can also type any portion of the equation or the entire equation directly into the Definition text box using *JavaScript*. The names are case sensitive, and you must replace spaces in item names with underscores ('_').

- 4 If necessary, click **Options** to set a new data type for the item.
- 5 When the equation is complete, click **OK**.

The computed item is added to the Data Layout and it is displayed as a column in the table.

Adding Grouping Columns

Grouping columns is a way of creating new data in your results set by grouping data from an already existing column. You can use grouping columns to consolidate non-numeric data values into more general group values and map the group values to a new column in the data set.

Grouping columns are new items added to the Table section and are available for use in report sections.

For example, your company sales database may contain the items: State, Sales Region, and Country, which enable you to aggregate data on different levels in reports. However, suppose you are looking to track sales by subregion, or want to see data for one state versus an average for all other states combined. You can do this by grouping states together to create a subregion item or other custom dimension.

- To add a grouping column:
- **1** Select a column from which to base grouping column.

2 Select Table > Add Grouping Column.

The Grouped Column dialog box is displayed.

Use the column values to build the grouping *categories* for the new item.

- **3** Type a name for the new column in the **Column Name** field.
- 4 Create custom group values and link them to values in the base column.
 - Click New Groups to create groups and add them to the Groups list.
 - Select a group, and then select items from the Available Values list and use the arrows to add them to the Items In Group list for the selected group.
 - Remove selected values from a group by using the arrow to move them back to the Available Items list.
 - Double-click a group name to modify it.
 - Specify options for ungrouped values as follows:
 - Column Name Names the new grouping column in the table.
 - New Groups Creates a custom group to be displayed as a value in the new grouping column.
 - **Options** Indicates how to represent unassigned values within the grouping column, that is, as null values, as members of a default group (named in the adjacent edit field), or as their own individual groups.
 - Groups Selects a custom group to define by adding or removing items.
 - Items In Group Removes an item from a selected custom group.
 - Available Values Adds items to a selected custom group.
 - Select one of the following options to define the preferences for ungrouped columns:
 - Null Leaves the values ungrouped and disaggregated.
 - Default Enables you to specify a default name to assign to all ungrouped values.
 - Individual Group Assigns each ungrouped values the name originally assigned to it.
- 5 When the grouping definitions are complete, click **OK**.

The new grouping column is added to the Data Layout and to the table.

Modifying Grouping Columns

You can modify a grouping column to change the group structure.

> To modify a grouping column, select the grouping column and choose **Table > Modify Column**.

Adding Date Groups

Use date breakout columns to separate date-typed columns into Year, Quarter, and Month items. The new items are automatically derived using date functions available to computed items.

For example, when you add a *date group* for an item *Order Date*, the item is broken into constituent date items. A new *Year* item is created as an integer, *Qtr* as a string, and *Month* as a new date.

- > To break out date items:
- **1** Select a **date-type column** in the Content pane.
- 2 Select Table > Add Date Group.
- **Note:** This feature automatically sets the display format of the new Month item to mmm so that the data sorts correctly. Quarters are based on the calendar year beginning 1/1.

Applying Data Functions to Tables

In the Tables section, a *Dashboard* can be used only for totals and subtotals. Data functions return to the *underlying values* and recalculate the value according to the type of function specified.

You can apply a break (subtotal), grand, or custom total to any column. A grand total on a numeric column applies a default sum function. However, each column can have a number of grand totals, each with a different aggregate function applied to it. Table 20 lists the data functions that you can use with break totals and grand totals.

Function	Description
Sum	Returns sum of underlying values.
Average	Returns average of underlying values.
Minimum	Returns lowest of underlying values.
Maximum	Returns highest of underlying values.
Count	Returns number of underlying values.
Other	Enables you to create a custom function using JavaScript.

Table 20 Break Total and Grand Total Data Functions

Column Totals

To calculate a column total, select the column to be totaled and click the summation icon on the Standard toolbar.

Interactive Reporting adds a row labelled *Total* to the bottom of the table and displays the total as the last entry in the selected column.

Grand Totals

- > To apply a grand total to a column using a data function:
- **1** Select a column and select **Table > Grand Total**.

The Insert Grand Total dialog box is displayed.

- 2 Select a data function from the Grand Total Function drop-down list box.
- 3 Select one or more columns to be totaled from the Add Grand Total To list and click OK.

The total and any subtotals in the column are computed to reflect the new data function.

Break Totals

- To apply a break total (subtotal):
- **1** Select a column and select **Table > Break Total**.

The Insert Break Total dialog box is displayed.

- 2 Select a break column from the At Every Break in drop-down list box.
- 3 Select the **data function** that you want to apply from the Break Total Function drop-down list box.
- 4 Select one or more columns on which to display the break total and click **OK**.

Working with Table Components

offers a number of options for working with table components (that is columns and rows) in the Table section. These commands are found on the Format and Results menus. Many of these commands also have corresponding toolbar icons and shortcut menu items.

Selecting Columns and Rows

> To select a column, click anywhere inside the column.

> To select a row, click the row header (row number).

Deleting Columns

To delete a selected column from the Results table (and Data Layout), select Results > Remove. If an item is removed from the Content pane, it is completely removed from the Data Layout and the data set.

Table Menu Command Reference

Table 21 provides a quick reference to the commands available on the Table menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	Shortcut Menu
Filter	Opens the Filter dialog box.	[Ctrl+L]	~
Sort Ascending	Sorts the selected column values in ascending order (alphabetical or numeric).		~
Sort Descending	Sorts the selected column values in descending order (alphabetical or numeric).		~
Add Computed Item	Opens the Insert Computed Item dialog box.		~
Add Grouping Column	Opens the Grouped Column dialog box. Use to merge dimension labels into new groupings and aggregate the associated data.		~
Add Date Groups	Separates date-type items into year, quarter, and month items.		
Modify Column	Use to modify a computed column or a group column.	[Ctrl+M]	
Remove	Removes the selected column (or Data Layout item).	[Del]	~
Break Total	Opens the Insert Break Total dialog box.		~

Table 21 Table Menu Commands

Caution! Remove items with caution as computed items and other report sections may draw data values from the deleted item.

Table 21 Table Menu Commands (Continued)

Command	Description	Keyboard Shortcut	Shortcut Menu
Grand Total	Opens the Insert Grand Total dialog box.		~
Hide Column	Hides the selected column from view.		~
Unhide Column	Opens the Unhide Column dialog box.		~

Querying Multidimensional Databases

This chapter explains how to use Interactive Reporting Studio to connect to and query a multidimensional database.

7

In This Chapter	OLAPQuery Section	144
	Defining OLAPQuery Options	145
	Refining OLAPQuery Data	151
	Processing OLAP Queries	157
	Applying Filters	158
	Changing Data Views	163
	Formatting OLAPQuery Items	166
	Drilling Through from a Multi-Dimensional Database to a Relational Database	166
	OLAP Menu Command Reference.	169

OLAPQuery Section

	Use the OLAPQuery section to query multidimensional) or OLAP databases. When you connect to a multidimensional database, the document opens an OLAPQuery section.
	The user interface is similar to the Pivot section, except the OLAPQuery section displays the multidimensional database components as a hierarchical tree in the Catalog pane. These components include:
(A)	• Dimensions – <i>categories</i> of information, such as Location, Products, Stores, and Time. Dimensions are for informational use only and cannot be used as query items. (If a dimension contains multiple hierarchies, they will be represented under the dimension.)
🗗 Level (All)	 Levels – Groups of similar types of members. For example, using the members listed in a Location dimension, Japan, USA, and France belong to the Country level. San Francisco, Paris, Tokyo, and Rome belong to the City level. 35 Main Street belongs to the Address level.
E Can2	 Hybrid (Analysis) – Further dimensional data is present by way of Hybrid Analysis (the lowest levels of an Analytic Services or DB2 cube that can reside in a relational database, but are not reflected in the cube structure (metadata) that is resident on the Analytic Services Server itself). If a dimension has Hybrid Analysis data, the Analytic Services Server, in conjunction with the Analytic Services Integration Server automatically retrieves the data from the appropriate relational table source and passes it to Interactive Reporting. If a dimension has available hybrid analysis, it is indicated to the right of the level as shown below.
·⊡ Drink ·⊡ Food	 Members – Content values for a dimension. In the Location dimension, they could be San Francisco, Japan, Paris, 35 Main Street, Tokyo, USA, France, Rome, and so on. These all are values for location.
Ð	Members are shown only under the Values (I) icon and can be dragged only into the Top Labels or Side Labels panes in Data Layout.
	• Member Property – A descriptive piece of information about a member that can be retrieved and displayed in the OLAPQuery. This information is metadata and does not in itself constitute a distinct member in the dimensional hierarchy. For example, let's assume the following hierarchy:
	Product
	Category
	Product Name
	The Product Name level might have the following properties defined:
	Product Description
	Product SKU
	Color
	Size
	Weight

You can drag a property into the Data Layout with its corresponding level, but you cannot drag it into the Slicer or the Measures pane, or apply a filter to it.

• Measures – Numeric values in a database cube that are available for analysis. The measures could be margin, cost of goods sold, unit sales, budget amount, and so on.

Individual measures are shown under the Measures icon and can be dragged only to the Measures pane.

Note: Analytic Services only. supports Analytic Services *Attribute Dimensions*. An Attribute Dimension displays in the OLAPQuery section with the word "attribute" to right of the Attribute Dimension's name. In addition, for each Attribute Dimension, creates an *Attribute Calculation* Dimension. An Attribute Calculation Dimension is displayed at the bottom of the Catalog pane. You can position the members contained within the Attribute Calculation Dimension in the Top or Side Labels of the Data Layout.

Defining OLAPQuery Options

enables you to set options that control various properties in the OLAPQuery section. You can define both general and database-specific options.

> To define OLAPQuery options, select **OLAP > OLAP Query Options**.

General OLAPQuery Options

General OLAPQuery options can be adjusted regardless of the database. The General tab in the OLAP Query options dialog box is displayed as follows:

Table	22	General	OLAPOuer	v Options
		aonorai	0 II (Quoi,	, epaono

Option	Description	
Design Options	Defines how retrieves database totals and requeries the database.	
	• Database Totals - Retrieves database totals when the query is processed.	
	• Hardwire Mode – Defines whether to requery the database when changes are made in the Data Layout.	
	 If you select Hardwire mode, requeries the database automatically when you add an item to or remove an item from the Data Layout and instantaneously retrieve the data. You do not have to click Process. 	
	 If you do not select Hardwire mode, you must click Process to requery the database whenever you make a change. 	

123

 Table 22
 General OLAPQuery Options

Slicer Display	Defines how to display slicer values.
	• Member List – Selects from a list of all members at the same level as the previously selected member.
	• Tree Control – Displays parent-child slicer value relationships. With Tree Control, you can select multiple values from a dimension as long as your database supports this function. (Databases such as Hyperion Analytic Services and IBM DB2 OLAP allow this).
Drill Options	Defines what level of data is the next level displayed when you drill down in an OLAPQuery.
	• Drill to Next Level – Automatically displays data for the next level below the selected member.
	For example, in a dimension with levels of <i>Year</i> , <i>Quarter</i> , <i>Month</i> , and <i>Date</i> , double- clicking on a Year level member name automatically displays all the data for the Quarter level belonging to that year.
	• Drill to All Levels – Automatically displays all possible levels of data below the selected member.
	For example, in a dimension with levels of <i>Year, Quarter, Month</i> , and <i>Date</i> , double- clicking on a Year level member name automatically displays all the data for the Quarter, Month, and Date levels belonging to that year.
	• Drill to Lowest Level – Automatically displays data for <i>only</i> the lowest level belonging to the selected member (intermediate member levels are not shown).
	For example, in a dimension with levels of <i>Year</i> , <i>Quarter</i> , <i>Month</i> , and <i>Date</i> , double- clicking on a Year level member name automatically displays all the data for the Date level belonging to that year.
Database-specific OLAPQuery Options

Database-specific OLAPQuery options depend on the database to which you are connected. As a result, the content on the tab in the dialog box varies according to the type of database connection.

Display Options for Analytic Services Databases

The tab for Analytic Services databases is displayed as follows:

 Table 23
 Display Option for Analytic Services Database

Option	Description
Query Options	Defines query options in the following areas:
	• Suppress Missing Rows – Suppresses the retrieval of any missing rows where all cells are null.
	• Suppress Zero Rows – Suppresses the retrieval of any zero rows where all cells are null.
	• Specify the number of decimal places to retrieve – Sets the number of decimal places that the server will return.
	• Enable Hybrid Analysis Value Retrieval This feature allow you to retrieve the lowest levels of an Analytic Services cube that reside in an external relational database. These levels are not reflected in the cube structure (metadata) that is resident on the Analytic Services Server itself. Instead, their existence and retrieval is performed by the separate Analytic Services Integration Server product.
	In order for this functionality to be performed, however, the Analytic Services server must first receive instructions to enable Hybrid Analysis retrieval. This instruction takes the form of an Analytic Services Report Script keyword:
	< HYBRIDANALYSISON>
	Conversely, to disable Hybrid Analysis retrieval, the instruction keyword
	<hybridanalysisoff></hybridanalysisoff>
	The applicable keyword used sent to the Analytic Services Server can be viewed on the Query Log.
	This feature is only supported in Hyperion Analytic Services version 6.5 and IBM DB2 OLAP version 8.1.
Alias Table	Defines the alias table to use in an OLAPQuery.
	• Select an Alias Table – When you use aliases to assign user-friendly names to database physical member and/or generation/level names, Analytic Services stores the aliases in an Alias Table in the cube. Since a cube can have multiple alias tables, you can select the alias table to use and modify the query based on the value you enter.
	For example, you could define Store Category members as codes, but define an alias for each Store Category to use as a descriptive alias in the Alias Table.
	In the following example, for the member name "0199", you could see either "January 11999, "Jan99" or Fiscal Month 1" depending on the selected alias table. By default, uses the default alias table and if another alias table were not selected, you would see "January 1999."
	Physical Member Name = 0199
	Default Alias Table Value = January11999
	Alias Table 1 Value = Jan99
	Alias Table 2 Value = Fiscal Month 1

Display Options for Microsoft OLE/DB

The tab for Microsoft/OLE DB for OLAP databases is displayed as follows:

Table 24	Display	Options for	or Microsoft	OLE/DB
----------	---------	-------------	--------------	--------

Option	Description
Suppress Empty Rows	Suppresses the retrieval of any empty rows for which there is no measure data.

Building OLAP Queries

Interactive Reporting Studio use the Data Layout to manage multidimensional information. The Data Layout enables you to plot, view, and manipulate dimensions. The three Data Layout panes are:

- Side Labels Contains non-quantifiable dimensions or members.
- Top Labels Contains non-quantifiable dimensions or members.
- Measures Contains quantifiable dimensions or measures. The Measures pane in Data Layout holds the values of the cube.
- Note: Interactive Reporting treats time and date dimensions as non-quantifiable values.
 - ► To build an OLAP query:
 - 1 Create an OLAP connection file, or if you already have an OLAP connection file, select Insert > New OLAPQuery.

The Insert OLAP Query dialog box is displayed.

2 Specify an OCE to use to connect to the multidimensional database; enter your user name and password if prompted.

The Interactive Reporting document creates and switches to an OLAPQuery section. The Catalog pane displays the hierarchy of your multidimensional database.

- 3 If the Data Layout is not visible, click **Data Layout** on the Section title bar.
- 4 In the Catalog pane, select one or more measures (such as units or amounts) and choose OLAP > Add Fact/Measure to add the item(s) to the Measures pane in Data Layout.
- 5 In the Catalog pane, select one or more levels or members and choose **OLAP > Add Side Labels** to add the item to the Side Labels pane in Data Layout.
- 6 In the Catalog pane, select one or more levels or members and choose OLAP > Add Top Label to add the item to the Top Labels pane in Data Layout.
- > To refresh the dimension values in the Catalog pane, select **OLAP > Retrieve Dimensions**.

When you drag items from the Catalog pane to the Data Layout, only the level names is displayed. For example, if you drag CA into the Data Layout, the Data Layout displays State. Level names are displayed in the Data Layout preceded by a $\overline{\gamma}$ icon.

You can reorient, or pivot, your OLAPQuery by interchanging the items in the top and side dimensions. This feature is useful for juxtaposing data in one dimension with data from other dimensions. By pivoting dimensions from the top to the side, alternate relationships become evident.

OLAPQuery Section Data Layout Rules

The following rules apply to the OLAPQuery section Data Layout:

- A dimension can be represented only on one axis. For example, if the level Year is in the Side Labels pane, you cannot drag the level Quarter to the Top Labels pane.
- If you move the level of one dimension, Interactive Reporting Studio automatically move all levels of the same dimension.
- To pivot data in the OLAPQuery section, move the items in the Data Layout panes. You cannot use the label handles to pivot data.
- Levels from the same dimension must be grouped together in both the Side Labels and Top Labels panes.

For example, you cannot use the following order for side-label levels, since levels from different dimensions are mixed (*Year, Quarter,* and *Month* come from the *Time* dimension, *Store Type* comes from the *Store* dimension, and *Product Category* comes from the *Product* dimension).

Year
Product Category
Quarter
Store Type
Month

Instead, the Data Layout requires that you use this order (*Year, Quarter*, and *Month* are all from they same dimension so they are grouped together).

Year	or	Product Category
Quarter		Year
Month		Quarter
Product Category		Month
Store Type		Store Type

- The hierarchy of a dimension cannot be broken. For example, *Year* must come before *Quarter*, which must come before *Month*.
- Analytic Services only If "Hybrid Analysis" is available and data is successfully retrieved from the Analytic Services servers, a "dummy" Level is added to the Data Layout called "Hybrid1". The new Level of data is shown in the Contents Pane. If data is not retrieved successfully from the Analytic Services servers (i.e. an error message is returned) the

message "No more levels to drill into" message is displayed. If you subsequently drills-down on a returned Hybrid Analysis Level of data, the retrieved Hybrid Analysis Level of data is called "Hybrid2" and so on.

• OLE DB only – If you retrieve dimensional level properties from the database, you can drag each property into the Data Layout after you add its corresponding level.

For example, if the Data Layout contains Country, Year and you drag Manager, (a property of Country) into the Data Layout, the Manager property is kept with Country and not added after the Year level. A property cannot be dragged into the Slicer pane, the Measures pane in Data Layout, or have a filter applied to it.

Note that in order to retrieve dimensional level properties, you must enable the Show Member Properties checkbox when creating the OCE.



OLAPQuery Member and Level Rules

If you include both members and levels together in a query, a union of the two data sets occurs, and not an intersection. For example, if you select a State level and then select San Francisco (which is a city), your query retrieves all states and San Francisco.

The following able shows the results of different queries.

Data Layout item(s)	Component Type	Query returns:
Country (All)	Level	All countries
State (All)	Level	All states regardless of country
City (All)	Level	All cities regardless of state or country
Canada	Member	Canada only
СА	Member	California only
San Francisco	Member	San Francisco only
State (All), San Francisco	Level, Member	All States regardless of country and San Francisco
Country (All), CA	Level, Member	All Countries and California (not just the USA)
Country (All), State (All), CA, NV, San Francisco	Level, Level, Member, Member, Member	All Countries, All States

Table 25 OLAPQuery Member and Level Rules

Refining OLAPQuery Data

Once you have identified the items to include in your OLAP query, you can perform numerous operations to refine the data such as setting slicer filters, specifying drill data, adding computed items, data functions, and so on. Review the following sections for information on each of these functions.:

- Specifying a Slicer
- Drilling Down
- Drilling Up
- Hybrid Analysis and Drilling (Analytic Services and DB2 only)
- Adding Computed Items
- Using OLAPQuery Functions

Specifying a Slicer

A slicer is a sort of third axis in a query that filters data. The other axes are the row axis and the column axis. A slicer defines a logical slice of the server cube by instructing the server to ignore all values not part of your slice. For example if you were running a query for general category stores, you could apply a slicer that slices the category stores into store subsets, such as computer stores, discount stores, and electronic stores.

When working with a slicer, use only an individual member from a dimension. The dimension cannot be used in a Top Label or Side Label (no dimension can be represented on more than one axis at any time).

- Tip: A query can have multiple slicers, each from a different dimension.
 - To specify a slicer:
 - 1 Click Slicer on the Section title bar to open the Slicer pane.
 - 2 Select a member from a dimension in the Catalog pane and drag it to the Slicer pane.

Every dimension folder contains a members subfolder named "Values for …" that domain. The subfolder contains the members eligible for selection in the Slicer pane.

3 Click Process.

If you are running in Hardwire mode (see "Processing OLAP Queries Automatically" on page 157), the slice is applied instantly.

Drilling Down

The Drill Down feature retrieves data from the multidimensional database cube, following the hierarchy down to the granular level. When you find a specific item that you want to learn more about, such as a product line, you can drill down into the item label. You can drill down on more than one item as well as drill down on all items at the same time.

For a member drill down, any Top Label or Side Label can be drilled down so that you can view the structure of the hierarchies for any particular dimension. Every time you select a specific label in a dimension row or column, you show only the data for that label value. When you select the dimension tab for a level, you show all the members of that dimension level.

Use one of the following methods to drill down on a label:

- Double-click the label.
- Select the label and choose Drill Down on the shortcut menu.
- Select the label and choose OLAP > Drill Down.
- **Tip:** You can specify what level of data is the next level displayed when you drill down in an OLAPQuery. See "Drill Options" for information.
- Note: You cannot set filters while in a drilled-down state on a dimension.
- **Note: Analytic Services only.** For a measure drill down, you can show how different measures consolidate together. A drill down on a measure is done on a progressive basis, one level at a time on a 1 to n path (sequential rather than nested). For example, if Profit is the parent of Tax and Pre-Tax Profit, and Revenue and Expenses are children of Pre-Tax Profit, then the Tax and Pre-Tax columns are drilled down first and you must double-click the Pre-Tax label to display the *Revenue* and *Expense* columns.

Drilling Up

If you used the Drill Down feature, you return to your original view of the data by drilling up one level at a time. To drill up, select the level to drill up to and use one of the following methods:

- Double-click the level.
- Choose Drill Up on the shortcut menu.
- Choose OLAP > Drill Up.

Hybrid Analysis and Drilling (Analytic Services and DB2 only)

When you drill down to retrieve Hybrid Analysis data, the drill down retrieves *only* the next level of information, even if there is a selection made for "Drill Through", or the "Drill to all Levels" or "Drill to Lowest Level" options have been enabled on the OLAP Query Options dialog.

If you can drill into data returned by way of hybrid analysis, Interactive Reporting adds a "dummy" level to the Data Layout called "Hybrid1" and displays the new Level of data accordingly in the Contents Pane. If data is not retrieved successfully from the Analytic Services servers, you get the message: "No more levels to drill into" message.

If you continue to drill down on the Hybrid Analysis level of data, each successful drill down returns the data as "Hybrid2" in the Data Layout (and "Hybrid3", "Hybrid4"... etc. for any subsequent Hybrid Analysis Level retrievals).

You can use drill up on Hybrid Analysis data to redisplay that level as the lowest level in the Contents Pan. All lower level Hybrid Analysis data is removed from the Contents Pane and Data Layout.

Adding Computed Items

Computed items allow you to create a new column by building an expression to compute measures, or by applying functions to existing measures. Computed items are like normal data measures and can be included in reports or reused to compute other measures.

Computed items are displayed in virtual columns (as opposed to columns that are physically stored in the cube). They are automatically calculated during the query and supplement the information already stored in the database.

For example, you can modify the Amount Sold item by building an expression around it, multiplying by the Unit Price item, and renaming the resulting item Revenue.

Note: This feature is only available for an MS OLAP database.

- > To compute or modify a measure:
- **1** Select **OLAP > Add Computed Item**.

The Modify Item dialog box is displayed.

Modify Item	
Name: Computed Item	ОК
Definition	Cancel
[Measures].[Units] * [Measures].[Amount]	Help
Add measures to the expression by selecting from the items below. To add a measure select the measure and press insert. For advanced functions press the functions button to view a list of the available functions. Measures:	
Units Amount	
Insert	
Functions	

- 2 Specify the information requested in the following fields:
 - Name Specify a new column name that reflects the computation result.
 - Definition Build an expression by adding items from the pad or the Functions dialog box.
 - Use the keypad to select and insert arithmetic and logical operators.
 - If you are familiar with MDX (Multi Dimensional eXpressions), type your instruction directly in the Definition field.
 - Measure Select the MDX equivalent from the list of available measures for the expression.
 - Functions Apply a numeric function to a selected measure in the Definition field. Depending on the function you select, the Functions dialog box changes to accommodate the selected function. For more information about functions, see "Using OLAPQuery Functions" on page 154.

3 When the expression is complete, click **OK**.

The new measure name is added to the Data Layout.

For more information on computed items, see Chapter 12, "Working with Computed Items."

Note: You can only add computed items if your database supports them. Examples of databases that support computed items are OLE DB for OLAP-compliant databases such as MS OLAP and SAP BW.

Using OLAPQuery Functions

Use OLAPQuery functions to insert standard numeric functions in computed measure expressions. Numeric functions compute a new measure for each value associated with it. You can use two types of OLAPQuery functions in the OLAPQuery section:

- Interactive Reporting Functions Non-MDX functions that allow you to perform common mathematical computations in MDX. The % of Column, % of Row, and % of Total functions allow you to use only a measure name from the query (not all measures in the cube) as a parameter.
- MDX functions Standard mathematical functions that you apply to computed item expressions. Hyperion Intelligence Clients supports a number of MDX functions. (For more information on MDX functions, consult your MDX documentation.)

Table 26 provides a quick reference to the commands available on the Query menu and lists any related shortcuts.

The following table lists the functions available in the OLAPQuery section.

Function	Type of Function	Description
% of Column	Interactive Reporting	Calculates the value of the specified measure as a percentage of the total for the column.
% of Row	Interactive Reporting	Calculates the value of the specified measure as a percentage of the total for the row.
% of Total	Interactive Reporting	Calculates the value of the specified measure as a percentage of the total for all rows and columns.
% Change	Interactive Reporting	Calculates the percentage change of the specified measure for a particular dimension from the previous member in that dimension. For example, this function could be used to calculate the percentage change from sales from the previous year.
Absolute Change	Interactive Reporting	Calculates the absolute change of the specified measure for a particular dimension from the previous member in that dimension. For example, this function could be used to calculate the difference in sales from the previous year.
Avg	MDX	Calculates the average of the selected measure evaluated over the specified dataset.
Correlation	MDX	Returns the correlation of a dataset against two measures.
Count	MDX	Calculates the number of members for the specified dimension in the report.
Covariance	MDX	Measures the tendency of two values to vary together. Variance is the average of the squared deviation of a value from its mean. The covariance is the average of the values of the deviations of feature values from their means.
Linregpoint	MDX	Calculates the linear regression of a dataset and returns the value of "b" in the regression line $y = ax + b$.
Linregr2	MDX	Calculates the linear regression of a dataset and returns r2 (the coefficient of determination).
Linregslope	MDX	Calculates the linear regression of a dataset and returns the value of "a" in the regression line $y = ax + b$.

Table 26 OLAPQuery Functions

Table 26	OLAPQuery	/ Functions	(Continued)
----------	-----------	-------------	-------------

Function	Type of Function	Description
Linregvariance	MDX	Calculates the linear regression of a dataset and returns the variance that fits the regression line $y = ax + b$.
Мах	MDX	Returns the maximum value of the selected measure evaluated over the specified dataset.
Median	MDX	Calculates the median value of the selected measure evaluated over the specified dataset.
Min	MDX	Returns the minimum value of the selected measure evaluated over the specified dataset.
Stdev	MDX	Calculates the standard deviation of the selected measure evaluated over the specified dataset.
Sum	MDX	Calculates the sum of the selected measure evaluated over the specified dataset.
Variance	MDX	Calculates the variance of the selected measure evaluated over the specified dataset.

To apply a data function:

1 Select OLAP > Add Computed Item and click Functions in the Modify Item dialog box.

The Functions dialog box is displayed.

2 Select the Interactive Reporting or MDX function that you want use from the Functions list.

A description of the selected function is displayed below the Functions list and explains the type of calculation the function performs.

3 Select the **measure** to which you want to apply the function.

You can select any measure in the cube, not just a measure in the query. The Cube Hierarchy list shows the organization of the cube including both members and levels.

Some functions require that you specify a second measure to perform the calculation. See the specific function if you are required to specify a second measure.

The Count function requires no measure.

4 Define the dataset by which to evaluate the function and click **OK**.

To add a member or level to the dataset from the Cube Hierarchy list, select a member or level and click **Add**.

To remove a member or level from the dataset, select a member or level and click Remove.

Processing OLAP Queries

After you build your OLAP query and apply filters, computations, sorts, and any other adjustments to further refine your request, you need to process it. Processing your query may take a few moments if your query is complex, or if the data in linked report sections needs to be refreshed.

> To process an OLAP query, choose Tools > Process Query > Option.

Since a document can contain multiple queries, the Process drop-down list has three processing options:

- Process Current Processes the current object. In some cases more than one query may be processed, for example, if a report references results sets from multiple queries. Process Current is the default selection when using the toolbar button.
- Process All Processes all the queries in the document.
- **Process Custom** Opens the Process Custom dialog box so that you can indicate which queries to process by selecting a query's check box.

sends the query to the database and retrieve the data to the OLAPQuery section. While the data is being retrieved, the Status bar displays a dynamic count indicating rate and progress of server data processing and network transfer.

Processing OLAP Queries Automatically

If you select to run in Hardwire mode Interactive Reporting requeries the database automatically when you add an item to or remove an item from the Data Layout and instantaneously retrieves the data. You do not have to click Process.

Note: You should consider the size of the cube you are querying to determine whether to use Hardwire mode.

To select Hardwire mode, choose OLAP > OLAP Query Options and select Hardwire Mode in the Design sectionally the General tab.

Working with an OLAPQuery Offline

To view, plot, and work with an OLAPQuery offline, download the data set to an OLAPResults section within the document. Once downloaded, the data can be integrated with the Chart, Table, and other reporting sections. If you need to modify the query, reconnect to the database and apply any necessary changes.

► To download the OLAPQuery data set, select OLAP > Download To Results.

An OLAPResults section is created for the query. You can use the OLAPResults data set to insert a new chart, pivot, or other report.

Creating a OLAPResults Section Automatically

You can have creates a Results section automatically when you click **Process**. This eliminates the need to select **OLAP** > **Download to Results** (see "Working with an OLAPQuery Offline" on page 157).

- > To automatically create a Results section when you click Process:
- 1 Select Tools > Options > Program Options and choose the OLAP tab.
- 2 Select Auto-Generate Results Section When Processing an OLAP Query.

When you select this option, Interactive Reporting creates an OLAPResults section automatically for any OLAPQuery section that you create in this session when that OLAPQuery section is first processed.

Applying Filters

Filters enable you to define and apply filters to a query once Top Labels or Side Labels have been added to the query. You set filters by applying comparison operators on the values for a specific member. Review the following sections for information on:

- Applying Member Selection Filters
- Applying Measure Filters (Analytic Services)
- Applying Variable Filters

Applying Member Selection Filters

Use a member selection filter to filter data retrieved from the server cube. A member selection filter is similar to a slicer, except that the member selection filter introduces the member value in your report, and multiple members may be selected from a single hierarchy.

To apply a member selection filter, drag individual member values from the Catalog pane to the Data Layout.

Another method of member selection uses an expression to dynamically retrieve the list of members that satisfy selected parameters, for example, the Top N or Bottom N. You specify these parameters in the Filter dialog box.

- **Note:** If you used the Drill Down feature to navigate down to a members level, you have to use the Drill Up feature to return to the original level before you apply Member Selection filters.
 - > To apply a member selection filter using an expression:
 - **1** Drag a level into the Data Layout and double-click the level name.

The Filter dialog box is displayed.

2 Specify the information requested in the Filter dialog box and click **OK**.

Applying Measure Filters (Analytic Services)

You can filter data retrieved from the server cube with a measure filter, which is similar to a member selection filter. A measure filter uses an expression to dynamically retrieve the list of measures that satisfy selected parameters, for example, the Top N or Bottom N. You specify these parameters in the Filter dialog box. Additional parameters are available based on the selected multidimensional database.

- > To apply a measure filter:
- **1** Double-click a **measure** in Data Layout.

The Filter dialog box is displayed.

2 Select the data operator from the Data Operator drop-down list.

The selections shown on this list depend on the database to which you are connected. See "Operator Types and Data Operators" on page 162.

- 3 Specify any database specific parameter requests, such as a column index (that is, the column on which to apply the measure filter) or value.
- 4 Move the member(s) to the Applied Filters list and click OK.

Applying Variable Filters

A variable filter is a filter you specify when you process a query. You can use variable filters for standardized documents that you distribute to many users, or to automatically reset filters when you need new conditions every time you run a particular query. A (v) next to an item indicates it has a variable filter.

You place a variable filter on an item using the Filter dialog box, which is accessed in either the Slicer pane or the Data Layout, depending on your database. Table 27 specifies how various OLAP databases access the Filter dialog box.

Table 27 How Databases Access the Filter Dialog Box

Database	Access Points
Analytic Services	Side Labels pane, Top Labels pane, Measures pane, Slicer pane
OLE DB	Side Labels pane, Top Labels pane, Slicer pane

Member Variable Filters

- To set a *member* variable filter:
- **1** In the Data Layout, double-click the item you want to define as a member filter.

The Filter dialog box is displayed.

2 Select the method for retrieving items from the database from the Operator Type list.

The selections shown on this list depend on the database to which you are connected. See "Operator Types and Data Operators" on page 162.

- 3 Specify any database-specific parameter requests.
- 4 Click **Show Values** to display values in the database.
- 5 In the Members field, select the member(s) that you want to define as a filter.
- 6 Move the member(s) to the **Applied Filters** list.
- 7 Click Set As Variable and click OK.

The OLAPQuery section is redisplayed.

8 Click Process.

The Filter dialog box is redisplayed.

- 9 Select the member(s) that you want to define as a variable filter in the Members field.
- **10** Move the member(s) as a variable filter to the **Applied Filters** list and click **OK**.

Measure Variable Filters

- > To set a *measure* variable filter:
- **1** Double-click the item to define as a **measure filter** in the Data Layout.

The Filter dialog box is displayed.

2 Select the data operator from the Data Operator drop-down list.

The selections shown on this list depend on the database to which you are connected. See "Operator Types and Data Operators" on page 162.

- 3 Specify any database-specific **parameter** requests, such as a column index or value.
- 4 Move the measure to the **Applied Filters** list.
- 5 Click Set As Variable and click OK.

The OLAPQuery Section is redisplayed.

6 Click Process.

The Filter dialog box is redisplayed.

- 7 Choose a measure variable filter by selecting the **data operator** from the Data Operator drop-down list and specifying any database-specific **parameter** requests, such as a column index or value.
- 8 Move the measure variable filter to the Applied Filters list and click OK.

Slicer Variable Filters

- To set a *slicer* variable filter:
- 1 Click **Slicer** on the Section title bar to open the Slicer pane.

2 Select a member from a dimension in the Catalog pane and drag it to the Slicer pane.

Every dimension folder contains a members subfolder named Values For *Domain*, which contains the members that are eligible for selection in the Slicer pane.

3 Double-click the member in the Slicer pane that you want to use to filter data.

The Slicer dialog box is displayed.

- 4 Select the member for which you want to filter data.
- 5 Click the Set As Variable field and click OK.

The OLAPQuery section is redisplayed.

6 Click Process.

The Slicer dialog box is redisplayed.

7 Select the member(s) you want to use as a variable filter and click OK.

Analytic Services Substitution Variables

In Analytic Services, a substitution variable acts as a dynamic filter. Substitution variables are defined on the server using Analytic Services's Application Manager. Your administrator names the substitution variable and set its value equal to a user-defined parameter. For example, a variable might be named *Latest Period* and have its value set to equal *November*. When the latest period ends, the variable's value could be reset to *December*, and so on.

An advantage of this type of variable is that saved queries capture the variable's name instead of hard coding the actual value. Each time the query is run, different data could be returned if the variable's value has been changed on the server.

- > To apply a substitution variable:
- **1** Double-click an item in the Data Layout.

The Filter dialog box is displayed.

2 Click Substitution Variable in the Operator Type drop-down list.

The Filter dialog box retrieves all available variables in the Substitution Variables list.

3 Move the member(s) to the Applied Filters list and click OK.

You can also double-click the member to add it to the Applied Filters list.

The OLAPQuery section is redisplayed.

4 Click Process.

The Filter dialog box is redisplayed.

- 5 In the Members field, select the member(s) that you want to use as a variable filter.
- 6 Move the member(s) you want to use as a variable filter to the Applied Filters list and click OK.

Operator Types and Data Operators

The OLAPQuery section supports the following operator types and data operators:

- OLE DB Operator Types and Data Operators (see Table 28)
- Analytic Services Operator Types and Data Operators (see Table 29)

Note: Not all providers support all operators.

Operator Type	Description
Select Members from DB	Creates a filter based on a member(s) retrieved from the database. (Supported for the Analytic Services, MSOLAP and SAP/BW databases).
Select by Measure	Creates a filter based on a measure that you specify.
Тор N	Retrieves only the top N values where each top N value is at least the specified Index value.
Top N %	Retrieves only the top N $\%$ values where each top N $\%$ value is at least the specified Index value.
Top Sum	Retrieves the top N (the smallest number possible) values, such that their sum is at least the specified Index value.
Bottom N	Retrieves only the bottom N values where each bottom N value is at least the specified Index value.
Bottom N %	Retrieves the bottom N% where each bottom N $\%$ value is at least the specified Index value.
Bottom Sum	Retrieves the bottom N (the smallest number possible) values such that their sum is at least the specified Index value.
Data Operator	Description
= Equal	Retrieves only records where the filtered item equals the specified value(s).
< > Not Equal	Retrieves only records where the filtered item does not equal the specified value(s).
< Less than	Retrieves only records where the filtered item is less than the specified value(s).
< = Less than or Equal to	Retrieves only records where the filtered item is less than the specified value(s).
> Greater than	Retrieves only records where the filtered item is greater than the specified value(s).
> = Greater than or Equal to	Retrieves only records where the filtered item equals, or is greater than the specified value(s).

 Table 28
 OLAP Query Operator Types and Data Operators

Operator Type/ Data Operator	Definition	
Equal	Retrieves only records where the filtered item equals the specified value(s).	
Match Member	The Match Member performs a trailing-wildcard member search for member names and alias names that match the pattern you specify. It returns the member and alias names it finds. The search uses the Analytic Services formula: mbrName genName levName, "pattern"). The character pattern to search for, including a wildcard character (* or ?). ? substitutes one occurrence of any character. You can use ? anywhere in the pattern. * substitutes any number of characters. You can use * only at the end of the pattern. To include spaces in the character pattern, enclose the pattern in double quotation marks ("").	
Data Operator	Description	
= Equal	Retrieves only records where the filtered item equals the specified value(s).	
< > Not Equal	Retrieves only records where the filtered item does not equal the specified value(s).	
< Less than	Retrieves only records where the filtered item is less than the specified value(s).	
< = Less than or Equal to	Retrieves only records where the filtered item is less than the specified value(s).	
> Greater than	Retrieves only records where the filtered item is greater than the specified value(s).	
> = Greater than or Equal to	Retrieves only records where the filtered item equals, or is greater than the specified value(s).	

Table 29 Analytic Services Operator Types and Data Operators

Changing Data Views

In the OLAPQuery section, there are numerous ways to change the way you view the data. Changing data view involves:

- Suppressing Rows
- Adding Totals
- Adding Data Functions
- Showing OLAP Results as a Chart

Suppressing Rows

You can suppress the following types of rows:

- Missing Rows (*Analytic Services only*) Suppresses the retrieval of any missing rows where all cells are null.
- Zero Rows (*Analytic Services only*) Suppresses the retrieval of any zero rows where all cells are null.

- Empty Rows (*OLE DB only*) Suppresses the retrieval of any empty rows for which there is no *measure* data.
- To suppress rows in your OLAPQuery data, select OLAP > OLAP Query Options and click the DB Specific tab.

The options that are displayed on the DB Specific tab depend on the database to which you are connected. See "Database-specific OLAPQuery Options" on page 147 for more information.

Adding Totals

enables you to add either database totals or local totals. Database totals are calculated by querying the actual database. Local totals are calculated and applied to surface values on the Interactive Reporting Studio.

Adding Database Totals

You can include or exclude database totals to tailor the look of the OLAPResults section and any charts you create using the Show As Chart feature. When you activate the Database Totals feature, totals are retrieved into the OLAPResults section as additional rows or columns. In the Chart section, database totals are plotted.

The OLAPQuery section includes database totals by default. You may wish to turn off this feature if you intend to use the Drill Down feature to navigate the multidimensional cube, or if you plan to export the OLAPQuery section.

- To add database totals:
- 1 Choose OLAP > OLAP Query Options and click Database Totals in the Design section of the General tab.
- 2 Click **Process** to add the totals to the data.

displays the result as the *first* item at each level of the dimension.

- **Note:** If you enable database totals in the OLAPQuery section, Interactive Reporting copies them as *static values* into any Results section. As a result, they will *not* be treated as dynamically updated totals.
 - To remove database totals, remove the checkmark next to Database Totals on the General tab and click **Process** to reprocess the query.

Adding Local Totals

To add local totals, select the desired dimension handle, right-click, and choose Add Totals. The totals the data and display the result as the *last* item at each level of the dimension. > To remove local totals, click on a total label and press the **Delete** key.

Adding Data Functions

Column or row totals added to the OLAPQuery section are aggregates and can be recalculated using data functions. When applied to totals, data functions apply the calculation to *surface* values.

When applied to surface values, data functions recalculate the values in the visible cells or surface of the OLAPQuery section. For example, you can show the total sale, average sale, and maximum sale of each product by quarter. Each of these dimensions is based on the same value. They only differ in the data function that is applied.

- To apply a total function:
- **1** Select the dimension handle for a particular measure.
- 2 Click the summation icon on the Standard toolbar to calculate the total.
- **3** Select a column of a particular measure.
- 4 Choose OLAP > Data Function and select a function.

The following table lists the data functions available in the OLAPQuery section.

Table 30 OLAP Data Functions

Data Function	Description
Sum	Returns sum of all values. This is the default function in all report sections.
Average	Returns average of all values.
Count	Returns number of values.
Maximum	Returns highest value.
Minimum	Returns lowest value.
% of Column	Returns surface values as a percentage of their respective column item.
% of Row	Returns surface values as a percentage of their respective row item.
% of Grand	Returns surface values as a percentage of all like values in the report.

Showing OLAP Results as a Chart

You can perform interactive analysis on the OLAPQuery data by viewing the data as a chart. When you select this option, an OLAPQuery Results section is automatically created, as an OLAP Chart section.

Note: Because the Chart created by the Show As Chart command is stationary, you cannot perform drill-down analysis on it. In addition, it is recommended that you deactivate the Database Totals feature since the chart plots the totals when totals are retrieved from the database.

> To show the query as a chart, choose OLAP > Show As Chart.

Formatting OLAPQuery Items

You can use the commands available on the Format menu to add corner and data labels, and to change the appearance of fonts, backgrounds, borders, color, row heights, and column widths. For more information on formatting options, see Formatting Text and Other Elements.

Drilling Through from a Multi-Dimensional Database to a Relational Database

In general, OLAP data is aggregated and obtained from a relational database source. As a result, there may be occasions where you want to see the relational data associated with the multi-dimensional data.

For example, assume that you create an OLAP analysis to show your company's sales aggregated to Country, State, and City levels. Assume further that the sales data for each store within a city is stored in a transactional, relational database. In this case, you could drill down in the OLAPQuery section to sales data for USA > California > San Francisco. To see the data for the stores in San Francisco, however, you would need to drill through to a relational database.

Drilling through from a multi-dimensional database to a relational database involves:

- Setting Drill-through Options
- Drilling Through

Setting Drill-through Options

Drill-through options define the mapping between a multi-dimensional database and a relational database.

- ➤ To set drill-through options.
- 1 Choose OLAP > Set Drill-Through.

The Set Drill-Through dialog box is displayed.

Dimension to Topic Mapping	Specify Relational Query:
	Query
Dimensions:	Relational Topics:
	Heta Data Columns Meta Data Joins Meta Data Tables Product Id Product Line Product Family
Products	Products
Map Dimension >>	<< Un Map Dimension

2 Fill in the options in the Set Drill-Through dialog box and click **OK**.

Table 31 Drill Through Options

Option	Description	
Dimensions	The dimensions in the current OLAP query. Click a dimension to select it.	
Selected Dimension	The dimension selected by the user. This is the dimension to use for mapping.	
Specify Relational Query	The relational query sections that are currently available. To specify a relational query, click the arrow to the right of the drop-down list and select a query from the list that is displayed.	
Relational Topics	The topics contained in the selected relational query. Click a topic to select it.	
Selected Topic	The topic selected by the user. This is the topic to use for mapping. The selected topic should have topic items with names corresponding to the selected dimension levels.	
Map/UnMap Dimension	Maps the selected dimension to a topic (or unmaps the selected dimension from a topic). When you map a dimension to a topic, Interactive Reporting Studio store an internal link between the dimension and the selected topic.	
Specify Fact Topic	The topic used to map to the OLAP measures. The selected topic should have topic items with names corresponding to the OLAP measures.	
	To specify a topic, click the arrow to the right of the drop-down list and select a topic from the list that is displayed. The topics that appear are the topics available in the selected query.	

Tip: In order to drill-down to any level in the relational data, enable the "Set as Dimension" property for the relational topics that represent the OLAP dimension data. To do this, right-click the topic in the original relational query section Contents pane, select Properties, and click the checkbox next to "Set as Dimension" in the Topic Item Properties dialog box.

Drilling Through

- > To drill through from a mulit-dimensional database to a relational database:
- **1** Select a dimension and drill-down to the lowest level.
- 2 Do one of the following:
 - Double-click the dimension.
 - Select OLAP > Drill Through.

The Drill-Through dialog box is displayed enabling you know that there are no additional OLAP levels to drill into.

3 Click Yes to drill through to the relational data source.

Interactive Reporting creates a new Pivot Section (along with associated Query and Results sections) for the relational data.

- **Note:** ignores slicers when drilling through to a relational database. If a slicer is present, a message is displayed letting you know that the slicer will be ignored.
 - **Tip:** After you drill through on a dimension and create a new Pivot section, you can return to the OLAPQuery section and drill down on additional dimensions if desired. When you drill down on an additional dimension, you can choose whether to create a new Pivot section or update an existing pivot section with the new dimension data.
- **Tip:** If you update an existing pivot section with new data, ensure that the new data to be added to the Pivot section maps to a Fact Topic that is the same as the Fact Topic in the existing Pivot.

OLAP Menu Command Reference

The following table provides a quick reference to the commands available on the OLAP menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	Shortcut Menu
Retrieve Dimensions	Refreshes the dimension values in the Catalog pane.	[F9]	
Add Side Label	Adds the selected item to the Side Labels pane.		
Add Top Label	Adds the selected item to the Top Labels pane.		
Add Fact/Measure	Adds the selected item to the Measures pane.		~
Add Computed Item	Opens the Computed Items dialog box. (This feature is for MS OLAP only)		v
Drill Down	Allows you to progressively narrow your focus on a selected item.		~
Drill Up	Returns the original view of data.		~
Drill Through	Drills through from a multi-dimensional database to a relational database.		
Remove Total	Removes local totals from selected dimensions.	[Del]	~
Hide Items	Removes selected items from the OLAPQuery report.		~
Show Hidden Items	Retrieves hidden items from the selected row or column to the OLAPQuery report.		~
Show All Items	Retrieves all hidden items to the OLAPQuery report.		~
Download To Results	Downloads the OLAPQuery data set to an OLAPResults section for offline work.		
Show As Chart	Charts the OLAPQuery data set; automatically creates OLAPChart and OLAPResults sections.		
OLAP Query Options	Accesses the OLAPQuery Options dialog box, where you can set options for your OLAPQuery.		
Data Functions	Recalculates the surface values in the OLAPQuery data set.		~
Set Drill-Through	Accesses the Set Drill-Through dialog box, where you can define the options for drilling through from a multi- dimensional database to a relational database.		

Table 32 OLAP Menu Commands

Analyzing Data with Pivot Tables

This chapter explains how to use pivot tables to quickly summarize or cross-tabulate large amounts of data.

In This Chapter	Pivot Section	172
	Creating a Pivot Table	172
	Working with Pivot Tables	173
	Using Data Calculations in Pivot Tables.	175
	Selecting Pivot Table Elements	174
	Pivot Section Aggregation and Surface Values	183
	Pivot Menu Command Reference	197

8

Pivot Section

The Pivot section enables you to extract meaningful information from your *query* results. Pivot tables are interactive tools used to slice and dice data for ad-hoc, interactive, and multidimensional analysis. Pivot tables enables you to add, move, rename, focus on, and group dimensions to gain customized views of the data. rotate or *pivot* rows and columns to see different summaries of data or display the details for areas of interest. You also can automatically include subtotals and grand totals, or use your own formulas by adding computed items.

Creating a Pivot Table

Pivot tables are made up of:

- Facts Core numeric data that you slice and dice dimensionally in your analysis.
- Dimensions Descriptive items that break aggregate data (facts) into logical categories. In the Pivot section, dimensions are either Column Labels or Row Labels.

For example, if you analyze *Unit Sales* by region, the numbers are your data values or facts. *Region* is a dimension. Presented in aggregate, facts are subdivided by your chosen dimension labels.

- > To create a pivot table:
- **1** Select Insert > New Pivot.
- 2 If the Data Layout is not visible, click Data Layout on the Section title bar.
- **3** Perform one of the following actions:
 - Drag each Request item to be included in the table from the Catalog pane to a pane in the Data Layout (Column Labels, Row Labels, or Facts).
 - In the Catalog pane, select one of more Request items and select **Pivot > Add Selected** Items > Column Labels, Row Labels or Facts
- **Tip:** Add Request items such as *Unit Sales* or *Amount Sales* to the Facts pane in Data Layout to create the data grid. Add dimensions such as *Product Line* or *Region* to the *column and row labels* panes in the Data Layout to create subdivisions.
- **Tip:** You can select and drag multiple items to the same Data Layout pane to add multiple values or dimensional levels for analysis. Items are hierarchically ordered in the sequence in which they are displayed in the Data Layout pane.

Pivoting Data

Use the Pivot feature to reorient the *axes* of a *pivot table* and view your data in new ways. *pivoting* a table enables you to more easily compare the new data to data in the originating table, which makes pivot tables more powerful than common spreadsheets.

To pivot data in a table, select a dimension handle and drag it to any position on the same or opposite dimensional axis.

Charting a Pivot Table

You can automatically generate charts from your current pivot table to view a graphic representation of your data.

> To automatically chart your pivot table, select Insert > Chart This Pivot.

Interactive Reporting creates a new Chart section that displays a bar chart based on the data from the pivot table.

Working with Pivot Tables

You can select pivot table elements and perform a wide range of tasks with data elements. Since report sections organize data hierarchically, if you alter a data value item, all instances of the item within the report are affected. Review the following sections topics for information on:

- Selecting Pivot Table Elements
- Moving Pivot Table Elements
- Changing Label Nesting Levels
- Sorting Pivot Tables

Selecting Pivot Table Elements

- To select a facts column for formatting, layout, or modification, click anywhere on the column of data values. Do *not* click on the label.
- To select a column of Row Labels or a row of Column Labels for formatting, layout, or analysis, click the *dimension handle* at the end of the column or row of labels.
- To select one complete row or column for formatting or analysis, press the modifier key ([Alt] for Windows or [Ctrl+Alt] for Motif. Then, select the row or column label.
- > To select an individual Column Label or Row Label for formatting, select the label itself.

Moving Pivot Table Elements

- To move a column to a new location in the Content pane, select the column in the Content pane and drag it to a new position.
- To remove Request items from the Data Layout or columns from the Content pane, select the element that you want to remove and select Pivot > Remove Selected Item.
- **Note:** When you delete a Request item from the Data Layout or a column from the Content pane, you cannot use the Undo feature to reinsert the column or Request item.
 - To move a pivot element item using the Data Layout, click the item name in the Data Layout pane to select it, then drag the item to a new position or to another Data Layout pane.

The display updates to reflect the repositioning or reassignment of the item.

Note: To move items between Column Labels or Row Labels panes and the Fact pane (or vice versa), you must first remove them from the Data Layout, and then add them again to the chosen pane.

Changing Label Nesting Levels

In pivot tables, labels from one dimension frequently are nested within another dimension. Nesting means that one set of labels is displayed as a subdivision of labels at a higher *layer* of data. You can change the way labels nest to emphasize different relationships.

For example, you can show Year and Quarter as data items in the Column Labels pane in the Data Layout. The Quarter labels (Q1, Q2, Q3, and Q4) are nested within each year label (1998, 1999). If you move Year after Quarter, then each year is displayed as a subset of each quarter. In this case, Q1 values are broken down by labels 1998, and 1999.

- To change the nested level of labels:
- 1 With more than one data item in an Data Layout pane, select a data item in the Data Layout.
- 2 Drag that item to the other side of the second data item in the same pane in the Data Layout.

The labels in those dimensions switch positions and the data is nested in a different manner.

Sorting Pivot Tables

Sorting facts or dimensions enables you to display objects in ascending and descending order according to value.

- To sort plotted values and labels:
- 1 If the Sort line is not visible, click **Sort** on the Section title bar.
- 2 In the Sort list, select the item that you want to use as the basis of your sort.
- 3 In the By drop-down list box, click the sort type (either a label or value).
- 4 In the Using drop-down list box, select the method of calculation for a data value.

By default Interactive Reporting plots data in ascending order. To sort in descending order, click the descending icon.

Using Data Calculations in Pivot Tables

The Pivot section provides a number of ways to perform data calculations that can help you analyze business trends. These calculations range from simple totals and subtotals that are useful in most types of pivot tables to more complex data function for specialized contexts. Review the following sections for information on:

- Adding Totals and Subtotals
- Adding Cumulative Totals
- Using Data Functions
- Using Surface Values in Data Functions
- Using Weighted Averages
- Adding Computed Items

Adding Totals and Subtotals

You can calculate totals for both columns and rows in a pivot table. If you layered dimension items along the column or side of your pivot table, you can calculate totals for any *layer* in the *hierarchy*. When you select inner dimensions for totaling, subtotals are created for each of the categories in the outer dimensions.

For example, assume your pivot table has facts of Units and Amount Sold. These facts are further broken down by Region and Territory on the side, and by Year and Quarter on column. Calculating totals by Region produces a total row at the bottom of the pivot table, summing the data from all regions for each column. Calculating totals by Quarter produces one total column under each year label, summing the data for each set of four Quarter labels.

- **Tip:** An intelligent aggregate is applied to the specified data when totaling unless you specify otherwise. For example, the total of a column of averages calculates an average rather than a sum total.
 - > To add totals to a pivot table:
 - **1** Click a row or column dimension.
 - 2 Select Pivot > Add Totals.

totals and breaks them according to the next higher dimension item.

- > To add subtotals to pivot tables:
- **1** Select an inner dimension.
- 2 Select **Pivot > Add Totals**.

Interactive Reporting adds subtotals to each one of the categories of the next higher dimension.

Adding Cumulative Totals

Add cumulative totals to break totals by dimension and restart them at each dimensional grouping in a pivot table. Cumes work best when all dimensions are located at the column or side of the pivot table, and data label column heads are placed orthogonally.

- To add a cumulative total:
- **1** Select a fact in the data grid of the pivot table.
- 2 Select Pivot > Add Cume.

The Pivot Cume dialog box is displayed.

Pivot Cume			
Name:	Cume of Unit Sales	OK	
Scope:	Default	Cancel	
	Default Region Territory Year		

- **3** If desired, type a new name for the pivot cume.
- 4 Select the scope of the pivot cume from the drop-down list box.

The Scope drop-down list box includes all of the dimensions in the pivot table. The default scope is the lowest level dimension that is displayed in the pivot table.

5 Click OK.

A new fact column is added that maintains a cumulative running total of the original fact by the dimension (scope) specified.

Using Data Functions

A data function enables you to change the nature of the values displayed in a pivot table and enables you to decide the kind of value represented in a pivot table. When you use a data function, Interactive Reporting recalculates the selected values according to the function applied to the underlying data values (which are originally from the Results section).

Data functions are particularly useful if you want to display different types of values side by side. If you add the same fact (such as *Amount Sales*) to the Data Layout several times, you can apply a different data function to the very same dimension.

For example, you can show the *total* sale, *average* sale, and *maximum* sale of each product by quarter. Each of these computed items uses *Amount Sales* as its underlying value. They only differ in the data function used to calculate them.

- **Note:** When you add multiple instances of a Request item to the Facts pane in the Data Layout, Interactive Reporting Studio appends number to the name (for example, Amount_2, Amount_3).
 - To apply a data function:
 - **1** Select a fact in the data grid of the pivot table.
 - 2 Select Pivot > Data Function > Function.

The data values are recalculated and populate the row or column of the pivot table.

The table below lists the data functions available in the Pivot section.

Function	Description	
Sum	Returns sum of all values. This is the default setting.	
Average	Returns average of all values.	
Count	Returns number of values.	
Maximum	Returns highest value.	
Minimum	Returns lowest value.	
% of Column	Returns surface values as a percentage of their respective column item.	
% of Row	Returns surface values as a percentage of their respective row item.	
% of Grand	Returns surface values as a percentage of all like values in the pivot table.	
Increase	Returns the incremental difference between the final two instances of a total column or row. Apply only at the innermost dimensional level of a pivot table.	

 Table 33
 Pivot Data Functions

Table 33 Pivot Data Functions

Function	Description
% Increase	Returns the percent difference between the final two instances of a total column or total row. Apply only at the innermost dimensional level of a pivot table.
Non-Null Average	Returns average of values; null values excluded.
Null Count	Returns number of null values.
Non-Null Count	Returns number of values; null values excluded.

Note: Null values are empty values for which no data exists. Null values are not equal to zero.

Using True Computed Item Totals

In the Pivot Section, break totals can be recalculated to have their value equal to the sum of their displayed detail cells. "True computed item totals" use aggregation according to the specified data function and do not rely on the computed item total formula. You also have the option to use the break total cell results derived from the computed item formula applied to the detail cell.

In this example, detail cell data is shown at the City level and break totals are shown when the State changes. The *Sum* data function has been applied.

]
	Month	Sales	Detail Cells
1	Jan	10	
2	Feb	15	-
3	Mar	17	
4	Apr	20	
5	May	22	
6	Jun	20	
7	Jul	25	
8	Aug	27	
9	Sep	30	
10	Oct	35	Break Total (
11	Nov	37	
12	Dec	40	

The "Computed" column is defined by the following formula:

(Units % 50) +1

where "%" represents modulo (remainder) operator. In other words, the formula is defined as: Integer remainder of ("Unit column cell value" /50) + 1 For the Unit column values for each city within a state, the formula works as expected. For example, in the Oakland, CA cell, the formula is:

Units 910

Modulo (remainder) of 910 / 50 = 10

Add 1 to assign a value of 11 (shown above).

For the California "Total" row, the value shown is 41, which is the result of the following formula:

```
Total "Units" for California = 12390
```

Modulo of 123900/50 = 40

Add 1 to assign a value of 41

The Modulo of 41 is not the sum of the displayed cell values for all cities in California, instead it is the modulo formula applied only to the cell containing the "Unit" column city total for California.

To see a break cell total value of *145*, use the True Computed Item feature total, which would reference the displayed values in the detail cells (this example assumes a *Sum* data function):

46 + 1 + 11 + 1 + 46 (= 145

To use true totals in a break total cell:

1 Select **Pivot Options** from the Pivot menu.

The General tab of the Pivot Options dialog box is displayed.

2 Select True Computed Item Totals and click OK.

Pivot Options			
Ge	eneral		
	 Use Surface Value True Computed Item Totals Consider Null Fact Value As Zero 		
	Cancel Help		

Using Surface Values in Data Functions

Data functions, when applied to total rows or columns, can either apply calculations to *surface values* (the values displayed in the pivot table) or *underlying values* (the values from the original Results section).

When applied to surface values, data functions recalculate the values in the visible cells or surface of the pivot table. When applied to underlying values, data functions return to the unaggregated values beneath the pivot table and recalculate based on those values. When

underlying values are used, the results often are displayed incongruous with the aggregate surface values of the chart element. In other words, a total of the underlying values does not match the total of the surface figures.

Consider a simple pivot table with two values of 20 and 30. Each of these is already a total of underlying values:

```
20 = 8 + 12
30 = 10 + 20
```

An average of the underlying values yields the result of:

 $12.5 = (8 + 12 + 10 + 20) \div 4$

An average of the surface values yields a result of:

 $25 = (20 + 30) \div 2$

To match surface-level values in your calculation, you can instead apply surface values to the totals derived from data functions. For example, if you use surface values for an average applied to a total, the total is converted to the average of the surface values in the corresponding element.

To use surface values, select Pivot > Use Surface Values.

Using Weighted Averages

Weighted averages are useful for a variety of purposes, such as survey research or when you want to include demographic information in your pivot tables. For example, assume you took a survey of 100 people, 75 male and 25 female. But according to census data in that geographic region you should have surveyed 50 males and 50 females. The data you have is skewed toward males.

To correct for this, you assign a weight or *weighting factor* to correct for the sampling error in your survey. To calculate a weight you take the expected amount and divide it by the actual amount.

In the example, the men would have a weighting factor of:

 $50 \div 75 = 0.6666$

The women would have a weight of:

 $50\div 25~=~2$

Any calculation would calculate each man as 0.6666 and each woman as 2.

Weighted averages can also be used to apply different levels of importance to a given item. Take, for example, a survey, which has multiple questions. The responses can be rated on a scale of 1 to 5. By assigning a weight to each question based on the level of importance (the higher the number the more important), and using that weight in calculating a weighted average, you can arrive at averages that are more meaningful.

To use weighted averages, you must add a column of data to the database. This data indicates the relative weight of each corresponding value in another column. The statistical calculation for weighted averages depends on the following mathematical formula:

 $(c \times w) \div sum(w)$

Table 34	Example of Weighted Averages

Customer ID	Units Sold	Type of Store	Weight
1435	80	Electronics	8
1539	200	Computer	10
1634	60	Electronics	8
1213	900	Discount	2

- To use weighted averages:
- **1** Ensure that a column of data with the weighted values exists in the database.
- 2 In the Query section, select the Topic item for which weighted values are needed.
- 3 Select Query > Data Functions > Weight.

The Reference dialog box is displayed.

4 Select the item that contains the weighted values and click **OK**.

The item in the Request line is renamed to indicate it is a weighted value.

5 Process the query.

The weighted values are returned in the Results section.

- 6 Go to the Pivot section and drag the Weighted item from the Catalog pane to the Data Layout. You may now use all of the various data functions on the weighted values.
- Note: Weighting functions work only in the Pivot section.

Adding Computed Items

Use the Add Computed Items command to create new elements in the Pivot section. Computed items enable you to build equations or apply functions to existing data values. Computed items are like normal data items and can be included in pivot tables or reused to compute other data. To add a computed item:

1 Select **Pivot > Add Computed Item**.

The Computed Item dialog box is displayed.

Computed Item				
Name: Computed	(OK)			
	Cancel			
Definition	Functions			
	Reference			
	Options			
	Help			
_				
+ · * / () == != and or if else mod < <= > >= not				

2 In the Name field, type a name that describes the computation.

The default name is Computed, which is numbered sequentially if there is more than one. If you assign a name to a computed item that is identical to an existing scalar function name, Interactive Reporting numbers the name starting with the number 2.

3 Define the new data item by building an expression in the Definition text box.

Use the operator buttons to insert arithmetic and logical operators at the insertion point.

- Click **Reference** to display the Reference dialog box, and select Request items to place in the equation.
- Click Functions to apply scalar functions using the Functions dialog box.

You can also type any portion of the equation or the entire equation directly into the Definition text box using JavaScript. The names are case sensitive, and you must replace spaces in item names with underscores ('_').

- 4 If necessary, click **Options** to set a new data type for the item.
- 5 When the equation is complete, click **OK**.

The computed item is added to the Data Layout and it is displayed as a column in the pivot table.

Considering Null Fact Values in Pivot Computed Items

Moving function calculations require that all displayed Fact cells must be considered in any computations. To consider null fact values in the Pivot Section, see Selecting Pivot Table Elements.

• For Simple and Weighted Moving Averages, if any values are missing/null in the source "Data Column", the Moving Average functionality compensates by subtracting the number of instances of null values from the "Window" divisor. For example, consider the following example for the "Sales" value for "Feb" which is null:
- Use Surface Value Recalculates the values in the visible cells or surface of the pivot rather than the values in the Results section.
- True Computed Item Totals Recalculates break totals so that each total value is equal to the sum of their displayed detail cells. True computed item totals use aggregation according to the specified data function and does not rely on the computed item total formula. If you disable this option, the break total cell values are derived from the computed item formula applied to the detail cell.

Average and Count aggregation Data Functions are not evaluated in True Total mode unless you enable Use Surface Values. If you do not enable Use Surface Values, the Average and Count aggregation are calculated using the count of the underlying Table/Result Section data values instead of the displayed Pivot values.

- Enable Null Facts In Computed Items Enable this option if you want to evaluates a null fact value (an empty cell value) as a zero fact value for non-Moving functions. For Moving Function calculations, where the presence of all displayed Fact cell must be considered in calculations, the following behavior occurs when null fact values are considered as zero. The default option to evaluate and show null values is disabled. If you want to evaluate and show null values for a new Pivot section, then you must the select the this option before creating the Pivot section.
 - For Simple and Weighted Moving Averages, if any values are missing/null in the source "Data Column", the Moving Average functionality compensates by subtracting the number of instances of null values from the "Window" divisor.
 - For Moving Maximum, Moving Minimum, Moving Difference, Moving Sum and Moving Median functions moving functions, if any values are missing/null in the source "Data Column", the missing source values are omitted and the calculation Window size "shrunk" correspondingly.
 - Exponential Moving Averages treat missing source Fact values as having a value of zero.
 - For all Moving Functions, except Exponential Moving Averages, if all the values in the particular calculation Window are null or zero, the Moving Function value is also, by default, null or zero.
 - For all Moving Functions, except Exponential Moving Averages, if the Moving Function calculation Window size exceeds the available number of input fact values, the returned set of Moving Function values should be null.
 - When this option is disabled, the Pivot Sections that receive their data from a relational Query Section, ignore Fact cells with null data when calculating Computed Items. That is, the calculation is simply skipped for the null cell.

Pivot Section Aggregation and Surface Values

This section describes how the Pivot section aggregates and displays data from underlying Table or Results sections. It also discusses the influence of the "Surface Values" property on any final displayed results.

There are three key component parts of the Pivot Section display including: Detail Fact cells, Break Total cells and Computed Items cells. The behavior of these component parts is discussed in various typical Pivot usage scenarios.

Fact Detail Cells

A Fact detail cell is the core numeric data that you slice and dice dimensionally in you analysis A variety of components effect how Fact detail cells behave and what sort of information you see when analyzing them.

Review the following sections for information on:

- Results/Table Section and Pivot Section
- Aggregation Data Functions
- Break Totals
- Computed Items
- Surface Values
- Pivot Section Levels

Results/Table Section and Pivot Section

Fact detail cells are derived from the data set that you retrieve from your database and can be analyzed in both the Results and Pivot section. The Results section is a flat representation of raw, un-aggregated data from a database or external data source as shown below. This example shows sample Sales data for several Cities in the USA.

	City	Units Sold
1	Los Angeles	500
2	Los Angeles	400
3	Los Angeles	100
4	Los Angeles	100
5	Miami	300
6	Miami	200
7	Miami	100
8	Seattle	250
9	Seattle	50
10	Tacoma	100

A Pivot Section is a flexible tabular display of aggregated data in Interactive Reporting that receives its source of data from either a Results or Table Section. As an example, the same Table Section City/Units Sold data from above can be converted to a simple Pivot Section as shown below.

In the Pivot Section, the Units Sold data has been aggregated; that is, there is only one occurrence of each City name - shown in the Pivot Section Row Labels. The Units Sold data for each City has been summed from the entries in the Table Section above to show a single value for each City - these are the Pivot Section Detail Facts.



Aggregation Data Functions

The method by which the underlying Table Section values are aggregated/collected into Pivot Section Facts can be varied by a user. As a default, the aggregation method is to sum the data (as shown in the Pivot example above); that is, the underlying Fact values are added together and the sum displayed as the Pivot Detail Fact cell value. There are several other aggregation Data Functions that can be applied to a Pivot Detail cell.

An aggregation Data Function can be applied to all Pivot Fact cells across a row or to the Fact cells down a column.

Break Totals

A Break Total is a user-enabled Pivot Section row that can display summarized Fact information at the change of a level of Row Label information.

For example, in the above Pivot Section, if you wanted to see the sum of Unit Sales for all Cities, you could add a "Sum" Break Total line at the end of the City Row Label level as shown below:

			Units Sold
		Los Angeles	1,100
		Miami	600
		Seattle	300
		Tacoma	100
reak Total Rows	\rightarrow	Total	2,100

Br

If you intend to use Break Totals in your Pivot Section, be sure to enable surface values instead of using the underlying table data. In this way, Break Totals reflects the aggregation of displayed detail data.

If you use the "Add Totals" menu option to add a Break Total, the Break Total preserves any data function applied to a Fact Column.

Computed Items

A computed item is a user-created column of Fact data in a Pivot Section that performs a userspecified calculation using existing fact data. As an example, below, a Computed Item column has been created to show the "Units Sold" fact amounts multiplied by 2. Each computed item column cell performs its calculations on the data that is used to create the corresponding cells of the fact column on which it is based:

	Units Sold	Units Sold * 2	
Los Angeles	1,100	2,200	Computed Item Column
Miami	600	1,200	
Seattle	300	600	
Tacoma	100	200	
	ſ	•	

Surface Values

Surface Values enable you to aggregate and calculate Pivot Section data based on data displayed in the Pivot Section as opposed to data from the underlying Results or Table Section.

Pivot Section Levels

Pivot Sections can contain multiple levels of Row Label and/or Column Label aggregations. As an example, a State level could be added to the Row Labels of the Pivot Section shown above as follows:

		Units Sold
CA	Los Angeles	1,100
FL	Miami	600
WA	Seattle	300
	Tacoma	100

Pivot Fact Detail Cell Behavior

Review the following sections for information on pivot fact detail cell behavior.

- Simple Pivot Section With Detail Cell Values Only (No Break Totals) —Surface Values Disabled
- Simple Pivot Section With Detail Cell Values Only (No Break Totals)—Surface Values Enabled
- Regular Break Total Line—Surface Values Disabled
- Regular Break Total Line—Surface Values Enabled
- Explicitly Specifying an Aggregation Data Function Across an Entire Break Total Row

Simple Pivot Section With Detail Cell Values Only (No Break Totals) —Surface Values Disabled

The example below shows a simple Pivot Section with detail cell values only (no break totals and surface values have been disabled). (Unless otherwise specified, this Table Section is used as the source data for Pivot Section examples shown throughout this appendix.)

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

Review the Table Section below that shows values of "Units Sold" for some sample cities in the USA. The visible Fact values for each City in the Pivot Section are equivalent to the sum of each City's individual constituent "Units Sold" values in the underlying Table Section.

In other words the Row Label of Los Angeles and the Pivot-displayed "Units Sold" value of 1,100 have been evaluated by summing each "Units Sold" value for Los Angeles in the underlying Table Section (i.e. 500 + 400 + 100 + 100 = 1,100).

100			
	Month		
1	Jan	1.5	
2	Feb		
3	Mar		
4	Apr		

In addition to the sum method for aggregation in the Pivot Section, you can explicitly specify a number of other aggregation methods, or Data Functions, in which to collect and represent the underlying Table Section data as Facts in the Pivot Section. These Data Functions include:

- Average The displayed Pivot Fact value is the average (mean) value of the Fact values for this Pivot Row Label in the underlying Table Section (i.e. the sum of the corresponding values in the Table Section divided by the number of occurrences (rows) of these values).
- **Count** The displayed Pivot Fact value is the sum of the number of occurrences of Fact values for this Row Label in the underlying Table Section.
- Maximum The displayed Pivot Fact value is the maximum value of all Fact occurrences in the underlying Table Section for this Row Label.
- Minimum The displayed Pivot Fact value is the minimum value of all Fact occurrences in the underlying Table Section for this Row Label.

The effect of applying the above different aggregation Data Functions on Pivot Fact columns can be illustrated in the following Pivot Section:

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

The "Units Sold" Fact column is evaluated using a Sum Data Function as above, whereas the "Average Of Units Sold" column uses the Average aggregation Data Function.

That is, the "Average Of Units Sold" Fact value for Los Angeles is evaluated according to the following formula:

The Sum of "Units Sold" values for Los Angeles in the underlying Table Section divided by the number of occurrences (rows) of "Units Sold" values for Los Angeles in the underlying Table Section

or

(500 + 400 + 100 + 100) / 4 = 1100/4 = 275

The "Count Of Units Sold" value for Los Angeles is 4, which is evaluated by using the Count aggregation Data Function. It is equal to the count or number of individual "Units Sold" row occurrences for Los Angeles in the underlying Table Section

The "Max Of Units Sold" for Los Angeles is 500, which is evaluated using the Maximum aggregation Data Function. This function returns the maximum single "Units Sold" value from all "Units Sold" values for Los Angeles in the underlying Table Section.

The "Min Of Units Sold" for Los Angeles is 100, which is evaluated using the Minimum aggregation Data Function. This returns the minimum single "Units Sold" value from all "Units Sold" values for Los Angeles in the underlying Table Section.

Simple Pivot Section With Detail Cell Values Only (No Break Totals)—Surface Values Enabled

If you enable the "Surface Values" property (by selecting *Use Surface Values* from the Pivot Option dialog. "Selecting Pivot Table Elements" on page 174) the Pivot Section display from above changes to:

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

The Fact columns have now been recalculated to use what would be a displayed (i.e. surface) summed, aggregated value for each City as a source of input data (i.e. the values that are actually displayed in the "Units Sold" column).

As a result, the "Average Of Units Sold" for Seattle references the sum of underlying "Units Sold" values for Seattle (that is, the same value that is displayed for Seattle in the "Units Sold" column). This value is 300 and there is only one value occurrence displayed for Seattle. Consequently, the "Average Of Units Sold for Seattle" is 300 divided by 1 which is 300.

Likewise, the "Count Of Units Sold" column uses the displayed "Units Sold" value for each City as a basis for calculation. Since there is always only one displayed value per City, the "Count Of Units Sold" for each City always has a value of 1.

The "Maximum Of Units Sold" and "Minimum Of Units Sold" column cell values are exactly the same as their equivalent cells in the "Units Sold" column. Since there is only one cell value available per City, by definition that value is always the Maximum or Minimum value available. Since the lowest level of data displayed in a Pivot section is always aggregated from the underlying Table section values, it is recommended that you disable surface values when applying a sum or other data function.

Regular Break Total Line–Surface Values Disabled

At any time, a "regular" Break Total line can be added to a Pivot Section by selecting "Add Totals" from the Row Labels right-click speed menu (or the Pivot menu in the Section toolbar).

	Units Sold	Average Of Units Sold	Count Of Units Sold	Max Of Units Sold	Min Of Units Sold
Los Angeles	1,100	275	4	500	100
Miami	600	200	3	300	100
Seattle	300	150	2	250	50
Tacoma	100	100	1	100	100
Total	2,100	210	10	500	50

When a Break Total row is added, it is labeled "Total" as shown below:

Notice that, the regular Break Total row performs aggregation of each Break Total cell value from the underlying Table Section data by using the aggregation Data Function that has been specified for that Fact column.

For example, the "Average Of Units Sold" Fact column Break Total cell has a value of 2100. This value has been evaluated by using the Average Data Function (which is the Data Function that has been specified already for that column) on the underlying Table Section data. The equation is:

Sum all "Units Sold" values for all Cities in the underlying Table Section and divide by the total number of occurrences of those values.

(500 + 400 + 100 + 100 + 300 + 200 + 100 + 250 + 50 + 100) / 10= 2100/10 = 210

Likewise, The "Count Of Units Sold" Break Total value refers to the underlying Table Section values and returns the total number of occurrences of "Units Sold" values there for all Cities. This amount equals 10.

The "Max Of Units Sold" Break Total value refers to the underlying Table Section and returns the single maximum "Units Sold" amount for all Cities. This amount equals 500.

The "Min Of Units Sold" Break Total value refers to the underlying Table Section and returns the single minimum "Units Sold" amount for all Cities. This amount equals 50.

Regular Break Total Line–Surface Values Enabled

If the Surface Value property is enabled, the previous Pivot Section, with regular Break Total line, is displayed as:

	Units Sold	Average Of Units Sold	Count Of Units Sold	Max Of Units Sold	Min Of Units Sold
Los Angeles	1,100	1,100	1	1,100	1,100
Miami	600	600	1	600	600
Seattle	300	300	1	300	300
Tacoma	100	100	1	100	100
Total	2,100	525	4	1,100	100

All cell values for each column in the Break Total line have been updated to show the aggregation of all Fact values that are displayed in that column using the aggregation data function specified for that column.

The Break Total value for the "Units Sold" column is shown to be 2100 which is the sum (i.e. the aggregation data function used for that column) of all displayed Fact detail cell values in the column (this value, 2100, happens to be the same as the value shown when surface values were disabled and the underlying Table Section values were used, because of the mathematical nature of a sum operation).

For the "Average Of Units Sold" column – the Average aggregation data Function is used for the Fact detail cells. As a result, this same data function is used to evaluate the Break Total cell value. The value of 525 is evaluated to be the sum of the displayed fact detail cells (= 2,100 = 1,100 + 600 + 300 + 100) divided by the number of occurrences of displayed fact detail cell values (4).

or

The "Count Of Units Sold" column uses the Count data function. As a result the Break Total value uses the Count data function operating against the number of occurrences of displayed Fact detail cells. Since there are four fact detail cell values in this column, the Break Total value is 4.

The "Max Of Units Sold" column uses the Maximum data function. As a result the Break Total value uses the Maximum data function to evaluate the largest number within the displayed Fact detail cells. This number is 1,100 which is the value displayed in the Break Total cell for this column.

The "Min Of Units Sold" column uses the Minimum data function. As a result, the Break Total value uses the Minimum data function to evaluate the largest number within the displayed Fact detail cells. This number is 1,100 which is the value displayed in the Break Total cell for this column.

Explicitly Specifying an Aggregation Data Function Across an Entire Break Total Row

With a regular Break Total line, as described above, the aggregation Data Function that is applied to each Break Total cell is the aggregation Data Function that is already applied to each Fact column.

However, you can explicitly specify an aggregation data function to apply to all cells on a Break Total row. To do this, select the entire Break Total row (by pressing the Alt and mouse left-click buttons simultaneously) and then select the appropriate data function from the mouse rightclick speed menu or the Pivot menu.

As an example, the row entitled "Total – Average" below, is a regular Break Total line which has had the Average aggregation Data Function explicitly applied to all of its cells (the row entitled "Total" is the regular Break Total row from above):

	Units Sold	Average Of Units Sold	Count Of Units Sold	Max Of Units Sold	Min Of Units Sold
Los Angeles	1,100	275	4	500	100
Miami	600	200	3	300	100
Seattle	300	150	2	250	50
Tacoma	100	100	1	100	100
Total	2,100	210	10	500	50
Total - Average	210	210	210	210	210

The "Total - Average" calculates the average value of "Units Sold" from the underlying Table Section for all cells, that is, the Total - Average is the aggregated sum of all underlying "Units Sold" values divided by the number of occurrences of "Units Sold" values.

You can see how this concept works across all cells on a Break Total row by reviewing the example below. This example Break Total rows named "Total Sum", "Total - Count", "Total - Max", "Total - Min" which have the Sum, Count, Maximum and Minimum Data Functions respectively forced across their rows:

	Units Sold	Average Of Units Sold	Count Of Units Sold	Max Of Units Sold	Min Of Units Sold
Los Angeles	1,100	275	4	500	100
Miami	600	200	3	300	100
Seattle	300	150	2	250	50
Tacoma	100	100	1	100	100
Total	2,100	210	10	500	50
Total - Sum	2,100	2,100	2,100	2,100	2,100
Total - Average	210	210	210	210	210
Total - Count	10	10	10	10	10
Total - Max	500	500	500	500	500
Total - Min	50	50	50	50	50

You can override any cell in a Break Total row which has a forced data function. To override the Break Total Data Function, you select the Fact cell in that column and then select the required data function from the speed-menu. As an example, the "Count Of Units Sold" column from above has been selected and the Count data function has been explicitly assigned to this column. The resulting Pivot display is nearly identical to the example above except the entire "Count Of Units Sold" column has its Fact Detail and Break Total cells evaluated by the Count data function as described above:

	Units Sold	Average Of Units Sold	Count Of Units Sold	Max Of Units Sold	Min Of Units Sold
Los Angeles	1,100	275	4	500	100
Miami	600	200	3	300	100
Seattle	300	150	2	250	50
Tacoma	100	100	(1)	100	100
Total	2,100	210	/ 10	500	50
Total - Sum	Z,100	2,100	10	2,100	2,100
Total - Average	210	210	10	210	210
Total - Count	10	10	10	10	10
Total - Max	500	500	\ 10	/ 500	500
Total - Min	50	50	10	50	50

Column Data Function (Count) - Overrides Break Total Rrow Data Functions

Pivot Section Computed Items

Computed items, are Fact columns that are created by the user and which use existing Table data (or displayed Pivot data if Surface Values are enabled) as a basis for calculation. Each Pivot computed item is represented as a new Fact column in the Pivot display and Interactive Reporting provides a multitude of built-in data manipulation and evaluation functions for the user that facilitate the construction of computed item columns.

Here are some key points to remember when working with computed items in the Pivot Section:

- To ensure consistency, the source of data for computed items should be governed by the type of computed item function rather than being governed "globally" by the Surface Values property.
- Computed Item functions that operate on aggregated data such as "Avg" and "ColMax" should always receive their input data from the underlying Results or Table data.
- Computed Item functions that expect a single value as input, such as Sqrt, should receive their data from the displayed surface values.
- Computed Items that refer directly to other Pivot Section columns without wrapping in a Function reference (such as "Units Sold * 2" above, should receive this data always from displayed surface values.

Computed Item Detail Cell Values

A very simple computed item can be created using a Label dimension and the values from the Fact Detail cells. As an example of computed item display, consider the original simple "Units Sold" Pivot display below:

Units Sold
1,100
600
300
100

Several computed item columns can be added to this Pivot using the "Units Sold" Facts as a source of input data. These computed items are shown below as new columns to the right of the "Units Sold" column (surface values have been disabled in this example):

	Units Sold	Units Sold * 2	Sqrt Units Sold	ColMax Units Sold	Average Units Sold
Los Angeles	1,100	2,200	33.16625	500	210
Miami	600	1,200	24.4949	500	210
Seattle	300	600	17.32051	500	210
Tacoma	100	200	10	500	210

These new Computed Item columns are evaluated as follows:

Units Sold * 2 = Units Sold values multiplied by 2 (i.e. computed item calculation: formula is: Units_Sold * 2)

Sqrt Units Sold = Square root of Units Sold values (i.e. computed item calculation formula: Sqrt (Units_Sold))

ColMax Units Sold = Maximum value of Units Sold in a column (i.e. computed item calculation formula: ColMax (Units_Sold))

Average Units Sold = Average value of Units Sold (i.e. computed item calculation formula: Avg (Units_Sold))

Computed Item Detail Cell Values - Without Surface Values

In general, when surface values are disabled, computed item columns receive their source "Units Sold". Fact values from the underlying Table Section. However, the specific values from the underlying Table Section that the computed item uses depends on the nature of the its calculation.

A calculation or computed item function that expects a single value as input uses the underlying values that correspond to the Pivot computed item column cell to be calculated, aggregated according to the aggregation data function applied to the computed item column. In the previous example, the "Units Sold * 2" column implicitly requires a single cell value. If it is used in a computed item column with an aggregation data function of Sum, all underlying Table Section values corresponding to this Pivot cell are summed and then multiplied by 2.

If you review the "Units * 2" value for Los Angeles, you see that it is evaluated by summing the underlying Table occurrences for Los Angeles and then multiplying this sum by 2.

The ColMax computed item function operates on a range of data as opposed to a single value.

The Avg computed item function operates on the summed, aggregated values corresponding to a Pivot cell. In the case of Los Angeles, this would be the sum of underlying values for Los Angeles divided by the number of occurrences of these values.

This behavior can be most easily seen from the "ColMax Units Sold" column which shows a value of 500 for each cell. The ColMax function interrogates the range of values in an entire column and returns the maximum value found: 500. This is the maximum cell value in the entire "Units Sold" column in the underlying Table Section (compared to 1,100 which is the maximum "Units Sold" value shown in the Pivot display).

Computed Item Detail Values - With Surface Values Enabled

Compare the above Pivot Section display with the following example. In this example, surface values have been enabled.

	Units Sold	Units Sold * 2	Sqrt Units Sold	ColMax Units Sold	Average Units Sold
Los Angeles	1,100	2,200	33.16625	1,100	525
Miami	600	1,200	24.4949	1,100	525
Seattle	300	600	17.32051	1,100	525
Tacoma	100	200	10	1,100	525

As you can see, the "ColMax Units Sold" and "Average Units Sold" Fact columns values have changed. With surface values enabled, the ColMax function refers to the displayed "Units Sold" Fact column for input data and the highest value in this column is 1,100 which is the value that is displayed in the "ColMax Units Sold" column above.

The Avg Computed Item function which is used to evaluate the "Average Units Sold" column takes its input from the sum of the displayed "Units Sold" Fact column values and divides this by the number of displayed occurrences of values in the "Units Sold" column - this is 2100/4 which equals 525 - the value displayed in the "Average Units Sold" column.

In general when you enable surface values for Pivot Section computed items, they operate on the summed, aggregated values from the underlying Table Section.

Computed Items With Break Totals - No Surface Values

If a Break Total row is added to the Pivot by using the "Add Totals" menu item described above, the following Pivot display results:

	Units Sold	Units Sold * 2	Sqrt Units Sold	ColMax Units Sold	Average Units Sold
Los Angeles	1,100	2,200	33.16625	500	210
Miami	600	1,200	24.4949	500	210
Seattle	300	600	17.32051	500	210
Tacoma	100	200	10	500	210
Total	2,100	4,200	45.82576	500	210

Since the surface values have been disabled, the values in the Break Total ("Total") cells for each Computed Item column are evaluated by applying the computed item calculation formula to the underlying aggregated Table Section "Units Sold" values summed for all Cities.

Computed Items With Break Totals - Surface Values Enabled

If a Break Total row is added to the Pivot by using the "Add Totals" menu and surface values are enabled, the following Pivot display results:

	Units Sold	Units Sold * 2	Sqrt Units Sold	ColMax Units Sold	Average Units Sold
Los Angeles	1,100	2,200	33.16625	1,100	525
Miami	600	1,200	24.4949	1,100	525
Seattle	300	600	17.32051	1,100	525
Tacoma	100	200	10	1,100	525
Total	2,100	4,200	45.82576	1,100	525

With surface values enabled, the various Break Total cell values are evaluated using the displayed values of the "Units Sold" Fact column. For example, the "ColMax Units Sold" total line value has changed (from 500) to 1,100 since the maximum value displayed in the "Units Sold" column is 1,100. The "Average Units Sold" total line value has changed (from 210) to 525 since the displayed sum of "Units Sold" is 2,100 and the displayed number of Cities is 4 - leading to an average calculation of 2,100/4 = 525.

In either case, whether surface values are enabled or disabled, the computed item Break Total values are calculated by applying the computed item calculation on the Break Total value for the column on which the computed item calculation is based.

There exists no facility whereby computed item Break Totals are evaluated by summing (or some other aggregate data function) its constituent displayed Detail cell values. If such a facility were to exist, the "Total" cell for the "ColMax Units Sold" column above would be 4,400 (= 1,100 + 1,100 + 1,100 + 1,100) i.e. the sum of all Detail Fact values in this column.

Pivot Menu Command Reference

Table 35 provides a quick reference to the commands available on the Pivot menu and lists any related shortcuts.

Dommand	Description	Keyboard	Shortcut
Command	Description	Snortcut	wenu
Add Selected Items	Adds the selected item as a Column Label, Row Label, or Fact.		~
Remove Selected Items	Removes the selected item.	[Del]	~
Modify	Modifies the selected computed item.	[Ctrl+M]	~
Sort	Reorders the selected item by labels, by values, ascending, or descending.		
Add Totals	Adds the selected item to the Measures pane.		~
Add Computed Item	Enables you to add a new data item derived from calculations performed on an existing item.		~
Add Cume	Adds cumulative totals to break totals by dimension and restarts them at each dimensional grouping.		~
Data Function	Applies a prebuilt data function to the selected item.		~
Drill Anywhere	Enables you to drill to any item.		~
Drillup Returns the original view of data that you drilled.			~
Focus On Items	Updates the pivot table to include only the selected data.		~
Hide Items	Hides the selected item.		~
Show Hidden Items	Restores the selected hidden item.		~
Show All Items	Updates the pivot table to include all items.		~
Group Items	Groups the selected dimensions.	[Ctrl+G]	
Ungroup Items	Ungroups the selected dimension.	[Ctrl+U]	
Restore Name	Restores the original name of a renamed item.		~
Refresh Data	Updates the data according to the selected option. Select between After Process, When Section Displayed, Manually, or Refresh Now.		
Pivot Options	Enable surface values, true computed totals, and null facts in computed items.		

Table 35 Pivot Menu Commands

Charting Data

9

This chapter explains how to use Interactive Reporting Studio charting features to perform interactive analysis of your data in a graphic format.

In This Chapter	Chart Section	200
	Charting Basics	200
	Creating Charts	205
	Selecting a Chart Type	205
	Working with Two-dimensional Charts	206
	Working with Multidimensional Charts	208
	Manipulating Chart Data	213
	Chart Scrolling and Scaling	222
	Customizing Chart Properties	227
	Chart Menu Command Reference.	237

Chart Section

The Chart section enables you to see meaningful summaries of your data. Graphic snapshots help you recognize patterns, trends, and other relationships that might not be apparent in columns and rows of tabular data.

The Chart section opens with an initial plot area for the chart. Because chart construction and manipulation is managed with the Data Layout, plotting, viewing, and reviewing are easy and intuitive. Interactive Reporting Studio charts respond dynamically to your commands. When you make a change in a charted item, you see your chart instantly redrawn to reflect the change. Experimenting with different combinations of data can be surprisingly informative.



Charting Basics

A chart is a graphic representation of data. Except for pie charts, all charts plot data with reference to a horizontal x axis and vertical y axis. Multidimensional charts sometimes plot data on an additional axis. A pie chart uses the metaphor of the pie as a whole to delineate the relative values of the parts or slices.

In , you construct a chart by dragging Request items from the Catalog pane to one of the *Data Layouts*. At least one y axis item must populate an Data Layout to plot a usable chart.

Chart Terminology

The following table defines the chart terminology used in Interactive Reporting:

Term	Definition
Axes	Straight lines on a chart that provide a framework for measurement and reference. Typically, the x axis and z axis are used to display label items, and the y axis shows values or facts (measurable items), such as units and amounts.
Values	Graphic indicators that represent data. bar charts display values in either vertical or horizontal bars. Pie charts use wedge-shaped slices to represent values.
Plot Area	The area bounded by the axes is called the plot area. In the case of the pie chart, the plot area is defined by a circle representing the totality of all data items.
Planes	In all charts (except pie charts), planes provide background and graphed reference for charted values. Planes define horizontal, vertical, and background fields for a chart.
Legend	An information box containing color-keyed labels used to identify different data values represented on a chart.
Grid lines	Straight horizontal and vertical lines arranged in scaled increments that provide calibrated guidelines for value interpretation.
Bar	A linear measure of a value used in bar charts.
Slice	A spacial measure of a value used in pie charts.
Chart top title	Text description related to the entire chart.
Chart subtitle	Additional text used to describe the chart.
Y axis label	Text description of the vertical quantity axis.
Y axis values	Text or numbers which indicate specific values along the y axis
X axis label	Text description of the horizontal quantity axis.
X axis values	Text or numbers which indicate specific values along the x axis.
Z axis label	Text description of the depth axis.
Z axis values	Text or number which indicate specific values along the z axis
Chart graphic	Central chart picture which contains Chart bars, background and also Chart axes.
Chart graphic border	Border which controls the position of the Chart's central bars, background and axes labels. The border can be resized and repositioned.
Chart border	Border which controls the general position of all other Chart components not controlled by the Chart graphic border. This border can be resized and repositioned.
Inserted text string	Optional text strings that can be placed anywhere within the Chart border.
Right y axis label	Optional label that can be placed to the right of the right-hand vertical edger of the Chart graphic.

Table 36 Chart Terminology (Continued)

Term	Definition
Right y axis values	Optional values that can be displayed on the right hand vertical edge of the Chart graphic
Bar value	Optional text which represents numeric value of the bar.

Table 37 Chart Terminology (Pie Specific)

Term	Definition
Pie slice	Individual y axis value representation for current x axis item.
Pie slice label	Text to identify x axis item.
Line to label	Line from pie slice label to pie slice to aid in slice identification.

Understanding Chart Dimensions

To understand the differences among charts, you have to distinguish between dimensions in space and dimensions of data. The two dimensions are distinct.

Space can be represented as three dimensions along the x, y, and z axes shown in Figure 2.



Figure 2 Data Layout Corresponding to the X, Y and Z axes in the Chart Area

Data can either be represented in a two-dimensional or a three-dimensional (2-D or 3-D) space. In two dimensions, data is represented along the x axis and y-axis only, In three dimensions, data is projected back along the z axis also.

Two dimensions of data must be represented in 2-D space. At least three dimensions of data are necessary to use the third spatial dimension Stack, Cluster or Depth (z axis). But three or more dimensions of data can be represented in 2-D space. For example, cluster and stack represent data categories in two spatial dimensions (X axis and Y axis only).



Using the Chart Data Layout

The Chart Section opens with an initial plot area for the chart. Because you manage chart construction and manipulation is with the Chart Data Layout plotting, viewing and reviewing are easy and intuitive.

You construct a chart by dragging items from the Catalog pane to a Data Layout pane. At least one items must populate the Data Layout to plot a usable chart.

The Data Layout consists of the following items:

 Table 38
 Chart Data Layout

Data Layout Pane	Description
X Axis Slice	The X- Axis is used for those items that you want to place on the x axis, which is a straight line on the chart. It is used as a qualitative data label for categorizing information. To place items on the x axis, use the X-Axis pane.
Stack Cluster Depth	The Stack Cluster or Depth pane represent the third dimension of data, that is, the z axis that projects out toward you or a location in space. This axis can either be qualitative or quantitative. For a Stack each dimension of data of the charts is represented by only one bar. This bar consists of as many components as the data file has data rows, with the data from each row stacked onto the previous row. For example, a single bar can represent the amount of sales for CD-ROM drives in one year on top of a bar representing sales for other years. You can stack the bar charts vertically or horizontally. By stacking items and assigning a different color to each item, you can display trends among comparable or related items, or emphasize visually a sum of several indicators. For a Cluster , data extended in the third dimension is shown as clusters displayed in the foreground. This category creates a vertical column (and only a vertical column) for each data value. If the chart is showing multiple data series, the values are grouped based upon the category value. For example, use clustered bars to compare stores of different types. Alternatively, cluster bars can be used to compare two different values items, such as Amount of Sales and Units Sold. For the Depth , data extends the length of the chart along the z-axis.
Fact Fact (Stack) Fact (Depth)	The Facts category indicates height in the coordinate system. It is used as a quantitative label as a way of categorizing information on the y axis. For the Fact (Stack) pane each dimension of numeric data is represented by only one bar. and shows the grouping along the y axis. This bar consists of as many components as the data file has numeric rows, with the numeric data from each row stacked onto the previous row. For example, a single bar can represent the amount of sales for CD-ROM drives in one year on top of a bar representing sales for other years. You can stack the bar charts vertically or horizontally. By stacking items and assigning a different color to each item, you can display trends among comparable or related items, or emphasize visually a sum of several indicators. For the Fact (Depth) pane, numeric data extends the length of the chart along the z-axis.

Creating Charts

Interactive charts consists of two layout elements: graphical elements (for example, chart bars or pie slices) and axis labels. When you assign Request items to the Data Layout, they become values or labels in your chart.

The instructions below are a starting point for building charts. As you use and become familiar with the Chart section, you learn ways to create the exact type of chart that fits your needs.

- To create a chart using the Chart Data Layout:
- 1 Select Insert > New Chart to create a new Chart section.
- 2 If the Data Layout is not visible, click **Data Layout** on the Section title bar to display the Data Layout.
- **3** Drag each line item to be included in the chart from the Catalog pane to an Data Layout:
 - Drag values or facts (such as Units or Amount) in to the y pane in the Data Layout to create bar charts, pie charts or ribbon charts.
 - Drag a label item (such as Name, Product, or State) into the x pane in the Data Layout to create a 2-D chart.
 - Drag a label item to the y pane in Data Layout to add a third dimension to your chart.
- 4 Select a chart type from the Chart drop-down list box.
- **Note:** You can select and drag multiple items to the same Data Layout to add multiple values or labels for analysis. Items are hierarchically ordered in the sequence in which they are displayed in the Data Layout.

Selecting a Chart Type

You can select a different chart type to show data in different views.

- To select a chart type, select Format > Chart Type and select a chart type from the menu that is displayed.
 - Vertical Bar Horizontal Bar Vigitical Stack Bar Horizontal Stack Bar Vertical Queter Bar De Stacked Area Area Line Bibbon Bar Line

Chart types include:

- Vertical bar
- Horizontal bar
- Vertical Stacked bar

- Horizontal Stacked bar
- Vertical Cluster bar
- Pie
- Stacked Area
- Area

Working with Two-dimensional Charts

Pie and bar charts (of the non-stacked variety) lend themselves well to representing two dimensions of data. For example, imagine charting the amount of sales by product type. In pie charts, the two dimensions are represented by slices of a pie. In bar charts, the data is represented by bars along the x axis and y axis).

Using Pie Charts to Analyze Data

Of all charts, the pie chart is the easiest to understand. Pieces (slices) of the pie are drawn to represent the relative value of a measurable item category to the whole. Pie charts represent additional dimensions of data by further subdividing the pie.

In a Pie chart, Request items placed in the x axis represent itemized slices of the pie. Request items placed in the y pane of the Data Layout define the quantitative whole of the pie.



Creating a Pie Chart

- To create a pie chart:
- 1 From the Chart drop-down list box, select Pie.
- 2 Drag a value from the Catalog pane to the y pane in the Data Layout.

A Pie chart without slices is displayed.

3 Drag one or more label items from the Catalog pane to the a pane in the Data Layout.

The Pie chart is differentiated to reflect subcategories. A legend depicting details of the selection is displayed.

Note: Since pie charts plot data using only two axes, the z pane is disabled in the Data Layout when creating a Pie chart.

Positioning Pie Slices

You can pull individual pie slices out of the pie chart.

To toggle the position of a pie slice, select a slice of the pie and choose Pull Out Slice on the shortcut menu.

A check mark is displayed on the shortcut menu next to Pull Out Slice to indicate that this feature is active. Select this option again to clear the check mark and restore the pie slice to its original position.

Showing Positive and Negative Values

Pie slices show positive values by default.

To toggle the display of negative and positive values, select a slice of the pie and select Show Negative Values on the shortcut menu.

A check mark is displayed next to Show Negative Values to indicate that negative values are shown. Select this option again to clear the check mark and show positive values.

Showing Pie Percentages

To toggle the display of each pie slice value as a percentage, select a slice of the pie and select Show Pie Percentages on the shortcut menu.

A check mark is displayed next to Show Pie Percentages to indicate that percentages are displayed on the chart (in parentheses next to the pie chart label). Select this option again to clear the check mark and remove the percentages from view.

Adding Lines to Labels

- To add a pointer line between an individual pie slice and its label, select the pie slice to which you want to add a pointer and select Format > Line To Label.
- **Note:** You cannot change a Chart label name when a Pie Chart contains 0 or null values. To add a label name in this case, filter out or hide the 0 or null values in the underlying Results or Pivot Section.

Rotating Pie Charts

- > To rotate the perspective angle or elevation of a pie chart:
- **1** Click **Rotate** on the shortcut menu.

The rotate icon is displayed on the pie chart.

2 Click the rotate icon and move the dotted line to a new location.

When you release the mouse button, the chart is redrawn to reflect the adjusted perspective. If you cannot drag the rotate icon in a certain direction, the chart has reached its farthest possible rotation in that direction.

Using Two-dimensional Bar Charts to Analyze Data

Bar charts are the most common type of business chart and are especially useful for comparative analysis when you want to focus on comparing values and place less emphasis on time. Use a bar chart to illustrate comparisons among individual items.

Two-dimensional bar charts are plotted using a single item in each of the x pane and y pane in the Data Layout. The z pane is not populated in 2-D bar charts.

- To create a two-dimensional bar chart:
- **1** Select a bar chart format from the Chart drop-down list box.

The default chart format is Vertical Bar.

2 Drag a label item from the Catalog pane to the the Data Layout.

Data labels are displayed on the horizontal axis in the Chart area.

3 Drag a value from the Catalog pane to the y pane the Data Layout.

A chart is plotted that summarizes the selected value on the y axis as it relates to the subcategories of the label item x axis.

automatically scales the data represented on the y axis and adds appropriate labels. The Legend provides an index of label information with a coordinated color scheme.

To use a different 2-D chart format, select another 2-D chart from the Chart drop-down list box.

Working with Multidimensional Charts

Frequently, you want to represent more than two dimensions of data at a time. For example, you may want to see how the sales of product types break down by years or quarter. There are numerous ways to chart three or more dimensions of data. You can project data into the third dimension of space. You can also represent the data in two spatial dimensions.

About the 3-D View

By default, Interactive Reporting imparts a 3-D look to your chart objects. These objects are displayed in the chart space as 3-D objects with depth. That does not mean that you are plotting three dimensions of data or using three dimensions of space to represent data. It is simply a visual effect that can be turned off.

Note: If you turn off 3-D View, you cannot view charts that use a third dimension in space.

To toggle 3-D View, select Format > 3-D View.

A check mark is displayed next to the 3-D View option to indicate it is active. Select this option again to clear the check mark and turn off 3-D view.

Note: You can also select to view objects in 3-D using the Properties dialog box. For more information, see "Customizing Chart Properties" on page 227.

Creating Three-dimensional Bar Charts

You can add more information to your bar chart by adding an additional item or items to the z pane in the Data Layout. Using multidimensional charts, you can show various relationships between three or more items in easy-to-understand bar chart formats.

Interactive Reporting plots the added data in rows that extend back along the z axis of the chart.

- To create a three-dimensional bar chart:
- **1** Select a bar chart format from the Chart drop-down list box.

The default chart format is Vertical Bar.

- 2 Drag a label item from the Catalog pane to the x pane in the Data Layout.
- 3 Drag a label item from the Catalog pane to the z pane in the Data Layout.
- **4** Drag a value from the Catalog pane to the y pane in the Data Layout.

A chart is plotted that summarizes the selected value y oabe as it relates to the subcategories of the label items (x axis and z axis).

Understanding Clustered Bar Charts

You can change your chart perspective so that the z axis data extended in the third dimension is shown as clusters displayed in the foreground. This charting option is useful when the z axis bars are hard to distinguish in standard bar formats.

You can use clustered bar charts to juxtapose categories in one label item. For example, use clustered bars to compare stores of different types. Alternatively, cluster bars can be used to compare two different value items, such as Amount of Sales and Units Sold.

Note: You can only display clustered bar charts in vertical format.

- To cluster bars representing divisions in label items (clustered on the z axis):
- **1** Select Vertical Cluster Bar from the Chart Type drop-down list box.
- 2 Drag a fact item from the Catalog pane to the y pane and label items to the z axis and x axis panes in the Data Layout.

- > To cluster bars representing two different value items (clustered on the y axis):
- 1 Select Vertical Cluster Bar from the Chart drop-down list box.
- 2 Drag two fact items to the y axis and a label item to the x axis.



Understanding Stacked Bar Charts

Another way to represent the third dimension of data is through stacking. In this way, a single bar on the chart can show data for more than one category of data. For example, a single bar can represent the amount of sales for CD-ROM drives in one year on top of a bar representing sales for other years. You can stack the bar charts vertically or horizontally.



Stacked bar charts show the relationship of parts to the whole. Stacking techniques differ depending on whether you are representing divisions within data label categories or stacking two separate numeric categories.

Stacked bar charts offer similar complexity to clustered bar charts by adding together component value items within chart bars or areas. By stacking items and assigning a different color to each item, you can effectively display trends among comparable or related items, or visually emphasize a sum of several indicators.

To create a Stacked bar chart, you need more than one Request item in the Y-Facts in the Data Layout. Each value item adds a segment to the length of the bar.

Understanding Area Charts

Area charts are essentially bar charts with the discontinuous breaks removed along the horizontal axis. Data is not broken into discrete bars but is displayed in a continuous ebb and flow as defined against the Y-axis. Consequently, area charts are particularly useful for emphasizing the magnitude of change over time. In addition, area charts can be used for the same purposes as bar charts.

Because area charts do not break data along the horizontal axis, they are most useful for charting three dimensions of data. The z pane should be used to either project data into a third spatial dimension, or to stack two categories of data in a stacked area chart.

Creating an Area Chart

- To create an area chart:
- **1** Select **Area** from the Chart drop-down list box.
- **2** Drag a value item from the Catalog pane to y pane in the Data Layout.
- **3** Drag a label item from the Catalog pane to the z pane in the Data Layout.
- 4 Select Legend On Z from the Legend drop-down list box.

Setting the legend on the z axis is properly distributes color.



Creating a Stacked Area Chart

- To create a stacked area chart:
- **1** Select **Stacked Area** from the Chart drop-down list box.
- 2 Drag a value from the Catalog pane to the y pane in the Data Layout.

Understanding Ribbon Charts

A ribbon chart is very similar to a line chart but with a few visual differences. In a ribbon chart, items in the y axis determine the height of the line, and items in the x axis itemize the line sections. You can create multiple lines by adding items to the z pane.

- To create a ribbon chart:
- **1** Select **Ribbon** from the Chart drop-down list box.
- 2 Drag a value item from the Catalog pane to the y pane in the Data Layout.
- **3** Drag a label item from the Catalog pane to x pane and one or more label items to the y pane in the Data Layout.



Understanding Line Charts

Line charts show trends in data at equal intervals and are effective for comparing highs and lows in a continuum. In a line chart, items in the y axis determines the height of the line, and items in x axis itemize the line sections. You can create multiple lines by adding items to z axis.

Line charts have one advantage over bar charts. They do not enable one set of data to obstruct the representation of another. Since lines are *thin* compared to bars, the data displayed in the front does not block out the data behind.

As a result, data that is not easily represented in bar or area charts work well in line charts. Many more dimensions of data can be superimposed without impairing the chart's effectiveness.

- **Note:** A line chart can have two different looks depending on if the chart was switched from a stacked or non stacked chart. If a line chart was switched from a stacked chart, the z axis items on the chart are stacked. If a line chart was switched from a non stacked chart, the z axis items are not stacked and are displayed as originally created.
 - > To create a standard line charts y axis for values):
 - 1 Select Line from the Chart drop-down list box.
 - 2 Drag a value item from the Catalog pane to the y pane in the Data Layout.
 - 3 Drag label items from the Catalog pane to the x pane and the yain the Data Layout.
 - 4 Select Legend On Facts (Depth) from the Legend drop-down list box to distribute colors along the z axis.



Understanding Combination Charts

Combination charts combine some of the strengths of bar charts with the advantages of line charts. Solid bars can be used for the most important data against which other dimensions are represented in lines. In this way, emphasis is given to a portion of data based on its importance. A combination chart is especially useful for comparing two numeric values, such as amount and units of sales.

- > To create a combination chart that compares values:
- **1** Select **Bar-Line** from the Chart drop-down list box.
- 2 Drag two values from the Catalog pane to the y pane in the Data Layout.
- 3 Drag label items from the Catalog pane to x pane and z pane in the Data Layout.



- > To create a combination chart that compares categories within a label:
- **1** Select **Bar-Line** from the Chart drop-down list box.
- 2 Drag a value item from the Catalog pane to the z pane in the Data Layout.
- 3 Drag one or more value items from the Catalog pane to the y pane in the Data Layout.
- 4 Drag a label to the x pane in the Data Layout.
- **Note:** A combination chart is most effective when the y pane contains only two value items. It represents one value as bars and the other value as a line. When more than two values are present, the chart alternates between bars and lines in depicting the values (1st, 3rd, 5th ... items are bars; 2nd, 4th, 6th ... items are lines).

Manipulating Chart Data

Interactive Reporting offers a number of ways to manipulate the data in your chart for better analysis. Review the following sections for information on:

- Using Different Scales to Compare Related Values
- Using Data Functions in Charts
- Adding Computed Items
- Sorting Chart Items
- Creating Pivot Tables from Charts
- Drilling into Charts
- Hiding and Focusing on Charted Data

Using Different Scales to Compare Related Values

To chart comparison values or to combine two related indicators on the same chart, you may need to compensate for different numeric scales. For line, clustered bar, and combination charts, you can use a second Y-axis to represent values on a *scalar function* that differs from the scale of the first y axis.

For example, you might like to chart the sales of your two fastest growing product lines together to get an idea of how business is growing. However, while the growth rates are similar, the two product lines may sell at entirely different volumes. The chart does not provide much comparative information because each line needs to be charted at a different scale.By using different scales for the two y axes, you can correctly scale each value for the most effective presentation of the data.

To use a different scale for a value on the second y axis, double-click the one of the two values in the y pane in the Data Layout.

automatically changes the scale of the y axis for the value you selected, and italicizes the value name in the Data Layout. Double-click the item again to return the scale to its default setting.

For information on manually setting scales for left and right y axis values, see "Setting Chart Value Axis Properties" on page 230.

Using Data Functions in Charts

Data functions allow to recalculate values.

- To apply a data function:
- **1** Select a label in the Chart area.
- 2 Select Chart > Data Function > Function.

Data Function	Description
Sum	Returns sum of all values. This is the default setting.
Average	Returns average of all values.
Count	Returns number of values.
Maximum	Returns highest value.
Minimum	Returns lowest value.
% of Grand	Returns values as a percentage of all like values in the chart.
Non-Null Average	Returns average of values; null values excluded.
Null Count	Returns number of null values.
Non-Null Count	Returns number of values; null values excluded.

Table 39 Chart Data Functions

Note: Null values are empty values for which no data exists. Null values are not equal to zero.

Adding Computed Items

You can create new chart elements by building equations to compute data items, or by applying functions to existing data items. Computed items are like normal data items, and can be included in charts or re-used to compute other data.

For example, you can modify the *Amount Sold* item by building an equation around it, multiplying it by the *Unit Price* item and renaming the resulting item *Revenue*. You can also apply a scalar function such as Cume to *Amount Sold* and return each individual value as a cumulative running total, or simply multiply *Amount Sold* by the local tax rate to find the tax owed on each sale.

- > To create a computed item:
- 1 Select Add Computed Item from a Section menu (for example, Query, Results, and so on).

Computed Item			
Name: Computed	(OK)		
	Cancel		
Definition	Functions		
	Reference		
	Options		
	Help		
x			
+ · × / () == != and or if else mod < <= > >= not			

The Computed Item dialog box is displayed.

2 In the Name field, type a name that describes the computation.

The default name is Computed, which is numbered sequentially if there is more than one. If you assign a name to a computed item that is identical to an existing scalar function name, Interactive Reporting numbers the name starting with the number 2.

3 Define the new data item by building an expression in the Definition text box.

Use the operator buttons to insert arithmetic and logical operators at the insertion point.

- Click **Reference** to display the Reference dialog box, and select Request items to place in the equation.
- Click Functions to apply scalar functions using the Functions dialog box.

You can also type any portion of the equation or the entire equation directly into the Definition text box using JavaScript. The names are case sensitive, and you must replace spaces in item names with underscores ('_').

- 4 If necessary, click the **Options** button to set a new data type for the item.
- 5 When the equation is complete, click **OK**.

The computed item is listed in the Data Layout and is added to your chart.

Sorting Chart Items

Data in charts is sorted alphabetically by default. You can use the sort buttons on the Standard toolbar to perform simple sorts on selected items and reverse the sort order. In charts, however, you generally want to override the default alphabetical setting and sort dimensional data with reference to other data.

For example, if a chart lists each type of widget your company sells and the total amount sold of each, initially the widget types are ordered alphabetically. But this data becomes more meaningful when you instead sort the widget types with reference to the total produced by each. This approach enables you to rank each widget from the highest to lowest total sales.

You can use the Sort line in the Chart section to impose a sort condition for each dimensional data item in your chart. The Sort line includes three drop-down menus used to define the sort conditions. The contents of the menus vary depending on the data items in your chart.

- > To specify a sort using the Sort line:
- 1 If the Sort line is not already displayed, click **Sort** on the Section title bar.
- 2 Select an item to sort from the Sort drop-down list box.
- 3 Select a value from the By drop-down list box as a sort reference, or select Label to sort the item alphabetically.
- 4 If desired, select an aggregate function from the Using drop-down list box when sorting by values.

The Using drop-down menu is not available when you sort by labels.

5 If desired, click the ascending or descending Sort button on the Sort line.

The Sort line stores a sort condition for each dimensional item included in the chart.

Sort Items

The *sort* drop-down menu lists the data items that can be sorted. Each dimensional item included in the chart (name and date) is listed in this menu. Dimensional items can include Pivot *column and row labels*.

Reference Items

The By drop-down menu lists items used as a basis for a complex sort condition (for example, sorting Cities by the revenue generated in each).

- Label By default, dimensional data items is sorted alphabetically by name when you create your chart: this is equivalent to sorting by label. When selected, labels indicates that the item chosen from the Sort list is sorted by label or name, rather than by reference to corresponding numeric data values in the chart
- Value Sorting by a numeric data item orders each value of the target item chosen from the Sort list by its corresponding numeric value in the Value list.

Sorting by values produces an entirely different sort order. For example, your chart may list each state in which your company has made sales revenue and the total cost-of-sales for each. The states are initially listed in alphabetical order. When you sort by cost-of-goods, the states are ranked in order by each corresponding cost-of-sales figure.

Functions

The Using drop-down menu contains aggregate statistical functions that are available when you sort by values. The sort aggregate functions are usually the same as the data functions available in a section. When you sort by values, labels are sorted by the corresponding numeric values of the referenced item (for example, sorting states by the sum total of the cost of goods sold in each state).

Creating Pivot Tables from Charts

Once you have a final version of your chart, use the automatic pivot table-generator to create a *pivot table* based on the layout of your chart.

To create a pivot table based on your chart, in the Chart section, select Insert > Pivot This Chart.



A Pivot Table Created Using The Pivot This Chart Feature

The workspace switches to a new Pivot section that displays the pivot table created using the data from the chart.

Drilling into Charts

The *drill anywhere* feature enables you to drill into items in the Chart section that are resident in the Results section without having to return to reprocess your query or locate the item in the Catalog pane. Drill anywhere items are automatically added as new label items.

The advantage of this feature is that it instantly enables you to add items to the data set to reflect temporary or hypothetical situations. You can always suspend or delete the item to return to the original chart display.

The extent to which you can drill into your data depends on how the original query was built, since Drill Anywhere retrieves data from the Results section. This feature does not enables you to interactively query the database.

> To drill anywhere into a chart:

1 Select one or more items for analysis and select **Chart > Drill Anywhere > Item**.

Interactive Reporting redraws the chart drilled to the selected item. In the Data Layout, an item selected for drill-down is identified with a drill-bit icon.

- 2 Select Chart > Drillup to return to the original view of your chart.
- **Tip:** If no options are available in the Drill Anywhere menu, all Request items have been used in the Data Layout.
- Note: Drill Anywhere is enabled on the General page of Data Model Options. To display the General tab, select DataModel > Data Model Options.

Hiding and Focusing on Charted Data

A straightforward way to refresh your view of a chart is to single out items for closer focus or remove some of the charted elements. This enables you to concentrate on particular items of interest.

Focusing on Items

- > To focus on a chart item:
- 1 Select one or more objects on which you want to concentrate.

Focused item(s) are displayed with a dotted outline.

2 Select Chart > Focus On Items.

The chart is redrawn to display only the chart object(s) selected. A drillbit icon is displayed in the Data Layout next to the item(s) on which you focused.

▶ To return to the original chart display, select Chart > Show All Items.
Hiding Items

- To hide charted data:
- **1** In the chart, select the objects that you want to hide.

displays the item(s) with a dotted outline.

2 Select Chart > Hide Items.

Interactive Reporting redraws the chart with the selected objects removed. A drillbit icon *** is displayed in the Data Layout next to the item(s) you hid.

Restoring Hidden Items

To restore hidden chart items, select Chart > Show Hidden Items.

Working with Chart Elements

In the Chart section, you can easily reorganize or reposition data to reconfigure your charts and highlight different relationships between the same items. You can drag items to a different order within an Data Layout, drag label items between Data Layout designated for labels, or delete items from the Data Layout your chart is redrawn to reflect your changes.

You can also work directly with elements in the Chart area. Most elements are selected by clicking the element in the chart or the element's label in the legend. For axis labels, a change to one axis label changes all labels along that axis.

Review the following sections for information on:

- Selecting Chart Elements
- Displaying Axis Grid Lines
- Inserting Text
- Changing Chart Legends

Selecting Chart Elements

- To select chart elements as graphic objects in the Chart area:
- **1** Place the cursor over the edge of an item in the Chart area.

The cursor changes to a move cursor .

2 Select the object.

A gray outline is displayed around the item. You can move the selected chart item anywhere in the Chart area or resize it using the handles. Also, check the shortcut menu for additional options.

Displaying Axis Grid Lines

Axis grid lines are straight lines on a chart that provide a framework for measurement and reference. Typically, the x axis and Stack, Cluster and Depth (z axis) are used for label items and the Fact, Fact (Stack) and Fact (Depth) (y axis) shows values or facts (measurable items), such as units and amounts.

You can view or hide these axis gridlines depending on the chart you are designing.

- > To toggle the display of axis grid lines:
- 1 Click anywhere within the main plot area of the chart.
- 2 Select Show X Axis Grid Lines (or Show Y Axis Grid Lines) on the shortcut menu.

A check mark is displayed next to the selected option to indicate that the grid lines are visible. Select this option again to clear the check mark and remove the gridlines from the Chart area display.

Inserting Text

You can insert text anywhere around or within the chart to further explain or emphasize a chart component.

- ► To insert text:
- **1** Select **Insert Text on** the shortcut menu.

The Set Inserted Text dialog box is displayed.

2 Type the text you want to display in the text box and click **OK**.

The text is displayed in the location where you initially invoked the Insert Text command.

Changing Chart Legends

You can select the axis along which you want to distinguishes your data by setting the chart legend on that axis. This is a great way to view values on the selected axis without rearranging the values in the Data Layout. A chart legend can be set on either the X Axis, Y-, or Z-axis. You can also reposition or resize a legend to take advantage of either the horizontal or vertical space within the chart area.

Note: Chart legend color settings are preserved when the chart type, chart legend axis, and number of chart axis label values are changed.

The following three examples shows how to place the legend on different axes to alter the appearance and data shown by the same chart. In the first example, the legend has been set on the X Axis. In the second example, the legend has been set on the Facts axis. In the third example, the legend has been set on the Z axis:



> To set the axis used for a chart legend:

1 Select Format > Set Legend On.

The Set Legend On dialog box is displayed.

Set Legend On		
X-Categories Y-Facts		ОК
Z-Categories		Cancel
	~	Help

2 Select the axis on which you want to set the legend and click **OK**.

Colors are redistributed to highlight the data associated with the selected item and color coordinates corresponding labels in the Legend.

From the Legend pull-down menu, click the axis on which you want to position the legend.

- To resize a chart legend:
- **1** Click a border on the legend.

Sizing handles are displayed on each corner of the selected legend.

2 Drag a sizing handle until the legend is the desired size that you want.

Chart Scrolling and Scaling

You may want to scroll through data, enlarge your chart to better work with chart details, or change the perspective or angle from which a chart is viewed. Review the following sections for information on:

- Scrolling through Chart Data
- Rotating and Elevating Charts
- Smart Scaling
- Auto Resizing Charts
- Fitting Charts to Screen

Scrolling through Chart Data

When you add a Catalog dimension to the Data Layout of a Chart, the rendered Chart contains all bars and labels corresponding to the total number of unique items in the dimension. If the number of items added is large and as a result, all items cannot display in one view, you can scroll the entire contents of a chart either vertically or horizontally using the scrollbars. The horizontal scrollbar controls the scrolling of X Axis items, and the vertical scrollbar controls the scrolling of Stack, Cluster and Depth items.

Optionally, you or an administrator can define a specific number of items to show for each view of the chart. If the number of total bars is greater than the view size, a full page of bars displays. For example, assume each bar has three views and you want to display four bars (such as: A-B-C-D). When you first display the chart only bars A-B-C display, then when you scroll to the right bars B-C-D display.

If not all bars of a chart display in one view, the you can display the text "Partial View" as an indicator in the top left corner of the Contents pane. You can show or hide this indicator on a per chart basis by enabling or disabling the "Show partial view indicator" field on the General tab of Chart Properties.

The following rules apply to Chart scrolling:

- The chart legend displays only items visible in the current view.
- Scrollbars are not available for the Pie chart type.
- If all bars can display in one view, the scrollbars are disabled.

Rotating and Elevating Charts

By default, charts are displayed in 3-D foreshortened perspective from above and to the right. You can alter the perspective from which a chart is viewed by rotating it.

Before you can rotate the chart, the background plane must be visible.

- > To rotate the angle of vision or change the elevation of a bar, line, or area chart:
- **1** Select Chart > Properties.

The Chart Properties dialog box is displayed.

- 2 Click the General tab to view the General properties page.
- 3 In the Planes area, select the Show Back Plane check box and click OK.
- 4 Place your cursor at the top right corner of the chart.

The cursor changes to indicate that you can rotate the chart.

5 Hold down the left mouse button and drag the chart to rotate it.

When you release the mouse button, the chart is redrawn to reflect the adjusted perspective. If you cannot rotate in a certain direction, the chart has reached its farthest possible rotation in that direction.

Smart Scaling

Smart Scaling describes a method for ensuring that chart text label components are intelligently placed in relation to each other so that there is no overlapping, visual obstructions or clipping. Smart Scaling governs:

- Initial chart displays size
- Chart behavior during resizing
- Ability to move or reposition charts

You apply Smart Scaling to prevent overlapping and clipping of objects. Understanding the initial display, and using the Do Not Remove Text and Minimum Font as modifiers are key to the Smart Scaling algorithm.

Smart Scaling is enabled on the General tab of the Chart Properties. By default, Smart Scaling is enabled.

Initial Display

Smart Scaling relies on a display hierarchy or a draw order when it determines what is displayed. The display hierarchy is based on the number of components and the initial size of the chart boundary. The display hierarchy for the Chart sections is as follows.

Bar, Line and Area charts have this display order:

- Planes
- Bar/Lines/Ribbons
- Stack, Cluster and Depth values
- X axis values
- Fact, Fact (Stack) and Fact (Depth) axis values
- Legend
- Titles
- Axis
- Inserted text

Pie charts have this display order:

- Slices
- Slice Labels
- Legend
- Titles
- Inserted text

As you add or modify various Chart label components, Smart Scaling uses the hierarchy to determine which components to add or omit when placing items.

For example, in a Pie Chart where several small slices are in the same vicinity of each other, some of the pie slice labels might typically overlap due to simple space allocations. If you have Smart Scaling enabled, some of the slice labels are simply not drawn when crowded conditions are encountered. If a current label overlaps an existing label, then the label is skipped and the next label is drawn. You can correct this by moving the label slice. If there is enough room for other labels to display unobstructed when the label slices are repositioned, additional labels are displayed. You can also enable the "Line to Label" feature to enhance the slice-label relationship, and identify and place items easier.

Retaining Text

When a chart object is reduced disproportionately to other objects in the bounding area and Smart Scaling is enabled, text may be eliminated based on the display hierarchy. You can prevent text from being removed by enabling the Do not remove text option available on the General tab of Chart properties. In the case where a chart object is reduced too much, text is preserved, but may overlap other object or appear distorted

Minimum Font Size

When the overall size of a chart is modified, Designer attempts to redraw chart components using the object's font size. If it is not feasible to redraw the component in the space available, the font size is reduced by one point, and another drawing is attempted. This process is repeated until the component can fit within the boundaries defined for it, or until a minimum font size is reached.

You specify the minimum font size on the "General" tab of Chart Properties. The default value for the minimum font size is 8.

The components which are modifiable by the minimum font size are:

- y axis, x axis and z axis
- pie slice values
- legend

When the component is first reduced, it continues to show the original font size and not the user defined font size. If the minimum font size is reached for the supported components and it is still too large to fit with obstructing or being obstructed, then the item is removed from display.

When Smart Scaling is used in conjunction with Auto Resize, changes to the Chart size are limited to all those methods which do not involve the movement of the chart border.

If you select the handles of the chart border and reposition them as the method to resize the Chart when Smart Scaling and Auto Resize are enabled, Auto Resize is automatically disabled. The Smart Scaling property remains enabled.

Moving Chart Components

Almost all chart components can be moved independently of each other within the borders of the chart once is has been drawn. With the exception of bar values and axes values, all other chart components can be moved.

With Smart Scaling, the chart title, subtitle, axes labels and legend cannot be moved over each other and cannot be moved over the axes values text and the chart graphic.

If Smart Scaling is not enabled, a Chart component can be moved anywhere within the border of the chart. If the component is layered on top of an existing component, it overrides that component in the display.



Smart Scaling Off

- To enable Smart Scaling:
- **1** Select Chart > Properties.
- 2 In the Chart Properties dialog box, click the General tab and select Smart Scaling.
- 3 To use a minimum font size, add a font size in the **Min Font Size** box and click **OK**.

Auto Resizing Charts

You can use the following options to resize a chart:

- Drag the handles of the chart border, which redraws the chart to fit the new border dimensions,
- Enable the "Auto Resize" option on the General tab of Chart Properties. This feature automatically expands the chart border to fit the maximum height or width of the Contents pane.
- Enable the "Auto Resize" property and adjust the size of the entire Designer application by dragging the corner the application window. This redraws the chart components automatically.

- Enable the "Auto Resize" property and adjust the size of the Contents pane by adjusting the Section/Catalog pane border. This redraws the chart components automatically.
- Enable the "Auto Resize" property and show or hide the Outline and or Sort line.

Resizing a chart can result in the overlap of chart components, especially when a chart is reduced in size. To prevent this overlap, you can use the Smart Scaling and Minimum Font Size features (both described above). By default, Auto Resizing is enabled.

- > To enable Auto Resize:
- **1** Select Chart > Properties.

The Chart Properties dialog box is displayed.

- 2 Click the **General** tab to view the General properties.
- **3** Select the Auto Resize check box and click OK.

Fitting Charts to Screen

When the dimensions of a chart are smaller or larger than the actual size of the Contents pane, and you wish to maintain relative proportions between the chart and the actual size of the pane, use the "Fit to Screen" icon on the Chart toolbar.

> To fit a chart to the Contents pane, click the Fit to Screen icon.

Customizing Chart Properties

Use the Properties command on the Chart menu to customize general chart, labels axis, and values axis properties, as well as general bar chart attributes. Review the following sections for information on:

- Setting Line Chart Data Labels Properties
- Setting Chart Label Axis Properties
- Setting Chart Value Axis Properties
- Setting Bar Chart Properties
- Setting Chart Label Axis Properties
- Setting Chart Value Axis Properties
- Setting Bar Chart Properties
- Setting Line Chart Patterns Properties
- Setting Line Chart Data Labels Properties

Setting General Chart Properties

General chart properties control the display of various objects, such as titles, legends, and borders, in the Chart area. They are also used to specify scaling, scrolling, sizing, default plane and rotation settings.

- To adjust general chart properties:
- **1** Select Chart > Properties.

The Properties dialog box is displayed.

2 Click the General tab to display the General properties page.

Properties		
General Labels Axis Val	ues Axis Bar Chart	
Show title Show subtitle Show subtitle Show legend Show border Show border	Sales by Region by Store Type	OK Cancel Help
 ✓ Show parka view indicator ✓ 3-D objects ✓ Print/PDF all views ✓ Auto resize ✓ Smart scaling Minimum font size 	Do not remove text	
Use legacy colors for data p Planes Show horizontal plane Show vertical plane Show back plane	Points Rotation Horizontal degrees 15 Vertical degrees 20	

- **3** Change any of the following properties:
 - Show title Toggles the display of the chart title as entered in the text field.
 - Show subtitle Toggles the display of a subtitle as entered in the text field.
 - Show legend Toggles the display of color-coded Chart legends in the Chart area.
 - Show border Toggles the display of a border around the chart. The border controls the general position of all other Chart components not controlled by the Chart graphic border. This border can be resized and repositioned.
 - Show partial view indicator Toggles the display of the "Partial View" indicator in the top left corner of the Contents pane. The indicator shows if not all bars are displayed in the current view. The indicator is displayed on all output media, including Designer, Workspace, PDF and print).
 - **3-D objects** Toggles the 3-D display of chart objects, including pie slices, chart bars, side planes, and perspective view.
 - **Print/PDF all views** Toggles the print of the entire rendered chart. The setting you specify here is not applied to the Export to PDF option in the Workspace.

- Auto Resize Toggles the automatic resizing of a chart within the Content pane whenever the Chart is resized.
- Smart Scaling Toggles the Smart Scaling feature, which governs how Chart components are initially displayed, how they behave during Chart resizing and how the Chart components can be moved or repositioned.
- Do not remove text –Toggles the do not remove text. When this feature is enabled, Smart Scaling (see above) does not remove any text (it does not clip or overlap text if an object is resized or repositioned to a minimum font size.). This feature is checked for new Chart sections, and unchecked for 8.3 releases and earlier that have Smart Scaling enabled. Inserted text and legends are not affected by your selection in this field. Smart Scaling may shrink these objects, but they are never scaled to the point where they are eliminated entirely.
- Min Font Size Toggles the minimum font size feature for text labels when the overall Chart size is changed and Smart Scaling (see above) is enabled. If it not possible to fit a component in the available space, the font size is reduced by one point and a redraw is attempted. The font size reduction process repeats until the component can fit within the component's boundary or until the font size specified here is reached. If the component is still too large to fit without obstructing another object, or it is obstructed, then the component is removed from the display. The default minimum font size is 8.
- Use legacy colors for data points Enables an individual Chart to use legacy colors from the Release 8.3 and earlier Color Palette. When this option is unchecked, the newly added Chart uses the default colors from the 8.4 and later Color Palette. To enable all Charts to use the legacy colors from the Release 8.3 and earlier Color Palette, check the "Use legacy colors for data points" option on the Default Fonts and Styles box. This option is saved with other options at the application level. For more information about this option, see "Customizing Chart Colors" on page 234.
- Planes Toggles the display of horizontal, vertical, and back planes of a chart.
- Horizontal\ Vertical rotation degrees Shift the perspective angle or elevation (–60 to 60 degrees) for all types except the pie chart, which has its own rotation (see below)
- Rotation For pie charts, rotates the pie the specified number of degrees (between -90 and 90), and changes the pie height the specified number of degrees (0 to 90). For all other charts except line charts, shifts the horizontal and/or vertical angel or elevation a specified number of degrees (between -60 and 60).
- 4 Click **OK** to apply your changes.

Setting Chart Label Axis Properties

Chart label axis properties control the display of the X Axis and Depth (Z-axis) labels, tickmarks, and values for items in the Data Layout.

- > To adjust chart label axis properties:
- **1** Select Chart > Properties.

The Properties dialog box is displayed.

2 Click the Labels Axis tab to display the Labels Axis properties.

XAxis	ОК
Show axis label Line Name	Cancel
Labels and Tickmarks	
Show tickmark At every 1 bar(s)	
F Show label At every 1 bar(s)	
Expand label box	
Scrolling Maximum bars displayed 8	

- 3 Modify the properties shown as desired and click **OK** to apply your changes.
 - Horizontal Vertical rotation degrees Select to display x axis and z axis labels, tickmarks, and values, and whether the intervals at which x axis labels and tickmarks display is automatically determined by or manually set to the specified value.
 - Expand label box Increases the label box size for anx axis label (y and z labels are unaffected) in an individual Chart. It is only active when the Show Label check box is enabled and the Auto Frequency check box is disabled. It is not available for pie charts. This feature is used primarily for sequential labels, for example, time, day or number labels. For regular name-based labels, this feature should be disabled so that the user can view each label name.
 - Scrolling Set the number of bars to display per view if multiple views that are enabled for a rendered Chart.

Setting Chart Value Axis Properties

Chart value axis properties control the display of axis labels, tickmarks, values, and position for items in Y-Facts in the Data Layout.

- To adjust chart value axis properties:
- **1** Select Chart > Properties, or right-click the chart and select Properties.

The Properties dialog box is displayed.

2 Click the Values Axis tab to display the Values Axis properties page.

Properties	
General Labels Axis Values Axis Bar Chart	
Show left axis label Total Unit Sales	ОК
Show right axis label	Cancel
Show tickmarks at intervals Show values at intervals Show values at initervals	Help
Left Axis Scale Logarithmic Scale Axis Scale Axis Scale Axis Scale Axis Scale Axis Scale Axis Scale Axio Auto Min Max 4000000 Max Auto Auto Auto Max Auto Aut	

3 Modify the properties shown as desired and click **OK** to apply your changes.

You can display labels, tickmarks at intervals, and values at intervals for primary and secondary value axes.

Allow adjustable scaling for faster performance – Select this field to calculate the min/max (Y) range based on the first two thousand bars. As you scroll through this range (moving to the right [X] or back [Z]}, of if any bar is above the maximum or below the minimum, the Y scale is expanded to all bars.

If this feature is unchecked, the min/max range is calculated based on all bars.

You may also set the left axis and right axis scales using a logarithmic scale, which plots line charts with logarithmic (base 10) values; and whether the values displayed on the scale are automatically set by Interactive Reporting, or set by manually specifying minimum and maximum values. For the left axis scale, you may specify the interval at which to separate axis values or have Interactive Reporting automatically assign the interval.

If all bars are displayed in a view (the chart scrolling is off), the true minimum and maximum are calculated.

If you print a chart when "all views" are active, the minimum/maximum range is calculated based on all bars.

Note: If you plot a line chart with a logarithmic (base 10) value axis, any chart values less than one (1) are represented as zero (0) on the logarithmic scale and an information message is displayed on the Windows status bar.

Setting Bar Chart Properties

Bar chart properties control the general attributes for the various bar chart formats available in Interactive Reporting.

To adjust bar chart properties:

1 Select Chart > Properties.

The Properties dialog box is displayed.

2 Click the Bar Chart tab to display the Bar Chart properties page.

General 📔 Labels Axis 🗍 Value	es Axis Bar Chart	
Bar Charts Show values on bars Clustered Bar Chart Cluster bars by items in Y' Cluster bars by items in Z'	axis axis	OK Cancel Help
Bar-Line Charts Show values on bars Show line values Ignore null values for lines Shift	Stack/Duster	
 Shift Points to Left Shift Points to Center 	 Stack bars along Y' axis Cluster bars along Y' axis 	

- 3 Modify the properties shown as desired and click **OK** to apply your changes.
 - All Bar Charts Toggles the display of values on each bar in the chart.
 - Clustered Bar Charts Clusters bars by items on the Fact axis (Y-axis) or X Axis. For example, if you have *Quarter* on the Y-axis, each cluster consists of four bars, one for each quarter.
 - **Bar-Line Charts** Toggles the display of values on bars and for lines, and the inclusion of null values (which disrupt the line, area, or ribbon). Zero' (0) values are included.

You can also place line chart plot points to the left of or centered on bars, and stack or cluster bars on the Y-axis.

Setting Line Chart Patterns Properties

Use the Line Chart Patterns tab to apply a pattern to the appearance of line elements and markers in a Line Chart.

Setting Line Chart Data Labels Properties

Use the Data Labels tab to toggle data labels for line chart elements.

Customizing Chart Patterns and Labels

In addition to the generic label axis properties that apply to all charts, you can change properties of individual charts items, such as patterns, colors, and data labels. Review the following sections for information:

- Changing Chart Color Schemes and Fill Patterns
- Changing Chart Data Labels
- Changing the Color of Chart Elements, Lines, and Text

Changing Chart Color Schemes and Fill Patterns

You can redistribute chart colors to emphasize specific charted items listed in the Data Layout and change chart focus. Colors are differentiated along a specific axis.

- > To change the pattern or color scheme of a chart:
- **1** Double-click a chart element in the legend, or select an element in the chart (such as a bar or a pie slice) and select **Chart > Properties**.

A Properties dialog box is displayed that may contain one or more tabs (Patterns, Axis, Data Labels), depending on the format of the active chart.

- 2 Click the **Patterns** tab to display the Patterns tab.
- 3 Select the fill pattern and color for the chart segment, line, or marker and click **OK**.
 - Automatic Automatically sets the fill pattern and color to the default setting. The default setting for fill pattern is *Solid*. The default setting for color varies between chart formats.
 - None Sets a transparent fill pattern and color for the selected element.
 - Custom Enables you to select a foreground color and fill pattern for the chart. Choices include Solid, Hollow, Horizontal, Vertical, Cross, Diagonally Up, Diagonally Down, Diagonally Cross.

For line charts, you can also select the width, style and color of lines, and the size, style, border colors, and fill colors of markers.

Changing Chart Data Labels

You can change the type of data used in the labels in your chart. The choices available depend on the chart format.

- To change a chart's data labels:
- 1 Double-click a chart element in the legend, or select an element in the chart (such as a bar or a pie slice) and select Chart > Properties.
- 2 Click the **Data Labels** tab to display the **Data Labels** tab.
- **3** Select the fill pattern and color for the chart segment, line, or marker.
 - Pie Charts Select whether to show pie slice labels, values, percentages, negative values, and lines that connect a data label to its pie slice.
 - **Bar Charts** Select whether to show individual bar values and the placement of the values inside the bar.
 - Line Charts Select whether to show values and ignore null values, and select the placement of the values relative to the line.

For line charts, you can also use the Axis page to specify whether to plot values on the primary (left) axis or secondary (right) axis.

Customizing Chart Colors

This section explains color options available in the Chart Section.

Default Color Palette and Legacy Color Palette

By default, Interactive Reporting bases the color scheme used in a Charts on the default colors in the Color Palette. This palette consists of a legacy color palette (pre 8.4 color palette) and a default color palette as shown below:



The legacy color palette consists of the first six rows in the color palette.

The default color palette consists of the last two rows. There are seventeen chart default colors and the chart background color is the last color on the bottom row.

The numbers below are the hexadecimal numbers for the rgb codes in the default Color Palette. The numbers map to the colors from left to right in the default Color Palette.

- ZColor(30583, 30840, 58596),
- ZColor(63479, 48830, 9766),
- ZColor(1542, 52428, 52685),
- ZColor(46517, 23387, 42405),
- ZColor(19018, 53456, 24929),
- ZColor(65535, 60395, 0),
- ZColor(52428, 13107, 13107),
- ZColor(39321, 39321, 39321),
- ZColor(63222, 13878, 41891),
- Scalar(7710, 28013, 11308),
- ZColor(61680, 65535, 26728),
- ZColor(39321, 0, 39321),
- ZColor(39321, 65535, 39321),
- ZColor(47031, 31097, 9252),
- ZColor(62965, 43690, 54741),

- ZColor(39321, 257, 13364),
- Scalar(52428, 52428, 39321),

Using Legacy Chart Colors

When you open a Chart created in Release 8.3 and earlier, the Chart uses the default colors available in the legacy color palette. This feature is enabled automatically for all 8.3 and earlier charts in the "Use legacy colors for data points" check box on the Default Fonts and Styles dialog box. In addition, when this is checked, all newly added charts use the default colors in the Legacy Color Palette that were used in 8.3 and earlier. When this is unchecked, which is a default value, all newly added charts use the default colors in the Default Color Palette. This value is saved with other application level parameters in this dialog.

For an individual Chart, you can use the default colors available in the Legacy Color Palette by enabling the "Use legacy colors for data points" check box on the General tab of the Chart Properties dialog.

If you modify the Chart (for example, you perform a drill down, or change the chart type), the colors of the Chart use the default color sets depending on the "Use legacy colors for data points" option. For example, if the check box is not checked, the new default colors are used, otherwise, the old default colors (pre 8.4) are used.

In the example below, the bar chart uses the default colors from the Legacy Color Palette:



Using Default Colors

By default, Charts created globally in Release 8.4 and later use the default colors available in the Default Color Palette. In this case, the "Use legacy colors for data points" check box is unchecked on the Default Fonts and Styles dialog box.

For an individual Chart, you can use the default colors available in the Default Color Palette by uncloaking the "Use legacy colors for data points" check box on the General tab of the Chart Properties dialog

If you want to use the default colors from the Legacy Color Palette, enable the "Use legacy colors for data points" check box on the General tab of the Chart Properties dialog

In the next example, the bar chart uses the colors from the Default Color Palette:



Changing the Color of Chart Elements, Lines, and Text

Use the Format toolbar to change the line, fill, and text color of a chart element or text and add emphasis to your chart.

- Note: If the Format toolbar is not visible, select View > Toolbars > Formatting.
 - ➤ To change line color:
 - **1** Select a plot point in the line whose color you want to change.
 - 2 On the Format toolbar, open the Line Color list and select a color from the palette.
 - > To change the fill color of a chart element:
 - **1** Select the chart element whose fill color you want to change.
 - 2 On the Format toolbar, open the Fill Color list and select a color from the palette.
 - To change the color of text:
 - **1** Select the label or other text element whose text color you want to change.
 - 2 On the Format toolbar, open the **Text Color** list and select a color from the palette.

Chart Menu Command Reference

Table 40 provides a quick reference to the commands available on the Chart menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	Shortcut Menu
Sort	Sorts the selected column values in ascending order (alphabetical or numeric).		~
Data Function	Applies a prebuilt data function to the selected item.		
Add Computed Item	Enables you to add a new data item derived from calculations performed on an existing item.		~
Modify Computed Item	Enables you to modify a computed item.		~
Remove Selected Items	Removes the selected items.	[Del]	~
Drill Anywhere	Enables you to drill to any item.		~
Drillup	Returns the original view of the data you drilled.		~
Focus On Items	Updates the chart to include only the selected items.		~
Hide Item	Hides the selected item from view.		~
Show Hidden Items	Restores the selected hidden item.		~
Show All Items	Updates the chart to include all items removed by focusing.		~
Group Items	Groups the selected items.	[Ctrl+G]	~
Ungroup Items	Ungroup the selected grouped item.	[Ctrl+U]	~
Restore Name	Restores the original name of a renamed item.		~
Properties	Opens the Properties dialog box for the chart or selected chart element.		~

Table 40	Chart Menu	Commands
	onuremonu	oommunuo

Report Designer

10

In This Chapter

Chapter	Report Section	240
	Creating a Custom Report	242
	Working with a Report Page	251
	Setting Up a Report	252
	Enhancing Report Data	254
	Using Multiple Data Sources in a Report	259
	Creating Smart Reports	261
	Formatting Report Items	261
	Converting Detail Reports from Versions Earlier than 6.0	262

Report Section

In the Report section, the Report Designer helps you to easily develop a complete range of reports, from small ad-hoc reports to mission-critical operational reports. After you create your database query, you can use this visual layout capabilities of the section to drag and drop columns, expressions, charts, pivot tables, logos, and graphic objects to quickly design and customize your reports.

Report Section Elements

The Report section differs slightly from other Sections in order to provide you with as robust a report designer as possible. Key differences include:

- Expression Line Accessed by way of the Expression button on the Section title bar, the Expression line enables you to build common computed expressions using JavaScript.
- Catalog pane Contains all of the of drag-and-drop elements that you use to create a custom report, including:
 - Query Contains all the sections associated with a selected query, including Results, Pivot, and Chart sections. You can drag individual Results columns from the Results and Table sections to a report page, as well as entire Pivot and Chart sections to create *Smart Reports*.
 - Graphics Contains standard vector graphics text labels, and pictures (bitmaps only).
 - Fields Contains predefined fields that can be dragged to various areas of the report to enhance the look and feel of the report.

To use any of these elements, simply drag them to the desired report component band.

- Data Layout Consists of the Groups Data Layout and Table Data Layout. The Table Data Layout is divided into the Dimension and Fact panes.
 - **Groups Data Layout** Defines the overall or highest levels used to group data in a report. When you designate an item to serve as a group header (also known as a break value), you are instructing Interactive Reporting Studio to organize the rest of the data in repeating collections of records according to the group header.
 - Dimensions pane in Table Data Layout Includes descriptive information such as a column in a table that is included in the body of the report.
 - Facts pane in Table Data Layout Includes the measurable or quantifiable data as a column in a table included in the body of the report. Interactive Reports calculates and inserts subtotals for each fact column.

Interactive Reporting quantifies values by group header and dimension. If you have a descriptive numeric value that should not be calculated, such as Retail Price or Target Sales, use it as a group header or table dimensions instead of a fact.

Report Section Toolbar

The Report Section toolbar provides icons that enable you to quickly maneuver multiple report objects.

- Align Aligns several objects at the same time. Objects are aligned to the first object you select. Select the first object, then hold down the Control key and select the remaining objects. Click the arrow on the Align icon and select an alignment option: left, center, right, top, middle, or bottom.
- Make Same Size Resizes the selected objects to the same size. Objects are resized to match the first object you select. Select the first object, then hold down the Control key and select the remaining objects. Click the arrow on the Make Same Size icon and select a resizing option: width, height, or both.
- Layer Stacks a single object in relative position to other objects. Layer includes four rearrangement options: Send To Front, Send To Back, Bring Forward, and Send Backward. Use this feature to layer multiple objects so that only the sections of the objects you want visible are shown.
- **Spring Objects** Maintains relative vertical spacing between dynamic objects. That is, you can *spring* one object to another so that if the first object is moved, increased or diminished, the second object moves in the same flow.

Select an object, then hold down the Control key and select the remaining objects. Click the Spring Objects icon to spring the objects.

To remove spring from objects, select the objects click the Spring Objects icon again.

• Zoom – Sets the magnification level of the report. Options include whole page, page width, or a percentage of magnification based on 100%.

Expression Line

Use the Expression Line to apply and display aggregate functions and computing fields. For each item in a report, you can display its JavaScript syntax and modify it to fit your own needs.



The Expression line includes the following components:

Data Function - Shows available data functions that can be applied to table columns.

- ¥ Cancels and removes a formula. This icon is displayed only when you enter syntax.
- $\sqrt{-}$ Accepts and applies the formula. This icon is displayed only when you enter syntax.
 - Edit bar Used to display, enter, and edit a Javascript expression.

Expression Syntax

Table 5-1 lists the default syntax for Report section items.

Table 41 Report Expression Syntax

Report Element	Syntax
Table Dimension	CurrBreak.Value["Column Name"] "Column Name" refers to the dimensional column name. The column name can be modified.
Table Fact	Tables ["Section"].Columns ["Column Name"].Sum[currBreak] "Section Name" refers to either the Results or Table section. Column Name refers to the fact column name. Sum refers to the return of underlying values and can also be an applied Data functions such as Avg, Max, and Min.
Group Label	CurrBreak.Value["Column Name"] "Column Name" refers to the dimension or fact column name.
Group Fact	<pre>Tables["Section"].Columns ["Column Name"].Sum[currBreak] "Section Name" refers to either the Results or Table section. "Column Name" refers to the fact column name. To apply a break at the parent level (the highest level), use the following syntax: Tables["Section Name"] .Columns ["Column"].Sum[currBreak. Parent]</pre>

You can concatenate the syntax listed in Table 5-1 with other JavaScript expressions to customize the content of labels, facts, and dimensions.

For example, if you wanted a group header to show a union of the Total label and the amount, you could insert a field in the Expression line to concatenate Total with the table fact syntax.

"Total: " + Format (Tables("Results").Columns("Amount").Sum(cunBreak). "#.##0") 1996 Total: 1.400.047

Or, you could insert a field to show the sum (or any other data function) of the column divided by the sum of the parent level expressed as a percentage.

Format [Tables['Results'];Columns['Amount'];Sum(currBreak) / Tables['Results'];Columns('Amount'];Sum(currBreak,parent); "D4'']
Quarter 1 23%

If you plan to use this feature, add the labels in a computed field. For information on how to add a computed field, see "Adding Computed Items" on page 5-34.

Creating a Custom Report

The Report section uses tables as the basic building blocks of custom reports. Tables contain columns of dimensions and facts as determined by the Results items you place in the Dimensions and Facts panes in Table Data Layout.

- To create a basic report:
- **1** Select Insert > New Report.

Interactive Reporting creates a new Report section and inserts a blank table in the Body band of the report in the Content pane.

2 If desired, select Report > Section Boundaries to view the bands for the report components.

If the Table Data Layout is not visible, click **Table** on the Section title bar to open the Table Data Layout.

- 3 Drag Results items from the Query sections of the Catalog pane to the Table Data Layout.
- **Tip:** The Table Data Layout has two panes Dimensions and Facts. Use the Dimensions pane to build dimension (label) columns, such as Month, Region, or Product Line. Use the Facts pane to build the facts (numeric values) in the report, such as Amount Sales or Units Sold. Results and Table items added to the Facts pane are totaled automatically.

Report Components

The Report Designer offers a variety of dynamic tools for constructing the report you want to create with all of the components you want it to contain. Understanding how these components perform and how they integrate with Report section elements is fundamental to building a successful report.

Report pages are structured areas, or bands, of information. Each band contains a different report component and can be customized to include other report elements such as graphic objects, predefined fields, computed fields, charts, and pivot tables.



A components of the report are:

- Body Contains one or more tables whose contents are based on the items listed in the Table Dimensions and Table Facts panes of the Data Layout. Every report has a body that holds a table of data. Tables can be based on the same or different results sets within the document.
- **Report Group Headers** Categorize data into repeating collections of records organized according to the header band.
- **Report Header/Footer** Fully customizable summary bands of information. Report headers print only on the very first page of the report. Report footers print only on the very last page.
- **Page Header/Footer** Contain data that is repeated on every page, such as page numbers. These bands are also fully customizable.

Table Column Formatting Options

Table 42 lists the column formatting options available in Interactive Reporting.

Options	Description
Changing the Column Name Heading (Title)	To modify the column title, click Expression on the Section title bar, then select the column title that you want to modify. In the Expression line, type the new title between the quotation marks ("") and click the check mark.
Hide/Show Column Name Heading (Title)	To toggle the display of column titles, select Column Titles on the shortcut menu. A check mark is displayed next to Column Titles to indicate that column titles are visible. Select this option again to clear the check mark and hide column titles from view.
Remove Column	To remove a selected column from the report (and Table Data Layout), select Remove Column on the shortcut menu. This option is available for dimension and fact columns. Items should be removed with caution as computed items may draw data values from the deleted item.
Hide/Show Column Total	To toggle the display of column totals, select Show Column Totals on the shortcut menu. A check mark is displayed next to Column Totals to indicate that column totals are visible. Select this option again to clear the check mark and hide column totals from view. This option is available only for fact columns.
Select Column	Click anywhere inside the column.
Suppress Duplicates	To suppress the duplicate values for a column, select the column and select Format > Suppress Duplicates . Use this feature if you want to display only the first instance in the column of a duplicate value when individual database records include redundant information. This feature is especially useful if records are associated with the same date, location or customer.
Text Wrap	To wrap text within a column, select the column and select Format > Text Wrap.

Table 42	Table Column	Formatting	Options
----------	--------------	------------	---------

 Table 42
 Table Column Formatting Options (Continued)

Options	Description
Sort Ascending/Descending	To sort column values, click Sort Ascending or Sort Descending on the Table menu, shortcut menu, or standard toolbar.
Reordering Columns	If you want the columns in the table to be displayed in a different order, drag the item in order of how you want it to be displayed in the Table Data Layout. You can also select the column in the actual report and drag it to a new position in the report.

Inserting Additional Tables in a Custom Report

A custom report may contain multiple tables, each originating from the same or different result sets in the document. Multiple tables in one band are a powerful way to compare values.

- > To add multiple tables to one report:
- **1** Select **Report > Insert Table**.

The cursor changes to a crosshair +.

2 In the Body band, left-click and hold your left mouse button to position your cursor where you want to insert the table, and then drag to create the table.

A numbered table is inserted.

3 Drag the Results items from the Query sections of the Catalog pane to the Table2 Data Layout.

Each new table that you insert has its own corresponding dimensions and facts pane in the Table Data Layout.

Adding Report Groups

A report group is the topmost level at which you can structure data in a report. When you drag an item from the Catalog pane into the Groups Data Layout, the Interactive Reporting supplies a report group header band automatically and adds a label inside the band that identifies the group.

For example, if you create a report to show purchases by state, each state would serve as a report group header in the report. You can place multiple items in one header or add a multiple levels of group headers.

This step is not necessary to create a report, but it may help you to better organize the content of the report. You can show additional headings, graphics, and totals in the report group header band.

- > To create a report group header:
- **1** If the Groups Data Layout is not visible, click **Groups** on the Section title bar to open the Groups Data Layout.

2 Drag the item on which you want to group data from the Catalog pane to the Report Group1 field in the Groups Data Layout.

The selected item automatically populates the group header.

Inserting Report Headers and Footers

Report headers and footers are summarizing bands of information. Report headers print only on the first page of a report. Report footers print only on the last page of a report.

- **Tip:** Do not confuse report headers and footers with report *group* headers. Report group headers categorize data into repeating collections of records in a header band, based on the Results columns in the Groups Data Layout.
 - > To view and create a report header or footer:
 - **1** Select **Report > Section Boundaries** to view the bands for the report components.
 - 2 Select Report > Headers And Footers > Option.

You can view the report header, the report footer, or both. The Report Header band is displayed on the first page of the report. The Report Footer band is displayed on the last page.

3 Type the information to be displayed in the report header or footer, or drag objects from the Catalog pane to the Report Header or Report Footer band.

The selected item populates the report header or footer automatically.

Inserting Page Headers and Footers

Page headers and footers enable you to specify data that is repeated on every page, such as a page number.

- > To insert a page header or page footer:
- **1** Select **Report > Section Boundaries** to view the bands for the report components.
- 2 Select Report > Headers And Footers > Option.

You can view a page header, a page footer, or both.

3 Type the information to be displayed in the report header or footer, or drag objects from the Catalog pane to the Page Header or Page Footer band.

The selected item automatically populates the report header or footer.

Tip: You can hide the page header on the first page of a report and show it for the rest of the pages in your report by typing the following code in the "Expression Line" for the page header object:

```
if (PageNm == 1)
{' '}
else
{"Query Processed: "+Format(newDate(), "d-mmm-yyyy")};
```

Inserting Page Breaks

You can insert a page break before or after a report group header.

- ► To toggle page breaks:
- 1 Select Report > Section Boundaries to view the bands for the report components.
- 2 Select a report group header.
- 3 Select Insert > Page Break Before or Page Break After.

A check mark is displayed next to the selected page break option to indicate that it is active. Select this option again to clear the check mark and remove the page break.

Adding Other Report Elements

You can resize report band items and use the tools available on the Format toolbar to change properties such as number, font, alignment, border, and background.

You can also add graphic items and fields from the Catalog pane to enhance your report. For example, use the Text Label tool to add text labels that identify values for numeric report group headers. Review the following sections for information on:

- Working with Graphic Elements
- Working with Fields
- Working with Computed Fields
- Inserting Filter Values

Working with Graphic Elements

Interactive Reporting provides a complete set of graphic elements to assist you in designing presentation-quality reports, (see Table 43). Lines, rectangles and ovals are considered vector graphics. Vector graphics consist of commands for creating the measures and shape of each line, rectangle, arc, and so on. The resolution of the output device defines the look of the vector graphic, since a vector graphic has no fixed resolution. This property enables you to resize the vector graphic without changing its resolution. Since a vector graphic contains only the instructions for creating an image, it requires less disk space.

Pictures are considered bitmapped graphics. A bitmap graphic consists of pixels, which have the appearance of small points. In computer memory, a pixel is shown as one or more bits, containing instructions for color, density and appearance for each pixel shown. As a result, bitmaps use more disk space and have a fixed resolution. If you resize a bitmap image, it may distort the original image.

Table 43	Report Section	Graphic	Elements

Element	Description
Line	Creates a line that you can rotate.
Hz Line	Creates a horizontal line.
Vt Line	Creates a vertical line.
Rectangle	Creates a rectangle.
Round Rectangle	Creates a rectangle with rounded corners.
Oval	Creates an oval.
Text Label	Creates a text label that you can use as a caption.
Picture	Enables you to insert bitmaps (. bmp extension).
Name	Sets the name of the object.
Visible	Enables the object to be visible during run mode. Use this to hide objects.
Locked	Locks an object. Use this to lock objects so they cannot be changed.

> To insert a graphic object, use one of the following options:

- Expand the folder in the Catalog pane that contains the desired graphic object; then, drag and drop the object into the Contents pane.
- Select **Report** > **Insert Graphic**, select a graphic object from the menu that is displayed, and click the Contents pane to insert the graphic.

Working with Fields

Interactive Reporting provides a computable field and predefined fields to assist you constructing and producing reports see Table 44). You can drag and drop these fields anywhere within the report page, body, report group header, report header/footer, and page header and footer. You can also customize how numbers, dates, time, and text fonts are displayed in your report. For numbers, you can use commas, decimals place, dollar signs or percentage symbols. For text items, you can select fonts styles and sizes. For Query and Result filters, you can select specific filters.

Field Name	Description
Field	Inserts a computed field.
Query Filter	Inserts a selected query filter.
Result Filter	Inserts a selected result filter.

Table	44	Report Section Field	s
IUDIC			9

Field Name	Description
Query SQL	Inserts the last SQL sent to the database when you process a query.
Page Number	Inserts a page number.
Number of Pages	Inserts the total number of pages.
Page X of Y	Note the following behavior of this field when the section is deployed in the Hyperion System 9 BI + Workspace:
	If you intend to use this field in a report deployed in the Hyperion System 9 BI + Workspace, any Report section which is more than one page shows as 'Page 1 of y'. It continues to show the pages as such (with an unknown 'y' value) until the end user navigates to the last page of the section.
	If an end user navigates to the last page using the "SHIFT+toolbar button" combination, all pages are generated and the page # toolbar icon tool tip shows the correct page position and total page value. Subsequent navigation continue to show correctly the current and total page numbers. These values persist for the entire user session.
	If the Report sections has fields in the section which show 'Page x of y', the fields match the page # toolbar icon tool tip.
	All pages are generated if the Report section is exported to XLS or PDF formats.
	If the Interactive Reporting document is saved without Results which the Report section references, processing the Interactive Reporting document file causes a generation of the first page only.
Last Saved	Inserts date on which the report was last saved in MM/DD/YYYY format.
Last Printed	Inserts date on which the report was last printed in MM/DD/YYYY format.
Date	Inserts and stamps the current date in MM/DD/YYYY format.
Time	Inserts and stamps current time in HH:MM:SS format.
Date & Time	Inserts and stamps date and time in MM/DDYYYY and HH:MM:SS format.
Date Now	Inserts the current date in MM/DD/YYYY format.
Time Now	Inserts the current time in HH:MM:SS format.
Date & Time Now	Inserts the current date and time in MM/DDYYYY and HH:MM:SS format.
File Name	Inserts an Interactive Reporting document (.bqy) name.
Path Name	Inserts the full path name of the document.
Report Name	Inserts the report name.

 Table 44
 Report Section Fields (Continued)

Working with Computed Fields

A computed field contain a single value based on a calculation. You can create customized JavaScript expressions or modify JavaScript syntax for an existing report element. A computed field is repeated based on the report component in which it is inserted.

> To create a computed field:

1 Select **Report > Insert Field**.

The cursor changes to a crosshair +.

2 Left-click and hold your left mouse button to position your cursor in the report component where you want to insert the field, and then drag to create the table.

A field labeled Empty Field is inserted.

3 Click the empty field to select it, and then enter JavaScript syntax in the Expression line.

You can copy and paste JavaScript syntax from existing report items into the Expression line to build the expression. To do this, select an existing item and copy the syntax in the Expression line. Then select the new field and paste the syntax into the Expression line.

4 When you have defined the equation, click the check icon |

The field displays the computed value.

Inserting Filter Values

You can select and show filters applied in the Query and Results sections to aid in the visual construction and production of the report.

- To use a filter value in a report:
- 1 Select Report > Insert Predefined Field > Query Filter or Result Filter.

The cursor changes to a crosshair +.

2 Left-click and hold your left mouse button to position your cursor in the report component where you want to insert the filter field, and then drag to create the field.

The Insert Filter Values dialog box is displayed.

3 In the Query or Result Sections list, select the section that contains the filters that you want to show.

A list of applied filters populates the Filter Names list.

4 Select the Filter Name that you want to use from the Filter Names list.

To show the operator used in the expression, select the Include Filter Operator check box.

5 Click **OK**.

Modifying Filter Fields

To modify the Filter field name or the values being shown, double-click the Filter field and modify the statement on the Expression line.

The JavaScript expression for the Filter Value field is displayed in the Expression line.



Working with a Report Page

The Report section gives you complete control of your report page setup and provides a number of features that assist you in designing effective, high-quality reports. Review the following sections for information on:

- Displaying Rulers
- Using Grids
- Using Design Guides

Displaying Rulers

Horizontal and vertical rulers help you line up items based on precise units of measure. Available units of measurement include inches, centimeters, and pixels, which you select by clicking the Measurement icon in at the intersection of the top and left rulers.

➤ To toggle the display of a ruler, select **Report** > **Ruler**.

A check mark is displayed next to Ruler to indicate the ruler is visible. Select this option again to clear the check mark and remove the ruler from view.

Using Grids

Use the layout grid to snap all objects to the closest grid spot automatically.

To toggle the display of the grid, select **Report** > **Grid**.

A check mark is displayed next to Grid to indicate the grid is visible. Select this option again to clear the check mark and remove the grid from view.

Using Design Guides

Design guides are horizontal and vertical lines that you place in your report to help you line up objects. Design guides are similar to grids in that objects automatically snap to align to the design guides.

If rulers are visible, click the ruler and drag one or more design guides from both the horizontal and vertical rulers.

> To toggle the display of design guides, select **Report** > **Design Guides**.

A check mark is displayed next to Design Guides to indicate they are visible. Select the option again to clear the check mark and remove the design guides.

Setting Up a Report

Use the Report Page Setup command to specify report page parameters including, page size, margins, and columns. Review the following sections for information on:

- Specifying Page Size
- Specifying Page Margins
- Setting Up Page Columns

Specifying Page Size

When deciding on page size for your report, consider the type of output you want. Do you want to print the report or view it on screen? You can select printer dimensions or specify custom dimensions for your report pages.

- > To specify page size:
- 1 Select Report > Report Setup.

The Report Setup dialog box is displayed.

- 2 Click the Page Size tab to view the Page Size page.
- 3 Select the page dimension option you want to use and click **OK**.
 - Use Printer Dimensions Determines the printable area based on your printer. If you want to only print the report, select the option.
 - Use Custom Dimensions– Enables you to specify the width and height of your report page.

If you export reports to HTML pages, you can use this setting to control the number of "pages" that are included in each HTML file. Each page is exported to a single HTML file by default. To ensure that all pages in your report are contained in a single HTML file, set the page height to a large number.

Specifying Page Margins

Page margins are the blank space that borders the report area on your page. Since the report area can hold a variety of objects, such as page numbers, text, and lines, change the margin area to accommodate the size of your report area. Margins are set for an entire report.

- > To specify page margins:
- **1** Select **Report > Report Setup**.

The Report Page Setup dialog box is displayed.

- 2 Click the Margins tab to view the Margin page.
- 3 Enter the Top, Bottom, Left and Right margin sizes and click OK.

Margins are measured in units of inches, pixels or centimeters. To change the margin units, click the Measurement icon in located above the Content pane until you find the measurement unit that you want to use.

Setting Up Page Columns

You can set up a multicolumned report and have data flow from the bottom of one column to the top of the next column. You can specify as many columns as you want.

- To create a multi-columned report:
- **1** Select **Report > Report Setup**.

The Report Setup dialog box is displayed.

- 2 Click the Column tab to view the Column page.
- 3 Select the number of columns that you want to include on the page.

If you want to include more than four columns on the page, enter the number of columns in the Other field.

The column width and spacing are determined automatically based on the number of columns specified, the page size, and the page margins. Column width and spacing measurements are set in either inches, pixels, or centimeters.

Enhancing Report Data

You can apply filters, sorts, computations, and break totals to refine the data in your reports. Review the following sections for information on:

- Sorting Report Items
- Adding Computed Items
- Applying Data Functions
- Hiding and Focusing on Reported Data

Sorting Report Items

Use the Sort icons to quickly sort a report group header or table column.

- ➤ To sort items quickly:
- **1** Drag the table column that you want to sort.
- 2 Click the Ascending or Descending icons on the standard toolbar.
- > To apply sort conditions using the Sort line for report group labels:
- 1 If the Sort line is not already displayed, click Sort on the Section title bar.

The Sort line is displayed below the Section title bar.

2 Click a report group label in the Content pane.

The Sort line should read Report Groupx, where x = 1, 2, 3, and so on.

- 3 Drag a report group label from the content area to the Sort line.
- 4 Double-click the name of the report group label on the Sort line to toggle between ascending and descending sort order.
- To sort table columns:
- **1** If the Sort line is not displayed, click **Sort** on the Section title bar.

The Sort line is displayed below the Section title bar.

- 2 Click anywhere within the table column (dimension or fact) in the Content pane.
- **3** Drag table dimensions or table facts from the Content pane to the Sort line.
- 4 Reorder the Sort items to determine the nested sort order.
- 5 Double-click the item in the Sort line to toggle between ascending and descending sort orders.
Adding Computed Items

Use the Add Computed Items command to build equations to compute totals, or to apply functions to existing values. Since a report may derive its values from a wide range of data sources (relational database queries, OLAP queries, imported data sets, and local-join queries), you select the Request item that you want to use from the applicable Results section.

- To add a computed item:
- **1** Select a table column and select **Report > Add Computed Item**.

The Computed Item dialog box is displayed.

Computed Item			
Column Name:			
Computed			OK
Reference			Cancel
Tables:	Columns:		
SalesResults 🗾 🔺	Year Quarter	_	Help
	Month Month Name		
	Product_Family Broduct_Category		
	Product_Line	•	
	Torritory	//	
Formula			
		A	
		T	
,			

2 In the **Column Name** field, type a name that describes the computation.

The default name is Computed.

3 In the Item pane, select the **Request** column from which to compute the new data item.

Note that the columns available for reference are only the columns that exist in the results set that is the source for the rest of the table.

4 Modify the existing syntax of the table column you selected or define a new computed item by building an expression in the Formula pane.

Interactive Reporting uses JavaScript to compute data items. The JavaScript engine supported in Interactive Reporting does not accept European or non-US number formats for computing data items. More information about JavaScript can be found in the *Interactive Reporting Object Model and Dashboard Development Services Developer's Guide, Volume 1: Dashboard Design.*

5 Click **OK** to apply the definition to the computed item column.

A new computed column is added to the Facts pane in the Table Data Layout for the selected table.

Applying Data Functions

A Dashboard enables you to determine the nature of the values represented in a Table Fact column in the Report section. Data functions retrieve underlying values and recalculate the value according to the type of data function that you specify. For example, the totals for a column are sums (the default data function) of the values in a column. You can use other data functions to change the totals to averages instead of sum.

Interactive Reporting provides a variety of prebuilt data functions that can be accessed from the Expression Line. You can also define your own data functions using JavaScript.

> To apply a prebuilt data function in the Report section:

1 Select a table fact.

2 Select Report > Data Function > Function.

The totals in the column are recomputed to reflect the selected data function.

The following table lists the prebuilt data functions available in the Report section.

Data Function	Description
Column Name	Returns the table column name.
Sum	Returns sum of underlying values. This is the default data function.
Average	Returns average of underlying values.
Count	Returns number of underlying values.
Maximum	Returns highest of underlying values.
Minimum	Returns lowest of underlying values.
% of Category	Returns group total percentage of the value.
Null Count	Returns number of nulls among underlying values.
Non-Null Count	Returns number of underlying values; null values are excluded.
Non-Null Average	Returns average of underlying values; null values are excluded.

 Table 45
 Prebuilt Data Functions in Report section

Applying Break Totals

- To apply a break total to a column in a report table:
- **1** Select a column and select **Report > Break Total**.

The Insert Break Total dialog box is displayed.

Insert Break Total	
At every break in:	
Amount Sales 🗾 💌	OK
Break total function:	Cancel
Minimum	
Add break total to:	Help
Unit Sales Year Quarter Region Product Line	Remove All

- 2 Select a break column from the At Every Break in drop-down list box.
- 3 Select the data function you want to apply from the Break Total Function drop-down list box.

See "Applying Data Functions" on page 5-35 for a list of data functions that you can use with break totals.

4 Select one or more columns on which to display the break total and click **OK**.

Using True Totals for Computed Items

Both report totals and break totals for tables can be calculated based on the column's data function applied to surface values, or the underlying formula of the column. This option is controlled by the "True Total Computation" feature.

Underlying values refer to values from the original Results section. Surface values refer to values in the actual report section. The two approaches produce values that may be displayed incongruous with the values in the report.

To understand this difference between underlying and surface values, consider a simple table with two values of 20 and 30. Each of these is already a total of underlying values (20 = 8 + 12 and 30 = 10 + 20). An average of the underlying value yields the result of 12.5 = (8 + 12 + 10 + 20)/4). An average of the surface value An average of the surface values yields the results 25 = (20 + 30)/2).

The True Total Computation enables you to standardize how total values in a computed column are calculated. When this feature is enabled, all total values are calculated based on the column's data function applied to the surface values. When this feature is disabled, all totals are calculated based on the underlying formula of the column.

Pre Release 8.3.2 tables (legacy) tables use the calculation method where one total is calculated by the underlying computed item formula for its computation, and the other total applies the column's data function to the surface values. You can change this behavior by also using the True Total Computation feature.

For legacy computed item columns, the "True Total Computation" feature shows the text "Legacy Total Computation" on the shortcut menu. This indicates that the column is using the inconsistent legacy total/break total computations.

- > To use true totals for the table total or break total in a computed column:
- 1 Select the computed column to which you want to apply the true total.
- 2 On the shortcut menu, select True Total Computation.

When the True Total Computation is deselected, totals and break totals are computed using the underlying formula of the column.

- > To use true totals for the legacy table total or break total in a computed column:
- **1** Select the legacy computed column to which you want to apply the true total.
- 2 On the shortcut menu, select Legacy Total Computation.

When you select a column, "Legacy Total Computation" is displayed on the shortcut menu. This means that the column is using an inconsistent total/break total computation.

Selecting the Legacy Total Computation from the shortcut menu disables the feature.

An Alert dialog box indicates that the change cannot be undone: "This action cannot be undone. Are you sure you want to use True Total Computations for Total and Break Totals?"

3 Click Continue.

True Totals are enabled for both Totals and Break Totals of this column.

The next time that you display the shortcut menu, "True Total Computation" is displayed as an option in place of "Legacy Total Computation". At this point, you can no longer apply the legacy behavior to this column.

Hiding and Focusing on Reported Data

A straightforward way to refresh your view of a report is to single out items for closer focus or remove some of the reported elements. This feature enables you to concentrate on particular items of interest. Hiding a group item removes the item from the report, but not from the Data Layout. Focusing a group item removes all items except the focused item from the report, but not the Data Layout.

Focusing on Items

- > To focus on a report item:
- **1** Select a group header and select **Report > Focus On Item**.

Interactive Reporting updates the report to focus on the data. A drill icon *** is displayed in the Data Layout next to the column you selected.

2 Select Chart > Focus On Items.

The chart is redrawn to display only the chart object(s) selected. A drill icon * is displayed in the Data Layout next to the item(s) on which you have focused.

Show All Items

Use this feature when you need to return the display to the original view.

To return to the original chart display, select Chart > Show All Items.

Hiding Items

Use this feature when you need to hide a selected item.

To hide a group item, select a group header and select **Report** > **Hide Item**.

You can only select one group item to hide at a time.

Interactive Reporting updates the report to hide the data. A drill icon *** is displayed in the Data Layout next to the column that you selected to hide.

To restore all hidden groups, Select **Report** > **Show Hidden Items**.

All items listed in the Data Layout are fully restored to view. The drill icon is displayed.

Using Multiple Data Sources in a Report

You can create reports that contain data from multiple queries that use a wide range of data sources (relational database queries, OLAP server queries, imported data sets, and local join queries).

In a sense, the data in a single table correlates data across queries in a report, retrieving all rows from the table on the "left" and any rows from the table on the "right" that have matching values. Unlike actual joined topics in the Query section, the tables are not linked in the same sense. In the Report section, data is only in the report.

The rules for using multiple data sources in a single report are:

- Report group headers require a common value that belongs to all results or table sets be included in the report. If you introduce a unique value belonging to one results or table set, nothing is returned for any band below that group header.
- Table Dimensions items can originate from only one data source.
- Values from multiple data sources can be included in Table Facts.
- The Results column that you place as a table determines the Facts used. For example, you have two Results sets:

Results 1	Region	Sales	Results 2	Region	Units
	East	10		East	100
	West	20		West	200
				South	400

If you place "Region" from Results 1 into the Table Dimension and then drop Results 1: Sales and Results 2: Units into the Table Facts, you get the following table:

Region	Sales	Units
East	10	100
West	20	200

- Note that the data from Results 2: South is not included.
- Smart charts and pivot tables added to a multiple data source report are subject to the same restrictions as described above.
- To build a multiple data source report:
- **1** Build the queries that you want to include in the report:
 - Verify item data types and associated data values in source documents so that you know how to correlate them when creating the reports.
 - Build the Request line, and add server and local filters, data functions, and computations to the query as needed.
- 2 Click Process.
- **3** Select Insert > New Report.

A new report section is displayed.

- 4 In the Catalog pane, expand the 🗄 🛅 button of the Results or Table set that you want to include in your report.
- 5 If necessary, click Groups on the Section title bar to open the Group Data Layout.

A group header categorizes data into repeating collections of records in a header band.

6 If necessary, click **Table** on the Section title bar to open the Table Data Layout.

The Table Data Layout shows two panes: Table Dimensions and Table Facts. Use the Table Dimensions pane to build the column dimensions (labels) in the report. Use the Table Facts pane to build the numeric values in the report.

- 7 Build the group header band by dragging the item(s) from the Results sections of the Catalog pane to the Groups Data Layout.
- 8 Build the column dimensions of the report, drag the item(s) from the Query sections of the Catalog pane to the Table Dimensions in Data Layout pane.
- 9 Build the column numeric values of the report by dragging the item(s) from the Query sections of the Catalog pane to the Table Facts in Data Layout pane.

Creating Smart Reports

Smart reports enable you to embed charts and pivot tables into a report body. These reports show only the data that is relevant to the report section in which they are placed. For example, if the report is grouped by year and you insert a chart in the report body, the chart replicates automatically so that there is a chart for each year of data in the report. Each chart contains data specific to that year.

- **Tip:** Smart reports are refreshed whenever you reprocess the query. Also, if you change a chart or pivot table in its home section, the Smart report is updated and the changes are reflected in the Report section.
 - > To embed a chart or pivot in a report:
 - **1** In the Catalog pane, drag an existing pivot table or chart to either the report group header, page header, footer, or report body.
 - 2 Size the pivot table or chart by clicking the embedded report until handles are displayed.

The pointer changes to a two-sided arrow.

Formatting Report Items

The following table lists formatting techniques you can use in the Report section.

Format	Description
Number	To modify the way numbers, dates and currency are displayed, select the item and select Format > Number . When the Number tab are displayed, apply any desired changes.
Style	To modify the appearance of a report item, select the item and select Format > Style > Option . Options include bold, underline, italic, and so on
Alignment / Justify	To modify the way an item is justified within a row or column, select an item, select Format > Justify > Option . Options include left, center, right, top, middle, and bottom.
Font	To modify the font of a report item, select the item and select Format > Font . When the Font tab of the is displayed, apply any desired changes.
Keep Together	Instructs not to split a band when a break is encountered. When a break is encountered, the entire band is moved to the next page.
Keep With Next	Instructs Interactive Reporting to keep bands within a group together when paginating a report. If the lower band cannot fit on the page when the report is paginated, both bands are moved to the following page.

 Table 46
 Report Formatting Options

Converting Detail Reports from Versions Earlier than 6.0

You can use the Report Designer to modify Detail reports created in documents prior to version 6.0.

Version 6.0 and later opens Detail reports in read-only mode. To work with the contents of a Detail report, first convert it to the new Report Designer format.

Note: You can convert Detail reports to Report Designer format only if the results set is saved with the document. If your results set is not saved with your document, reprocess the query and then select File > Save Results With Document.

To convert a Detail report to Report Designer format:

1 In the Section catalog, select the Detail report.

The detail report is displayed in the Content pane and a Detail menu is displayed on the menu bar.

2 Select Detail > Convert To Report.

Detail report is converted to the new Report Designer format and inserts a new Report section into the document. The original Detail report is left unchanged.

While every effort has been made to make the converted report as close as possible to the original Detail report design, you may need to clean up some reports, especially those that use computed items, multiple page headers/footers, or complex formatting.

Detail reports can be exported from the current version of the product to all of the previously supported formats, including Excel (.xls), Lotus 1-2-3 (.wks), tab-delimited text (.txt). comma-separated text (.csv), and HTML (.htm).

Display Differences

The page margins of a report created by the conversion process may be displayed smaller than in the original Detail report. This is because the new Report section is a WYSIWYG display that includes the unprintable area that exists outside of report margins. In version 5.x, Detail reports did not display or store information about the unprintable area. You can manually adjust the margin sizes in the converted report if needed.

If a Detail report contains multiple page header and footer sections, the current version of the product resolves these to a single page header and footer section during the conversion process. The height of the headers and footers in the converted report is equal to the combined heights of all headers and footers in the original Detail report. This may cause graphic and text objects in the headers and footers to overlap. You should manually adjust the properties, size, and/or position of these objects as necessary.

Conversion of Detail Report Categories

Detail Report categories from Detail reports are converted into report groups in the new Report section. The converted report displays Report Group Header and Footer sections if the corresponding category header and footer bands are visible in the original Detail report. The height of these header and footer sections remains the same. Detail report category labels are converted into JavaScript-based computed fields. Display properties remain unchanged.

Conversion of Data Area

To preserve as much of the original data and layout as possible, the conversion process translates the Detail report body information to a single table object in the body section of the converted report. This may lead to a loss of fidelity when converting complex or non-tabular style Detail reports.

Conversion of Facts

During report conversion, Detail report facts are added to the right of the dimension columns in the body section table. Number formats applied to Detail report facts are migrated to the converted report. If no number format exists, then the default number formats are applied to date, time, and number fields.

Because version 6.x uses JavaScript as the scripting, the conversion process does not automatically convert any computed facts contained in the Detail report. For computed items located in the Detail report body, the expression is omitted from the version 6.x table object. The conversion process may attempt to replace the computed column with another fact. If that occurs, simply delete the extraneous column.

For computed items located in a group header or footer, the code of the old scripting language expression is placed in a text field in the associated header/footer area to aid in manual conversion. Before you correct these placeholder items, Hyperion displays an error message in the field which reads <string>(1):unterminated string literal. The Detail report's *Surface Math* option has no equivalent in the converted report and is ignored.

Conversion of Smart Reports

Version 6.x is able to convert Detail reports containing multiple *smart* charts and pivot tables. Some converted smart charts may look different or even invisible once converted and displayed in the 6.x Report section. This is due to some changes in the chart section that affect the way that charts are plotted within a given sized area. As a result, embedded charts may need to be resized manually by the user.

Conversion of Graphic Objects

Certain graphic object properties are no longer supported and are ignored during the conversion process. These properties include the shadow, sunken button, and raised button fill effects.

The width of line objects in Detail reports is expressed in whole pixels. When a report is converted, the pixel line widths are converted to the nearest point equivalent.

Report Menu Command Reference

This table provides a quick reference to the commands available on the Report menu and lists any related shortcuts.

Command	Description	Keyboard Shortcut	shortcut Menu
Design Guides	Toggles the display of design guides.		
Grid	Toggles the display of grid lines.		
Rulers	Toggles the display of rulers.		
Section Boundaries	Toggles the display of section boundaries (bands).		
Page Margins	Toggles the display of page margins		
Sort	Sorts the selected column values in ascending order (alphabetical or numeric).		~
Data Function	Applies a prebuilt data function to the selected item.		~
Add Computed Item	Opens the Insert Computed Item dialog box.		~
Break Total	Opens the Insert Break Total dialog box.		~
Focus On Items	Updates the report to include only the selected data.		~
Hide Item	Hides the selected item from view.		~
Show Hidden Items	Restores the selected hidden item.		~
Show All Items	Updates the report to include all items removed by focusing.		~
Insert Table	Inserts a blank table in the report.		~
Insert Field	Inserts a blank field in the report.		
Insert Predefined Field	Inserts a predefined field in the report.		
Insert Graphic	Inserts a vector graphic in the report.		
Picture	Inserts a picture in the report.		
Remove Selected Items	Removes the selected item from the report.	[Del]	~

Table 47 Report Menu Commands

Table 47 Report Menu Commands (Continued)

Command	Description	Keyboard Shortcut	shortcut Menu
Headers and Footers	Toggles the display of report and page headers and footers.		
Report Setup	Opens the Report Page Setup dialog box for adjusting page size, margins, and columns.		

•

Using Filters

11

This chapter explains how to use filters in the Query and Results sections to refine data for more intelligent analysis and filter away the data you do not need. Review the following topics for information on:

In This Chapter

napter	About Filters	268
	Setting Simple Filters	273
	Setting Compound Filters.	276
	Setting Variable Filters	277
	Customizing Filters Options	277
	Filtering Data in a Table Report	278
	Multiple Filters and the Meta Topic	279
	Filtering Computed Items	280

About Filters

When building a query, you usually do not want to see information on every product you sell. Instead, you want to see information that relates only to a specific product or product line. Similarly, you probably do not want to see this information for every year the product has been available, but only for recent periods.

When you set a filter in the Query section, data is returned from the database only if it meets the specified conditions. For instance, suppose you only want to see customers who spend more than \$400,000 per year, or who buy gardening products in the Midwest. By applying a filter you are instructing the database to "give me only the data which satisfies the following conditions" (sales > \$400,000; or, state is in Midwest Region and Product Line = Garden).

For example, a filter placed on Item Type (which includes an "=" (equal) operator and value "Keyboard") returns only records associated with keyboard sales. Records associated with all other products are excluded from the Results set. The data set could be expanded to include modem sales records by adding the value "Modem" to the filter expression.

Similarly, the filter "> 5000" applied to the Amount Sold item filters out all sales transactions less than or equal to \$5,000. Alternately, the expression "between 5000, 10000" would exclude transactions above \$10,000 and eliminate any below or equal to 5,000.

Server versus Local Filter Processing

You apply filters in the Query or Results sections in basically the same way. The difference is in whether the filter is applied at the server or on your desktop.

Server Filters in the Query Section Local Filters in the Results Section

Server Filters in the Query Section

A filter applied in the Query section instructs the database server to filter unwanted information from your request. Only the filtered data is returned to your desktop.

If your query is potentially large, and you are more or less certain of both the information you need and how to define the correct filters, it is best to apply filters in the Query section. By using the database server to filter the data, you return only the data you need across the network and onto your computer.

Another advantage of applying filters in the Query section is that you can apply a filter to any topic item in the data model, even if the item is not on the Request line. For example, if you request State, Year and Units Sold, you can filter any of these items. But you can also place a filter on Operating System (OS) if it is displayed in one of the topics in the workspace. If you filter OS to the UNIX only, the server retrieves only sales information related to the UNIX operating system. You need not place OS on the Request line.

You can also filter computed items.

Local Filters in the Results Section

You can apply filters to columns to locally filter the data set in the Results section. Since the other reporting sections reference the results set, these local filters are disseminated to these sections.

Local filters are useful for managing your data set. If you do not need all information retrieved by your query, you can use a local filter to exclude data from view. Local filters are useful for temporary or hypothetical situations. You can always suspend or delete the filter to view the data and make it available for reporting.

Note: You can only apply local filters to items on the Request line.

Filter Line

Using filters involves dragging an item from the Content pane to the filter line and then setting filters in Filter.

The Filter line is a drag-and-drop command line similar to the Request and Sort lines. Filtered items are displayed on the Filter line. You can move, size, dock, and hide the Filter line.

Filter indicators on the Filter line and next to the item in the topic. Filter line expressions available in the Query section Filter line:



• Var – Indicates a variable.

() – Encloses suboperations.

- AND Retrieves data that meets both condition.
- **OR** Retrieves data that satisfies either of two conditions.
- () Encloses suboperations.

Var – Indicates a variable.

AND – Retrieves data that meets both condition.

OR – Retrieves data that satisfies either of two conditions.

Filter Line Functionality

In the Query section, the Filter line includes special functionality:

- The Filter line is an interface for building compound filter, which are multiple filters linked together to form complex filter equations. See Setting Compound Filters.
- The Filter line enables you to convert a filter to a variable filter, which prompts the user of the document to select filters as the query is processed. See Setting Variable Filters.
- When placing a filter on a request item computed with a data function, a divider is displayed on the Filter line, and the Filter icon is placed to the right of the divider. The divider indicates that the filter is applied in the SQL Having clause.
- Subqueries and correlated subqueries use the result from an inner query as the value of a filter in an outer query.

Filter Line Syntax

Syntax rules that apply to all Filter line expressions:

- By default, all equations are solved from left to right, with enclosed sub-operations evaluated first.
- AND is evaluated before OR.
- The AND operator retrieves data that meets both conditions. For example, if you query customers, and filter State to "Florida" AND Item Type to "Modem," the data retrieved would apply only to customers buying modems in Florida, not to modems bought in Minnesota or keyboards bought in Florida.
- The OR operator retrieves data that satisfies either of two conditions. For example, if you filter State to "Florida" OR Item Type to "Modem," the data retrieved would include Florida customers and any customers purchasing modems. It would not include customers purchasing keyboards (unless they lived in Florida), or customers in Minnesota (unless they bought modems).
- Suboperations allow you to override the default evaluation order, and may be required for certain operations involving both AND and OR operators.

Filter Controls

When applying a filter, you supply (or select) data values associated with a data item and use mathematical logic to apply the values as constraints.

	Limit: Product Name	
	Name: Product Name OK	
	Include Nulls Cancel	
	Not = Equal Ignore	
The Custom SQL feature is displayed only if you access Filter from the Query section.	Show Values 1001 Ways to Reward Emploid Custom Values A Course in Game Theory Custom SQL A Life of Picasso: 1907-1917 Select All A Love Supreme Transfer A Short History of Byzantium A Unique, 3D Tour of Archite A devanced A + Exam Cram Advanced	This feature is displayed as <i>Advanced</i> in the Query section and
	Loaded Values Settings O Load From File	<i>Options</i> in all other sections.
	C Load From Database	
	Create SubQuery Create a subquery with the current query as its parent.	Create Subquery is displayed only if you access Filter from the Query section.

Name – Descriptive name for the filter.

Include Nulls – Toggles the inclusion/exclusion of null values.

Not – Reverses the effect of an operator (for example, 'Not >=' is equivalent to <).

Using Operators – Comparison operators for the filter expression. Values that pass the comparison test will be included.

Edit field – Enter a value (or multiple values separated by commas), and click the check mark to add them to the custom values list to complete the filter definition.Click the "x" to erase the contents of the Edit field.

Note: The Edit field is displayed *only* if you are entering a custom value.

Show Values - Shows all potential values from the database that are associated with the item.

Note: Show values cannot be used in the Query section for filters on computed or aggregate computed metatopic items.

Custom Values – Lists potential values saved with the filter or read from a file. This feature enables you to select values from a predefined pool. You can create and save a custom list with each filter.

Custom SQL – Displays custom SQL for coding Filters directly in SQL.

Note: The Custom SQL button is displayed only if you access Filter from the Query section.

Select All – Selects all values displayed in the list of values.

Transfer – Adds selected values to the custom list.

Ignore – Temporarily suspends a filter without deleting it.

Fractional Digits – Specify the number of digits that you want to place to the right of the decimal point in an SQL numeric literal statement. The SQL statement uses this setting to determine which values to recognize. For example, if you type the filter value, 0.12345678, the SQL statement writes "where column = 0.12346" (the rounded value), but if you change the spinner to 8, then the SQL writes "where column = 0.12345678".

Advanced – Displays Loaded Values settings and subquery options.

Loaded Values Settings – Toggles a custom values list to be read from a file or from the database. Change File allows you to specify the file name. If you read values from a text file, vertical tabs or paragraph markers must defilter each value. Use Show Values to display the file contents.

Create Subquery – Creates a subquery.

Note: If you access Filter from the Results section, the *Advanced* button is displayed as *Options* and the Create Subquery option is not displayed.

Showing Values

The Show Values command is a powerful option for selecting values to define filters. Show Values provides a list of values actually in the database (or in the data set in Results) and allows you to choose a value based on the data available.

This feature makes it possible to set filters accurately without being familiar with the contents of the database. Show Values is also advantageous when values change frequently and custom lists quickly become outdated.

- ▶ To Show Values and define a filter:
- 1 Click Show Values in Filter to retrieve the item's values from the database.
- 2 Select values in the Values pane and click **OK**.
- **3** Click **Select All** or select values individually.

To create a snapshot of the values, click Select All and then click Transfer to move the variables to Custom Values.

- **Tip:** Because Show Values retrieves every value available, it is best not to use it in the Query section when the data item is large, consists mostly of values, or does not change frequently (for example, telephone numbers). In these situations, custom lists are sufficient and help avoid extra calls to the database. You administrator may disable Show Values if database hits are a concern.
- Note: Show Values cannot be used for filtering computed or aggregate items.

Defining Custom Values

Custom value lists are created by or supplied to Interactive Reporting. When users open the filter to choose new values, they can choose a filter from a custom list that has been supplied.

One reason to use custom lists with distributed documents is that many data items change rarely, if at all. For example, a Gender item has three consistent values (male, female, and unknown). A Product line item has many more values, but may only change every year or so. Data that changes rarely is said to have low cardinality. Under such circumstances, it makes sense for users to select from a custom values list, rather than continuously querying to show database values.

Custom lists are also useful when setting variable filters. When a document creator sets a variable filter, a custom list can be prepared and added to the document. The subsequent users who sets the filter can use the custom list or read more custom values in from an external file.

> To create a custom values list, use any method:

- Click **Show Values** in Filter to retrieve the item's values from the database, then select the values to be included in the custom list and click **Transfer**.
- Click **Custom Values** and enter values individually in the edit field, then click the **check mark** to add each to the custom list.
- Click Advanced and select Load From File to read values from a separate file. Use the Change File button to choose a file from the Select Filter File window. Click Show Values to load the values from the file to the list of values. Click Transfer to add selected values to the custom list.

Values added to the custom list can be selected to determine the filter. If the filter is converted to a variable filter, the custom list is stored in the document and the saved values are available to users who resolve the filter.

Tip: Imported values must be vertical-tab or paragraph delimited.

Using Operators

Logical operators for defining filters support standard SQL wildcards, including "%" and "_".

Operator	Description
Equal (=)	Retrieves records where the filtered item equals the specified value(s).
Not Equal (<>)	Retrieves records where the filtered item does not equal the specified value(s).
Less Than (<)	Retrieves records where the filtered item is less than the specified value(s).
Less or Equal (<=)	Retrieves records where the filtered item is no greater than the specified value(s)
Greater Than (>);	Retrieves records where the filtered item is greater than the specified value(s).
Greater or Equal (>=);	Retrieves records where the filtered item is no less than the specified value(s)
Begins With	Retrieves records where the filtered item begins with the specified value(s) up to and including the end value.
Contains	Retrieves records where the filtered item contains the specified value(s) regardless of location.
Ends With	Retrieves records where the filtered item ends with the specified value(s).
Like (with wildcards)	Retrieves records where a text string is displayed and reflects the placement of the specified value(s).
	For example, Name Likes_ would retrieve records for all employees whose names have the letters Ze followed by one character at the end
s Null	Retrieves records where the filtered item has no value; for example a field in which no data has been entered.
Between	Retrieves records where the value of the filtered item lies between (and does not equal) the specified values.
Not (with operator)	Negates the operator it precedes, reversing the results of the equation

 Table 48
 Logical Operators for Defining Filters

Note: Text strings are evaluated differently among database systems, and are often case-sensitive (that is, "RED", "Red" and "red" may not be equivalent in comparison). Text characters are generally valued in ascending order (0 1... 9 A B C... Z a b c... Z.), so that "A" is less than "a" in string comparisons, and strings beginning with numbers are considered "less" than strings beginning with alphabetic characters. See your administrator for specific information.

Setting Simple Filters

Simple filters are applied in basically the same manner in the Query and Results sections. If you are working with a distributed document, some of the buttons in the Filter dialog box may not be available.

- Filtering Queries
- Filtering Results
- **Note:** In the Interactive Reporting Web Client, you can only add filters in the Query section if you have the "Data Model, Query, and Analyze" adaptive state.

Filtering Queries

The simplest way to filter a query is to apply filters individually to topic items.

- ► To filter a query:
- 1 Select an item in the data model and choose Query > Add Filter(s).

Filter is displayed.

- 2 Select an arithmetic or logical operator from the list.
- **3** Define the potential filter values by clicking one of the following options:
 - Show Value– Shows database values associated with the item.
 - Custom Values Supplies an empty field for inputting custom values. Click the check mark to add a value to the list of values. You can also display values from a previously saved custom list or values loaded from a file.
 - Custom SQL Supplies an empty box for typing an SQL clause to be included in the query statement. Choose View > Custom SQL to check the complete statement logic.
- 4 Select the values to include in the filter definition in the Values list.

Individually select values or click **Select all** and deselect the values you do not want to include. Only selected items are applied to the filter definition. To create a snapshot of the values, click **Select All** and then click **Transfer** to move the variables to the Custom Values.

5 When the correct values are highlighted in the Values list, click **OK**.

The filter is applied to the topic item and an icon is added to the Filter line.

In the Query section, the filter is applied when you process the Query.

Note: When placing a filter on an aggregate item (computed with a data function) in the Query section, a filter divider is displayed and the filter icon is placed to the right. The divider indicates the filter is in the SQL Having clause. You must drag the aggregate item from the Request line to the Filter line to create a filter on an aggregate item.

Removing a Query Filter

To remove a filter on a query, select an item on the Filter line and select Remove on the shortcut menu.

Filtering Results

- > To filter the display of data in the Results section:
- 1 Select a column (click the column heading) and choose Results > Filter.
- 2 Filter is displayed.

Limit: Year		
Name:	Year	ОК
	Include Nulls	Cancel
🗖 Not	= Equal	Ignore
Show Values Custom Values	1999 2000	Help
Select All Transfer		
	*	Options

- **3** Select an arithmetic or logical operator from the list.
- 4 Define the potential filter values by clicking one of the following options:
 - Show Values Shows database values associated with the item.
 - Custom Values Supplies an empty field for inputting custom values. Click the check mark to add them to the list of values. You can also display values from a previously saved custom list or values loaded from a file.
- 5 Select the values to include in the filter definition in the Values list.

Individually select values or click **Select all** and deselect the values you don't want to include. Only selected items are applied to the filter definition.

6 When the correct values are highlighted in the Values list, click **OK**.

An indicator is added to the Filter line and the filter is immediately applied to the data set.

Removing a Results Filter

> To remove a local filter, choose an item on the Filter line and select **Results** > **Remove**.

Setting Compound Filters

In some cases, you may want to set two distinct filters on the same Request item or create compound constraints using more than one item.

Use the Filter line to build compound filter expressions. The Filter line enables you to apply more than one filter to one item, or create compound conditions dependent on more than one constraint.

Drag an item to the Filter line more than once and apply different logical operators to create a complex constraint. For example, to retrieve dollar values greater than \$100 or less than or equal to \$10, drag the Amount item to the Filter line twice, and set two separate filters. An AND operator is placed between the icons on the Filter line and the data set is constrained by both conditions.

Similar logic can be applied using two filters and substituting the OR operator. For example, to retrieve customers in the Cleveland area and those who have purchased more than \$100,000 worth of goods, you can place a filters such that *City* = *Cleveland* and *SUM*(*Sale Amount*) > \$100,000 and join them with the OR operator.

- Note: The second instance of an item on the Filter line displays a "_2" next to the item name.
 - > To create compound Filter line expressions:
 - 1 Add two or more items to the Filter line and apply individual filters using Filter dialog box.

An AND operator is displayed between each item on the Filter line.

2 In the Query section only, click the small arrow at the left edge of the Filter line.

The Filter line is adjusted to display the Filter line control buttons.

- **3** On the Filter line, select filter controls to complete the equation.
 - To switch Boolean operators AND and OR, double-click the operator you want to change.
 - To enclose suboperations, select the items you want to enclose and click the parentheses button. To remove parentheses, select a parenthesis and click **Remove** on the shortcut menu.
 - With an item selected on the Filter line, click **Var** to make the filter variable.

When using compound filters, verify that the expression delivers the correct results.

- **Tip:** The following syntax rules apply to all Filter line expressions. When creating a compound filter, be sure to verify that the expression is delivering the correct results.
 - The AND operator retrieves data that meets both conditions. If you want to retrieve data which satisfies either of two conditions, use the OR operator.
 - By default, equations are solved from left to right, with enclosed sub-operations evaluated first. AND is evaluated before OR.
 - Sub-operations allow you to override the default evaluation order, and may be required for certain operations involving both AND and OR operators.

Setting Variable Filters

A variable filter is a filter in the Query section that is resolved only when a query is processed. At that time the user is prompted to select or enter filter values and complete the constraint. You can use variable filters in standardized documents and distribute them to users to supply different filter values for each process.

Variable filters work particularly well with custom lists. If a custom list has been saved with a variable filter, the user can respond to the prompt by selecting a value from the custom list.

For example, you may have a document you use monthly to monitor inventory levels. Each time you use the document, you run it separately for each product line you carry. You can accelerate the process by making the filter variable on the product line item, and create a custom values list. Each time you process the document, you can select a new product line without redefining filters.

- **Note:** In the Interactive Reporting Web Client, variable filter functionality is only available if your document has query and analyze privileges.
 - > To set a variable filter:
 - 1 If necessary, add a topic item to the Filter line and define a filter.

If you are providing a custom list, add all the values you want to make available to the list of values.

- **Note:** You must select at least one value in the list to save the filter, even if no custom list is provided and the user clicks Show Values to choose from database values. This selection does not influence the values available to the variable filter, which offers all values in the database or in the custom list.
 - 2 Select a topic item on the Filter line.
 - **3** Choose **Query > Variable Filter**.

The Filter item is displayed with a V(1) beside the item name to indicate it is a variable filter.

If other items are set to variable filters, they are displayed with V(2), V(3), and so on, to indicate the order in which the user is prompted to respond to Variable Filter when the query is processed.

Customizing Filters Options

Customize Filter enables you to control access to features, which can be especially useful when you distribute documents to end users. For instance, if you plan to distribute a document to novice users who must set a simple variable filter, it may be preferable to disable or even remove such features as Include Null or the Custom Values buttons.

Note: Customizing affects one filter. Setting global filter preferences that restrict the options available throughout a distributed data model is an advanced feature not covered in this book.

- > To customize filter options:
- 1 Select an item on the Filter line and choose Query > Customize Filter.

Customize FilterCustomize Filter is displayed.

- 2 Customize the filter as follow and click **OK**.
 - Title Add a title or text to the filter dialog to instruct users on how to set the filter or on what the values mean.
 - **Prompt** Supply explanatory comments or instructions.
 - Values Disable Show Values to reduce database hits or remove the Custom SQL or Custom Values option.
 - Options Disable elements and lock in operators.
- 3 Click OK.

Filtering Data in a Table Report

You can set filters on columns in a table report to filter the data displayed. Table section filters are applied in addition to the filters set in the Results section. Filters set in the Table section are automatically propagated to the other reports that inherit their data set from the Table. You can always suspend or delete filters to return data to the display and make it available for reporting.

The Status bar displays the total number of rows in the table report. You can apply one filter only per column.

- > To filter data in a Table report:
- 1 Double-click the column in which you want to place a filter.

Filter is displayed.

- 2 Select a logical operator from the shortcut list.
- 3 Complete a filter definition by supplying constraining values.
 - Create a Customs Values list by supplying a value (or values, separated by commas) in the Edit field, and clicking the check mark.
 - You can also click Show Values to display column values and select one or more values depending on the comparison operator.
- 4 When the values are highlighted in the values pane, click **OK**.

The filter is applied to the column and the column name is added to the Filter line.

- To remove a filter in a Table report, select the filter item that you want to remove, and select Remove on the shortcut menu.
- To remove all filters in a Table report, select Filter on the Filter line and then select **Remove** on the shortcut men.

Multiple Filters and the Meta Topic

Filters are a very important part of the querying process, helping to refine requests, filter unneeded information, and reduce system resource consumption to manageable levels. One way to further increase the reach of filters in a managed query environment is to think of them in terms of both the query and the underlying Data Model, and of ways in which fixed and variable filters may be used together.

Fixed filter values are defined with a query, and remain static until changed manually. Variable filters prompt for new values each time a query is processed. In certain instances you may want to apply both types of filter to one item in order to restrict the available variable filter values and simplify users' choices, or to grant users the latitude of variable filters only within a prescribed range.

For example, consider users of a large data warehouse or legacy system containing sales data going back for many years. For practical or security reasons the administrator might choose to confine some users' access to information dating after 1990. However, it is still important for these users to query the database on-the-fly, and retrieve a variety of data for analysis.

This situation requires two filters, one fixed and one variable, on one topic item. The fixed filter qualifies the values set available to the user in the variable filter. The fixed filter on a date column item instructs the server not to return any values < 1/1/90. The variable filter on the same date column subsequently furnishes a list of server values > 12/31/89 for the user to choose when the Show Values command is invoked in Filter.

The Two-Tier Strategy

Double-filters can be implemented by dragging an item to the Filter line twice. However, one deficiency to this method is that when applying multiple filters on one item, browse level settings are not effective; that is, the server values retrieved are not constrained by another filter on the same item, because both filters are defined using the same pool of server values. Nevertheless, an elegant solution to this issue is available through the use of the Interactive Reporting StudioInteractive Reporting Web Client meta topic, which enables the administrator to establish filters at different logical levels.

Interactive Reporting StudioInteractive Reporting Web Client meta topics are logical views which overlay the original tables-and-joins model of a database. Items that display in meta topic view inherit any modifications made to the original items in the underlying physical topic view. In this instance, a fixed filter can be applied to the date item in the original database topic. Once a meta topic, which includes the date item, is created to serve as the user view of the Data Model, a variable filter is applied to the date item in the meta topic.

Since the fixed filter is applied to the original date item, the meta topic date item incorporates this filter transparently. The values excluded by the fixed filter are not associated within the new meta topic, and are not returned by the Show Values feature for fixed or variable filters applied to the new meta topic item. Note also that Filter Browse Level settings need not be modified under this scenario because the filters in question are applied to different logical layers.

Filtering Computed Items

In the Query section, you can set a filter on a computed item that resides on the Request Line and invokes an SQL HAVING clause when you process the query.

When you create a computed item, its only reference is through the Request Line. In the same way that you can add an item to Request Line by drag and drop, you can drag and drop a Computed Item onto the Filter Line to prompt for Filter. If the item uses an aggregate function, it displays to the right of all other filters, indicating its use in the HAVING clause of the SQL statement.

- > To create and set a filter on a computed item in the Query section
- 1 Click an item on the Request line and select **Query > Add Computed Item**.

Computed Item is displayed.

2 Enter the information in Computed Item and click OK.

The computed item displays on the Request line under a new name.

- 3 If the Filter line is not already displayed, click the Filter button on the Section Titlebar.
- 4 Drag the computed item from the Request line and drop it in the Filter line.

Filter is displayed.

5 Optional: Enter a descriptive name for the filter in the Name field.

The name is displayed on the filter line icon and can help differentiate multiple filters set on the same item. The name is also used when scheduling a document.

- 6 Select an arithmetic or logical operator from the list to begin defining a filter expression.
- 7 Define a pool of potential filter values by entering a filter value in the edit field and clicking the check-mark to add each to the values list pane.
 - You cannot use the **Show Values** command since Interactive Reporting Studio cannot retrieve a values list for a computed or aggregated request item from the database.
 - Click Custom Values to supply the values in the edit field and click the check-mark to add them to the values pane. You can also display values from a custom list previously saved with the document or loaded from a file.
 - If you are familiar with SQL, click **Custom SQL** and type an SQL Where clause to be included in the query statement. You can open the Custom SQL dialog to check the complete statement logic.
- 8 Select the values to include in the filter definition in the values pane.

Select values individually or click **Select all** and then deselect those you do not want to include. Only selected items are applied to the filter definition.

9 When the correct values for the filter expression are highlighted in the values pane, click **OK**.

An indicator is added to the Filter line. In the Results section, the filter is applied to the data set immediately. In the Query section, the filter is applied when you process the Query.

When placing a filter on a Request item computed with a data function in the Query section, a divider is displayed on the Filter line, and the filter icon is placed to the right of the divider. The divider indicates the filter is applied in the SQL Having clause.

10 Click **Process** to achieve the final results set.

Filter

Use Filter to apply filters, which refine your data set by excluding unneeded data. When applying a filter, you supply (or select from a list) data values associated with a data item, and use mathematical logic to apply the values as constraints.

- **Tip:** All filters are applied through Filter. Filters apply only to the chosen data item. You can combine filters into more complex Compound Filters using the Filter Line.
- **Note:** Filters apply only to the chosen data item. You can combine filters into more complex Compound Filters using the Filter line.

Customize Filter

Use Customize Filter to control access to filter features. This can be especially useful when you distribute documents to end users. For example, if a document is to be distributed to novice users who must set a simple variable filter, it may be preferable to disable or even remove such features as Include Null or the Custom Values buttons.

Feature	Description	
Title	Add a title or text to the Filter to instruct users on setting the filter, or describe the value.	
Prompt	Supply explanatory comments or instructions.	
Values	Disable the Show Values option ro reduce database hits, or remove the Custom SQL or Custom Values options.	
Option	Disable elemtns and lock in operators.	

Table 49 Customize Filte	Table 49	Customize	Filter
--------------------------	----------	-----------	--------

12

Working with Computed Items

This chapter describes how to use calculations to compute new data items in the Query, Results, and reporting sections. Such calculations are important for supplementing the information already stored in the database.

In This Chapter	About Computed Items	284
	Adding Computed Items	289
	Using Functions	293
	Common Computed Item Examples	304
	Scalar Function Examples	307
	Trend Functions	324

About Computed Items

When looking for answers to basic business questions, raw data cannot always help. For example, while a database might track dollar sales figures, sometimes this information is much more valuable with reference to more complicated calculations such as cost of sales or profit margin.

Administrators sometimes anticipate these requirements, designing the internal database structure so that it makes such calculations as the raw data is collected. But at times you might need information that no one has yet anticipated.

Computed data items are a means of generating new information, usually from data that is already stored in the database. If your database doesn't offer particular information, use Interactive Reporting to compute it for yourself.

Computing New Data

Computations can be performed by your database server or by Interactive Reporting on your desktop. A computation does not add data to the database; instead, new data items are added or data items are recalculated in your data set.

Interactive Reporting provides different tools for computing data items. These tools allow you to:

- Build arithmetic expressions {Units * Amount = Revenue}
- Build logical expressions {If Score > =50 Then "Pass" Else "Fail"}
- Build mixed expressions {Tax = Revenue *.35)
- Apply functions { (abs (Amount_Sold) }

Examples of computed items include:

- A Full_Name item that concatenates the values in the First_Name and Last_Name items.
- A Profit item derived by subtracting the Cost of Goods column from the Gross Revenues column.
- A Grade item that uses if...else logic to assign letter grades derived from test scores.
- A Sine item that computes an item for the sine of an angle.
- A Moving Average item that smooths a set of data points, i.e., removes irregular fluctuations in the general trend of data.

Computed Items in Sections

You can add computed items in the Query, OLAPQuery, Results, Table, Chart, and Pivot sections. This feature works similarly in all sections with a few differences.

In all Interactive Reporting sections, you can use computed items to create a completely new data item on the Request line from an already existing data item. Each value in the original data item is computed to produce a new value in the newly generated data item. A one-to-one correspondence exists between the original values and the derived computed values.

Computed Items in the Query Section

In the Query section, a computed item is a set of instructions to the database server. Interactive Reporting uses the computing power of the database server to perform calculations as it retrieves data from the database.

For this reason, the Query section allows you to use computed items in a way that is not possible in the other sections. Instead of creating a new data item, the new values simply replace the original values in the data item as they are retrieved from the database.

Additionally, you can compute items using any topic item in the data model and any scalar functions provided by your RDBMS.

Since computed items are new data items, you may want to confirm or change the new item's data type to preserve the precision of a mixed-data type computations, or to change the way a data item is handled (for example, interpreting number as strings). This ensures the correct handling of data in server computations.

Local calculations (Results, Pivot) are handled internally, and adjustment between 16- and 32bit integers, for example, can be handled safely using the automatic or number data type specification. For a list of Hyperion-supported data types, see Table 50.

Data type	Specification
Automatic	A data type is determined automatically given the data type of the reference items and the computations performed.
Byte	Variable data type of length determined by a single byte of computer storage. Bytes can store numeric values from 0 to 255, or a single text character.
Date	Calendar date in server default format (typically mm/dd/yy).
Integer (16-bit)	Retains a 16-bit value (2 bytes). A 16-bit integer stores integer values from 0 to 16,777,216, and signed integers between +8,388,608 and -8,388,608.
Integer (32-bit)	Retains a 32-bit value (4 bytes). A 32-bit integer has a range of 0 to 4,294,967,296 if unsigned. If signed, -2,147,483,648 to 2,147,483,647.
Packed Real	Real numbers packed for use with EDA middleware. The results in Interactive Reporting Studio are the same as real numbers.
Real	Decimal numbers up to 5 positions right of the decimal.
String	Text strings to a maximum length of 256 characters.
Time	Time in format set by user preference.
TimeStamp	Date/time combination in format set by user preference.

Table 50 Data Types and Specifications

Computed Items in the Results and Reporting Sections

In the Results and reporting sections, Interactive Reporting performs computations on the desktop. The computations involve only the data in your Results set or on the surface of a reporting section.

In these sections you can only create new computed items: you cannot modify original data items retrieved directly from the database. For the same reason, scalar functions used to compute items in these sections are provided by the Interactive Reporting, rather than the RDBMS.

Computed items in the Results and reporting sections differ in two respects:

- In the Results and Table sections, reference items are filtered to the items that are displayed on the Request line.
- In the remaining reporting sections (excluding the Report Designer section), reference items are limited to the items placed in the Fact pane in the Data Layout. Computations in these sections work on the aggregated cell values that make up the core of the report. To perform computations on data before it is aggregated, compute the new item in Results.
- In the Report section, the break totals of a table can be calculated

Computed Items in the Report Section

In the Report section, both report totals and break totals can be calculated based on the column's data function applied to surface values, or the underlying formula of the column.

Underlying values refer to values from the original results section. Surface values refer to values in the actual report section. The two approaches produce values that may be displayed incongruous with the values in the report.

To understand this difference between underlying and surface values, consider a simple table with two values of 20 and 30. Each of these is already a total of underlying values (20 = 8 + 12 and 30 = 10 + 20). An average of the underlying value yields the result of 12.5 = (8 + 12 + 10 + 20)/4). An average of the surface values yields the results 25 = (20 + 30)/2).

In the following example, both the Computed and Computed2 columns have the Average data function applied. The Computed column is calculated using the data function applied to the surface values. The Computed2 column is calculated using the underlying values

Store	Amount	Computed	Computed2
Computer Central	1,982.66667	1,982.66667 1,982.6	
Total Computer Cen	1,982.66667		
Computer City	1,883.53333	1,883.53333	1,883.53333
Total Computer City	1,883.53333		
Computer Super Sto	1,784.4	1,784.4	1,784.4
Total Computer Sup	1,784.4		
Computer Town	892.2	892.2	892.2
Total Computer Tow	892.2		
Computer World	1,784.4	1,784.4	1,784.4
Total Computer Wor	1,784.4		
Discount Electronics	892.2	892.2	892.2
Total Discount Elect	892.2		
Grapevine Periphera	1,189.6	1,189.6	1,189.6
Total Grapevine Per	1,189.6		
Hi-Tech Mart	1,784.4	1,784.4	1,784.4
Total Hi-Tech Mart	1,784.4		
Power Computers	1,487	1,487	1,487
Total Power Compu	1,487		
Wolfe"s Disc. Electro	2,200.76	2,200.76	2,200.76
Total Wolfe"s Disc.	2,200.76		
	15,881.16	15,881.16	1,625.78667

If you are calculating totals and break totals for a BQY created in Release 8.3.2 and early (legacy computed items) you can still calculate totals based on the column's data function applied to surface values, or the underlying formula of the column.

For legacy computed item columns, the "True Total Computation" is changed to "Legacy Total Computation".

If the user selects "Legacy Total Computation" from the speed menu (essentially trying to turn it off), the user will get an alert dialog indicating that the change cannot be undone ("This action cannot be undone. Are you sure you want to use True Total Computations for Total and Break Totals?" ... Continue/Cancel). If the user chooses "Continue", then "True Totals" will be enabled for both Totals and Break Totals of this column. The next time they display the context menu it will show "True Total Computation" option in place of "Legacy Total Computation". The user may no longer apply the legacy behavior to this column.

Computed Items in the Pivot Section

When you analyze and interpret Pivot Section computed item break totals, two methods are available for you to present the data. These methods include:

- Break total cell results derived from the computed item formula applied to the detail cell
- True Computed Item totals

When determining which method to use, look at how the two types of totals differ from each other. The former method is derived from the formula used to calculate the detail cell.

True computed item totals use aggegation according to the specified data function and they never rely on the computed item total formula. Depending on what you are trying to achieve, each method will show different totals

You can see how the two types of totaling computed items differ in the example below. The first part of the example shows what happens when the totals are derived from the computed item formula used in the detail cells.

Detail cell data is shown at the City level and break totals are shown when the State changes.

		Units	Computed	
CA	Burbank	2,395	46-	Detail cells
	Glendale	990	41	
	Oakland	910	11	
	San Diego	7,400	1	
	Westwood	695	46	
	Total	12,390	41	
NY	New York	4,640	41	
	Rochester	1,615	16	
	Total	6,255	6	Break Total Cells
Total		18,645	46	

In this example, the *Sum* data function has been applied. The "Computed" column is defined by the following formula:

(Units % 50) +1

where "%" represents modulo (remainder) operator. In other words, the formula is defined as:

Integer remainder of ("Unit column cell value" /50) + 1

For the Unit column values for each city within a state, the formula works as expected. For example, in the Oakland, CA cell, the formula is:

```
Units 910
Modulo (remainder) of 910 / 50 = 10
Add 1 to assign a value of 11 (shown above).
```

For the California "Total" row, the value shown is 41, which is the result of the following formula:

```
Total "Units" for California = 12390
Modulo of 123900/50 = 40
Add 1 to assign a value of 41
```

The Modulo of 41 is not the sum of the displayed cell values for all cities in California, instead it is the modulo formula applied only to the cell containing the "Unit" column city total for California.

If you expected to see a break cell total value of *145*, you need to use the True Computed Item feature total, which would reference the displayed values in the detail cells (this example assumes a *Sum* data function):

46 + 1 + 11 + 1 + 46 (= 145

Using Surface Values

Interactive Reporting enables you to use underlying or surface values when working with totals. Underlying values refer to values from the original results section. Surface values refer to values in the actual report section. The two approaches produce values that may be displayed incongruous with the values in the report.

To understand this difference between underlying and surface values, consider a simple pivot table with two values of 20 and 30. Each of these is already a total of underlying values (20 = 8 + 12 and 30 = 10 + 20). An average of the underlying value yields the result of 12.5 = (8 + 12 + 10 + 20)/4). An average of the surface values yields the results 25 = (20 + 30)/2).

- **Note:** Average and Count aggregation Data Functions are not evaluated in True Total mode unless the Use Surface Values property is also enabled (see Use Surface Value above). If Use Surface Values is not enabled, the Average and Count aggregation are calculated using the count of the underlying Table/Result Section data values instead of the displayed Pivot values.
- Note: By default, the Surface Values feature is deactivated.

Computed Items and Data Functions

Computed items and data functions are fundamentally different, and the functions available in the Computed Item dialog box do not calculate data in the same way as data functions.

- Computed items calculate a fresh value for each original value, based on the computation (for example, *Revenue* calculated from *Price* and *Units Sold*). The new values are part of a new data item or replace the original values. Computed items *never* reduce the original number of records.
- Data functions, by contrast, summarize groups of database records and replace the original values with new summary data. Because data functions summarize values, the number of records are frequently reduced.

Pivot Options

The Pivot Options tab allows you to enable and disable surface values, how to handle true computed item totals, and how to treat null fact values. For more information, see "Selecting Pivot Table Elements" on page 174.

Adding Computed Items

In the Query section, a computed item is a new data item derived from calculations. In the Results, OLAPQuery, Pivot, Chart and Report Designer sections, you add computed items by building equations to compute data items, or by applying functions to existing data items.

Computed items are like normal data items, and can be included in reports or reused to compute other data.

- To create (or modify) a computed item:
- 1 Select Add Computed Item from a Section menu (for example, Query, Results, and so on).

Modify Item Name: Computed OK Cancel Definition Function . Refere Computed Item Name: Computed ΰK Optio Cancel He Definition Functions. Reference · * / () Options Help Local Meta-topic and or if Server Level Level Operators Operators

The Computed Item or Modify Item dialog box is displayed.

2 In the Name field, type a name that describes the computation.

The default name is Computed, which is numbered sequentially if there is more than one. If you assign a name to a computed item that is identical to an existing scalar function name, the Interactive Reporting numbers the name starting with the number 2.

- 3 Define the new data item by building an expression in the Definition text box.
 - Use the operator buttons to insert arithmetic and logical operators at the insertion point.
 - Click Functions to apply scalar functions using the Functions dialog box.
 - Click **Reference** to display the Reference dialog box, and select items to place in the equation.

You also can type any portion of the equation or the entire equation directly into the Definition text box using JavaScript. The names are case sensitive, and you must replace spaces in item names with underscores ('_').

- 4 If necessary, click the **Options** button to set a new data type for the item.
- 5 When the equation is complete, click **OK**.

The computed item is listed in the Results Data Layout and is displayed as a column in the results set.

In the Query section, the computed item is displayed on the Request line with its new name. In the Results section, computed items are displayed in the Data Layout in blue type.
Operators

The following sections describe the operators available for creating computed items. Use the following guidelines as you add operators to your computation.

- Type the word null (no quotes) into the Expression text box to represent null values.
- Enclose all text string constant values and date constant values entered in expressions in single quotes. (Numbers can be entered without quotes.)
- To join items with a space or other character, reference or type items and strings into the Expression text box and join them with the + operator (for example, City + `, ' + State). To join without additional characters, use the Concat function.
- In division operations, the divisor may not be null or equal to zero. If a data item serves as the divisor in an expression (for example, 5000 / Units_Sold) and includes null or zero values, first create a computed item using if/else logic to remove null and zero values, and then compute the item containing the division operation.
- Two date items can be subtracted, but not added. The Add Month function adds an integer value to a date.
- You cannot nest functions inside the Sum, Cume, Chr, and Breaksum functions.

Arithmetic Operators

Arithmetic operators take numerical values (either logical or variables) as their operands and return a single numerical value.

Operator	Name	Description
+	Add	Used at both the server level and the local metatopic level for all sections.
-	Subtract	Used at both the server level and the local metatopic level for all sections.
*	Multiply	Used at both the server level and the local metatopic level for all sections.
/	Divide	Used at both the server level and the local metatopic level for all sections.
(Begin suboperations	Used at both the server level and the local metatopic level for all sections.
)	End suboperations	Used at both the server level and the local metatopic level for all sections.
++	Increment	Used at both the server level and the local metatopic level for all sections except the Query section.

 Table 51
 Arithmetic Operators

Table 51Arithmetic Operators

	Decrement	Used at both the server level and the local metatopic level for all sections except the Query section.
Mod (%)	Modulus The modulus operator returns the remainder of dividing var1 by var2. For example, 5% 4 returns 1.	Used only at the local metatopic level.

Comparison Operators

A comparison operator compares its operands and returns a logical value based on whether the comparison is true. The operands can be numerical or string values. When used on string values, the comparisons are based on the standard lexicographical ordering. The comparison operators in Table 52 are only available at the local metatopic level.

Table 52 Comparison Operators (Local Metatopic Level)

Description
Returns true if the operands are equal.
Returns true if the operands are not equal.
Returns true if the left operand is less than the right operand.
Returns true if the left operand is less than or equal to the right operand.
Returns true if the left operand is greater than the right operand.
Returns true if the left operand is greater than or equal to the right operand.

Statements

Executes a set of statements if a specified condition is true. If the condition is false, another set of statements can be executed.

Note: Statements are only available at the local metatopic level.

Table 53 if...else Statement

Statement	Description	
ifelse	if executes a set of statements if a specified condition is true. The specified condition may be another statement and can include other nested if statements. Braces, {}, must enclose multiple statements. If the condition is false, another set of statements can be executed if the optional else statement has been included in the script.	
	A sample if else statement looks likes this:	
	if (condition) {	
	statements1	
	}	
	else {	
	statements2	
	}	

Logical Operators

Logical operators take Boolean (logical) values as operands and return a Boolean value. The logical operators in Table 54 are only available at the local metatopic level.

Table 54 Logical Operators (Local Metatopic Level)

Logical Operator	Description
AND (&&)	Connects two conditional expressions and retrieves records only if each expression is true.
OR ()	Specifies a combination of expressions and retrieves records that include at least one of the expressions. For example, if one of the words is Washington or Oregon, every record with the expression "Washington" and every record with the word "Oregon" is included.
NOT (!)	Computes and shows items more accurately stated in a negative way. In effect, all records are retrieved except those that fulfill the conditional expression.
	You enter the conditional expression with the NOT (!) logical operator preceding the conditional expression. The conditional expression can be a simple value or nested within other conditional expressions, for example, expressions using AND and OR.
	A combined condition expression that uses NOT is true if the conditional expression following NOT is false. A combined conditional expression is false if the conditional expression following NOT is true.

Using Functions

This section lists the functions available in Interactive Reporting Studio, including:

- Scalar Functions
- Teradata Version 3 OLAP Functions
- Functions for Returning the Day of the Week

Scalar Functions

Tables 55 through 60 list the available scalar functions.

- **Note:** In the tables for Conditional, Date, Math, String and Trend functions below, the variables *n*, *s*, *d*, and *exp* (and *val*) represent data items and columns (State, Amount Sold) or actual values ('NY', 6000) as arguments to scalar functions, and indicate number, string, date, or variable types, respectively.
- **Note:** The variable c indicates that only a data item reference may be used, and not a constant value. If constant values are substituted for data items, dates and text strings must be enclosed in single quotes. Examples in the tables that follow use a mixture of constants and data items, which are generally interchangeable.

ctions
ctions

Function	Description
Decode (c,exp,val,exp,valdef)	Compares value of item <i>c</i> to one or more expressions <i>exp</i> , and returns the value val matched to each expression, or a default <i>def</i> .
Nvl (exp1,exp2)	Returns <i>exp2</i> if null, and <i>exp1</i> otherwise.

Function	Description
AddMonths (d,n)	Adds <i>n</i> months to date <i>d</i> .
DayOfMonth (d)	Returns the day of month for date <i>d</i> .
LastDay (d)	Returns date of the last day of the month containing date <i>d</i> .
MonthsBetween (d1,d2)	Returns the number of months between dates $d1$ and $d2$ as a real number (fractional value).
NextDay (d,s)	Returns the date of the first weekday s after date d . If s is omitted, add one day to d .
Sysdate (c)	Returns the current system date and time for each record in item c.
ToCha (d/n, 'f' or "f")	Converts the date or number d/n into a string in the specified format. This function does not change the data, but rather the item data type. The results cannot be computed mathematically.
	If you are referencing a Date or Number column for the first argument (Date field), single quotes are not required. If you are passing a data value, single quotes are required.
	A comma must immediately follow the first argument. In the second argument (Format field), single quotes or double quotes must enclose values.
ToDate (s)	Returns date type in place of date-string s. This function does not change the data, but rather the item data type. The results can be computed mathematically.
ToMonth (d)	Returns a numeric month value for each value of <i>d</i> . You can change the value to display as a month string by adding and applying a <i>mmm</i> date format.

Table 56	Date Scalar	Functions
----------	-------------	-----------

Table 5	6 Date	Scalar	Functions	(Continued)
10010 0	o Dulo	ooului	i unociono	(continuou)

ToQtr (d)	Returns a string quarter value for each value of <i>d</i> .
ToYear (d)	Returns the integer year for each value of <i>d</i> . You can convert the year to display without commas by applying the 0 numeric format.

Table 57 Math Scalar Functions

_

Function	Description
Abs (n)	Returns the absolute value of number <i>n</i> .
Atn (n)	Returns arc tangent of number <i>n</i> radians.
Ceil (n)	Returns the smallest integer value greater than or equal to number <i>n</i> .
Cos (n)	Returns cosine of number <i>n</i> radians.
Cosh (n)	Returns hyperbolic cosine of number <i>n</i> radians.
Count (c)	Returns the number of row values in c (including nulls).
Exp (n)	Returns <i>e</i> (2.718) raised to exponential power <i>n</i> .
Max (a,b)	Returns the larger of items <i>a</i> and <i>b</i> for each new value.
Min (a,b)	Returns the smaller of items <i>a</i> and <i>b</i> for each new value.
Mod (n,m)	Returns the integer remainder of number <i>n</i> divided by number <i>m</i> . If <i>m</i> is larger, the default value is <i>n</i> .
Power (n,m)	Returns number n raised to exponential power <i>m</i> .
Round (n,m)	Returns number <i>n</i> rounded to m decimal places. The default value for <i>m</i> is 0.
Sign (n)	Returns indicator of -1, 0, or 1 if number <i>n</i> is variously negative, 0, or positive.
Sin (n)	Returns sine of number <i>n</i> radians.
Sinh (n)	Returns hyperbolic sine of number <i>n</i> radians.
Sqrt (n)	Returns square root of number <i>n</i> .
Tan (n)	Returns tangent of number <i>n</i> radians.
Tanh (n)	Returns hyperbolic tangent of number <i>n</i> radians.
Trunc (n,m)	Returns number n truncated to number m decimal places. The default value for m is 0.

Function	Description
Avg (numbers, break_col, break_value)	Returns the average (arithmetic mean) of values in a number column. The average includes null values when calculating the arithmetic mean.
AvgNonNull (numbers, break_col, break_value)	Returns the average (arithmetic mean) of values in a number column, excluding null values.
chr (n)	Returns string converted from ASCII numeric code n.
ColMax (numbers, break_col, break_value)	Returns the largest value in a column of numbers.
ColMin (numbers, break_col, break_value)	Returns the smallest value in a column of number.
Count (numbers, break_col, break_value)	Counts and returns the number of rows in a column.
CountDistinct (numbers, break_col, break_value)	Counts and returns the number of values in a column.
CountNonNull (numbers, break_col, break_value)	Counts the number of rows in a column.
CountNull (numbers, break_col, break_value)	Counts the number of rows in a column that contains null values.
Cume (numbers, break_col)	Returns a cumulative running total for each value in a column of numbers.
Next (c)	Returns the next row value of the referenced item c.
Prior (c)	Returns the prior row value of the referenced item c.
Sum (numbers, break_col, break_value)	Returns the total of a column of numbers.

Table 58 Numeric Scalar Functions

Table 59 Statistical Scalar Functions

Function	Description
Median (numbers, break)	Returns the median of a column of numbers.
Mode (numbers, break_col)	Returns the most frequently occurring value in a column of numbers.
Percentile (numbers, n, break_col)	Returns the <i>n</i> th percentile of values in a column of numbers in ascending order.
Rank (numbers, break_col)	Returns the rank of a number in a column of numbers. example There is a restriction on setting a filter on an aggregate column (which "Rank" is considered). To set a filter on the column, insert a new table into the document and drag the columns into the table Data Layout. This creates a "copy" of the columns which can have a filter set on it.
RankAsc (numbers, break_col)	Returns the rank of a number in a column of numbers in ascending order.

StdDev (numbers, break_col)	Estimates standard deviation based on a sample. The standard deviation is a measure of how widely values are dispersed from the average value (the mean). If your data represents the entire population, then compute the standard deviation using the function.
StdDevp (numbers, break_col)	Calculates standard deviation based on the entire population given as arguments. The standard deviation is a measure of how widely values are dispersed from the average value (the mean). If your data represents a sample of the population, then compute the standard deviation using the function.
Var (numbers, break_col)	Estimates variance based on a sample. The var function assumes that its arguments are a sample of the population. If your data represents the entire population, then compute the variance using the $varp$ function.
Varp (numbers, break_col)	Estimates variance based on the entire population. The varp function assumes that its arguments are the entire population. If your data represents a sample of the population, then compute the variance using the function.

 Table 59
 Statistical Scalar Functions (Continued)

Table 60 String Scalar Functions

Function	Description
Ascii (s)	Returns an ASCII numeric representation of string s.
Concat (s1, s2)	Returns text strings s1 and s2 concatenated.
Initcap (s)	Returns string s with the first letter of each word capitalized, and remaining characters in lower case.
Instr (s1,s2,n,m)	Returns position of m^{th} occurrence of string s2 in string s1, beginning at position number n . If n is negative, the count is made backwards from the end of s1. If no values are found, 0 is returned.
Length (s)	Returns character count of string s.
Lower (s)	Returns string s in lower case.
Ltrim (s1,s2)	Trims string s1 from the left, up to the first character not included in string s2.
Replace (s1,s2,s3)	Returns string item s1 with all occurrences of string s2 replaced by string s3. The default for s3 deletes each occurrence of s2.'
Rtrim (s1,s2)	Trims column string s1 from the right, up to the first character not included in string s2.
Substr (s,n,m)	Returns a portion of string s, m characters long, beginning at numeric position n . The default action for m includes all remaining characters.
Translate (s1,s2,s3)	Returns string s1, with each character contained in string s2 replaced by the corresponding characters in string s3.
Upper (s)	Returns string s in upper case.

Teradata Version 3 OLAP Functions

Interactive Reporting supports a number of Teradata version 3 OLAP and system functions, which dramatically reduce query time. See Tables 61 through 71 for complete descriptions of these functions.

Field	Description
Description	Accumulates a sum over an ordered set of rows, providing the current values of the SUM on each row
Syntax:	CSum(value_expression, sort_expression_list)
value_expression	A value_expression is a scalar numeric column expression for which a running sum is to be computed.
sort_expression_list	The sort_expression_list is a list of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).

Table 61	CSum	(Cumulative	Sum)	Function
	ooum	(oumunutivo	ounny	1 unocion

Table 62 MAvg (Moving Average) Function

Field	Description
Description	Computes the moving average of a column using the current row and the preceding width-1 rows.
Syntax:	MAvg(value_expression, width, sort_expression_list)
value_expression	The value expression represents a scalar numeric column expression for which a moving average is to be computed.
	The expression cannot contain any OLAP or aggregate functions.
width	The width represents the number of previous rows to be used in computing the moving average. The width value is always a positive integer constant. The maximum width is 4096.
sort_expression_list	The sort_expression_list is a list of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).

Field	Description
Description	Returns the moving difference between the current row-column value and the preceding n^{th} value.
	The moving difference is a common business metric used to compare activity for some variable in a current time period to the activity for the same variable in another time period at a fixed distance in the past.
Syntax:	MDiff(value_expression, width, sort_expression_list)
value_expression	The value expression represents a scalar numeric column expression for which a moving average is to be computed. The expression cannot contain any OLAP or aggregate functions.
width	The width represents the number of previous rows to be used in computing the moving average. The width value is always a positive integer constant. The maximum width is 4096.
sort_expression_list	The sort_expression_list is a list of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).

 Table 63
 MDiff (Moving Difference) Function

Table 64 MSum (Moving Sum) Function

Field	Description
Description	Computes the moving sum of a column using the current row and the preceding $n-1$ row.
Syntax:	MSum(value_expression, width, sort_expression_list)
value_expression	The value expression represents a scalar numeric column expression for which a moving average is to be computed. The expression cannot contain any OLAP or aggregate functions.
width	The width represents the number of previous rows to be used in computing the moving average. The width value is always a positive integer constant. The maximum width is 4096.
sort_expression_list	The sort_expression_list is a list of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).

Table 65 MLingreg (Multiple Linear Regression) Function

Field	Description
Description	Returns a predicted value for a column expression based on a least squares multiple linear regression of the previous width-1 based on the sort_expression column values. When there are fewer than width-1 preceding rows, the MLinreg function computes the regression using all the preceding rows.
Syntax:	MLingreg(value_expression, width, sort_expression_list)
value_expression	The value expression represents a scalar numeric column expression for which a moving average is to be computed. The expression cannot contain any OLAP or aggregate functions.
width	The width represents the number of previous rows to be used in computing the moving average. The width-1 previous rows are used to compute the linear regression and the row value itself is used to calculate the predicted value. The width value is always a positive integer constant greater than 1. The maximum width is 4096.
sort_expression_list	The sort_expression_list is a column reference used to sort the values and to define the dependent variable for calculating the linear regression. The sort_expression_list is an expression with optional sort direction specification. The default sort direction is ascending (ASC). Only one sort_expression is allowed with this function.

Table 66 Quantile Function

Field	Description
Description	Computes the quantile scores for the values in a column. A quantile is a generic interval of user-defined width.
Syntax:	Quantile(quantile_constant, sort_expression_list)
quantile_constant	A positive integer constant used to define the number of quantile partitions to be used.
sort_expression_list	List of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).
Quantile Value Range	0 through (Q-1) where Q is the number of quantile partitions specified by the quantile constant.

Table 67 Rank Function

Field	Description
Description	Returns the rank (1n) of all the rows in the group by the value of sort_expression_list, with the same sort_expression values receiving the same rank.

 Table 67
 Rank Function (Continued)

Syntax:	Rank(sort_expression_list)
sort_expression_list	List of expressions (with optional sort direction specifications) separated by commas. That is, it specifies the column references used to sort the values. The default sort direction is ascending (ASC).

Table 68 Current Timestamp Function

Field	Description
Description	Returns the current system timestamp and current session Time Zone displacement.
Syntax:	Current_Timestamp(fractional_precision)
fractional_precision	An option precision range for the returned timestamp value. The value range is 0 through 6, inclusive. The default is 6.
Properties	Data type: TIMESTAMP WITH TIME ZONE Length: 12 Not nullable
Fields	YEAR, MONTH, DAY, HOUR, MINUTE, SECOND, TIMEZONE_HOUR, TIMEZONE_MINUTE

Table 69 Quality Function	ı	Function	alify	69	able	Ta
---------------------------	---	----------	-------	----	------	----

Field	Description
Description	Filters results of a previously computed OLAP function according to user-specified conditions.
Syntax:	Qualify search_condition
Qualify	Represents a conditional clause in the SELECT statement.
search_condition	One or more conditional expressions that must be satisfied by the result rows. Aggregate operators with a Qualify clause can be used.
Usage Notes	When you specify a QUALIFY clause in a query, you must also specify a statistical function in one of the following locations within the query.
	select_list of the SELECT clause grouping_key of the GROUP BY clause search_condition of the QUALIFY clause
	When the WHERE, GROUP BY, and QUALIFY clauses are used together in a SELECT statement, the order of evaluation is:
	WHERE GROUP BY QUALIFY

Table 70 Sample Function

Field	Description
Description	Reduces the number of rows to be considered for further processing by returning mutually exclusive samples of rows specified either as a list of fractions of the total number of rows or as a list of numbers of rows from the SELECT query.
Syntax:	Sample(fractional_description [count_description])
fractional_description	Represents any number of floating point constants in the closed interval (0, 1) and separated by a comma. This is a list of fractions, the sum of which must not exceed 1.
count_description	Represents a positive integer constant list of row counts. A warning is returned if there are not enough rows in the result to satisfy the sampling request completely.
Usage Notes	No more than 16 samples can be requested per SELECT statement.
	SAMPLE operates on the evaluated output of the table expression, which can include a WHERE clause and GROUP BY, HAVING, or QUALIFY clauses, sampling the result according to use specification.
	A sampling request cannot be repeated. The identical sampling query run twice against the same data will report different rows in the result.
	Sampling can be used in a derived table, view, or INSERT-SELECT to reduce the number of rows to be considered for further computation.
	You cannot use SAMPLE in a subquery.
	If a <i>fraction_description</i> results in no rows being returned, a warning is generated.
	If a <i>count_description</i> cannot be completely satisfied, a warning is generated and the sample size is reduced to the number of remaining rows.

Table 71 SampleID Function

Field	Description
Description	Identifies the sample to which a row belongs, distinguishing rows belonging to different samples specified in the SAMPLE clause of a SELECT statement.
Syntax:	SAMPLEID
Definition	The sample ID identifies the sample to which a row belongs in the left-to-right order of the SAMPLE clause specification, from 1 through <i>n</i> (where <i>n</i> is the number of samples requested in the SAMPLE clause).
Where to Specify SAMPLEID	SAMPLEID can only be specified with a SAMPLE clause and can be displayed either as part of a SELECT clause or in an ORDER BY clause.

Functions for Returning the Day of the Week

If you need to return the day of the week on which a given date falls, some database systems enable this through functions that can be applied in the SELECT statement. Examples are the datepart() function in Microsoft and Sybase SQL Servers (which requests the 'weekday' part of the date), and Oracle's to_char() function, which specifies a format of D (for day of week number 1 through 7) or DAY to get the name of the day.

If you database does not support this function, you can add a computed item to the Results section to derive it.

Formatting Day of Week Data

If you simply need to format the day of the week for displaying or printing out the date, then no special computation or statement is needed.

- > To format the day of the week:
- 1 In the Results section, select the date item and choose Format > Number.

The Number page of the Properties dialog box is displayed.

- 2 Select **Custom** from the Category list.
- 3 In the Format field, type ddd to display a three-letter day abbreviation or dddd to display the full name, and click OK.

Analyzing Data Based on Day of Week Data

If you add a format such as dddd mm/dd/yyyy, it displays a date as T*uesday* 05/19/2000. This may not be sufficient if you need to perform analysis on data based on the day of the week. Despite the display format, the data in the field is still a date. That is, if you need to compare sales for Mondays versus other days of the week over a given time period, changing the display format does not address your original question. In this case, you need to group all Mondays to do that sort of analysis.

- > To analyze data based on the day of the week:
- 1 In the Results section, select the item and choose **Results > Add Computed Item**.
- 2 In the Name field, assign a new name to the column.
- 3 In the Definition field, type: to_char(<MyDate>,'dddd')

Replace *MyDate>* with the name of the column for which you need the day of the week information. This creates a string from the date column with the desired format, as discussed earlier. You can also add the following:

```
: decode((NEXT_DAY ( <MyDate>, 'Sunday' )) -<MyDate>, 7,'Sunday', 6,'Monday', 5,'Tuesday', 4,'Wednesday', 3,'Thursday', 2,'Friday', 1,'Saturday', 'Error!')
```

Common Computed Item Examples

The examples that follow show you how to apply some of the mathematical, numerical, and statistical calculations available in the Pivot and Chart sections using computed items.

- Math Functions
- Central Tendency
- Calculated Averages
- Percentile
- Rank

Math Functions

A mathematical equation consists of the argument and a simple or complex arithmetic operator that is applied to the argument. In this example, the sales tax is calculated as a percentage of the revenue (3.5%). The "Net Amount" column is calculated by subtracting the tax dollars from the revenue.



Central Tendency

If you need to distinguish patterns within a given set of data, you can begin looking for the center of distribution where statistics tend to reside. This form of measurement involves finding the "average" in the data set and is typically referred to as the *Measure of Central Tendency* technique.

Three types of measurements are associated with this technique:

- Mean The value is equal to the sum of the measures divided by the number of measures.
- Median The value is representative of the positional middle measure.
- Mode The value that occurs with the greatest frequency.

In this example, Unit Sales represent the total number of product units purchased. *Mean* of Unit Sales represents the average purchase size. *Median* of Unit Sales represent the number of product units that scores exactly in the middle of all purchase quantities. The *Mode* of Unit Sales presents the number of product units most commonly purchased at one time.

	Unit Sales	Mean Value	Median Value	Mode Value	
1999	5272112	411	276	370	
2000	884533	621	404	441	
=Avg(Unit_	_Sales,Year)	=Media	an(Unit_Sales,Ye	ear)	=Mode(Unit_Sales,

Calculated Averages

Calculated averages can be applied to break columns and break values. In this example, a variety of columns are created and display different average calculations based on the Amount_Sales column.

					1999		
		Amount Sales	Entire Year	By Quarter	By Quarter and Product Line	For Q1	For Books
Q1	Books	\$13,744,748.12	\$8,908.96	\$7,392.77	\$8,816.39	7,392.769	10,745.692
	Music	\$4,451,016.53	\$8,908.96	\$7,392.77	\$5,098.53	7,392.769	10,745.692
	Videos	\$2,533,558.55	\$8,908.96	\$7,392.77	\$6,810.64	7,392.769	10,745.692
Q2	Books	\$13,774,642.22	\$8,908.96	\$6,915.81	\$8,383.84	7,392.769	10,745.692
	Music	\$4,433,484.45	\$8,908.96	\$6,915.81	\$4,721.50	7,392.769	10,745.692
	Videos	\$2,608,450.82	\$8,908.96	\$6,915.81	\$6,094.51	7,392.769	10,745.692
Q3	Books	\$16,747,678.86	\$8,908.96	\$7,743.84	\$9,092.12	7,392.769	10,745.692
	Music	\$5,657,958.46	\$8,908.96	\$7,743.84	\$5,601.94	7,392.769	10,745.692
	Videos	\$3,141,276.30	\$8,908.96	\$7,743.84	\$7,027.46	7,392.769	10,745.692
Q4	Books	\$31,769,448.07	\$8,908.96	\$12,704.96	\$15,634.57	7,392.769	10,745.692
	Music	\$9,326,459.77	\$8,908.96	\$12,704.96	\$8,060.90	7,392.769	10,745.692
	Videos	\$6,077,618.05	\$8,908.96	\$12,704.96	\$11,598.51	7,392.769	10,745.692

The following table shows how averages have been calculated in the figure above.

Column	Numeric Function
Amount Sales	sum of Amount Sales by quarter and Product Line
	= Sum (Amount_Sales)
Entire Year	average purchase amount (Amount Sales) across all quarters and Product Lines
	= Avg (Amount_Sales)
By Quarter	average purchase amount (Amount Sales) in a specific quarter
	= Avg (Amount_Sales, Quarter)
For Quarter and	average purchase amount (Amount Sales) in a specific quarter for a Product line
Product Line	= Avg data function applied to Amount_Sales column
For Q1	average purchase size (Amount Sales) across all Product Lines for Q1 specifically
	= Avg (Amount_Sales, Quarter, 'Q1')
For Books	average purchase size (Amount Sales) across all Quarters for Books
	= Avg (Amount_Sales, Product_Line, 'Books')

 Table 72
 Calculated Averages

Percentile

Suppose Sales Managers qualify for a special bonus if they are within the 80th percentile (Qualify column). You can define an 80th percentile value for Amount Sales.

	Amount Sales	Eightieth Percentile	Qualify	
Redentor	365,131.171	7,258,107.987		The Eightieth Percentile column calculates the 80th
Robinson	1,982,942	7,258,107.987		percentile value for all Sales managers:
Saint	1,302,044.456	7,258,107.987		=Percentile (Amount Sales, .8)
Schmidt	4,067,065.456	7,258,107.987		
Schultz	808,609.729	7,258,107.987		Important! Surface values must be used in this type of export.
Shay	3,159,782.897	7,258,107.987		
Shibasaki	9,688,780.173	7,258,107.987	~	
Stuber	691,960.051	7,258,107.987		The Qualify column determines whether a Sales Manager
Vilhena	731,976.471	7,258,107.987		is within the 80th percentile:
Wah	7,266,260.345	7,258,107.987	-	
Wayne	32,713,875.41	7,258,107.987	~	{`a'}
White	7,245,879.451	7,258,107.987		
Wilson	6,353,954.035	7,258,107.987		
	,			

The second example identifies countries that make sales transactions under \$10,000.

	05% of Color
Amount Sales	Amounts under the value
\$692,299.72	\$12,879.02
\$14,598,717.21	\$33,025.46
\$1,097,107.64	\$5,805.23
\$5,711,431.51	\$19,578.64
\$5,976,121.79	\$11,650.27
\$5,978,108.14	\$10,926.27
\$3,159,782.90	\$16,449.54
\$14,957,294.27	\$22,995.99
\$1,627,858.74	\$25,594.52
\$2,774,068.00	\$8,372.94
\$15,053,892.94	\$53,040.96
AC1 051 005 05	650 400 00
	Amount Sales \$692,299.72 \$14,598,717.21 \$1,097,107.64 \$5,976,121.79 \$5,976,108.14 \$3,159,782,90 \$1,627,858.74 \$1,627,858.74 \$2,774,068.00 \$15,053,892.94

Rank

You can return the rank of a number in a column of numbers. The Rank function works as if you were to sort the list in descending order. In this example, Amount Sales values are ranked for each Country.

Note: The Rank function assigns duplicate numbers the same rank, which affects the ranks of subsequent numbers.

		Amount Sales	Rank by Country
Books	Argentina	347,544.361	12
	Australia	9,412,848.248	4
	Brazil	568,645.399	11
	Canada	3,667,768.272	7
	France	3,359,648.059	5
	Germany	3,568,789.42	6
	Ireland	2,061,186.883	8
	Japan	9,344,287.124	2
	Norway	1,178,789.796	10
	Sweden	1,594,555.432	9
	United Kingo	10,234,860.015	3
	USA	43,622,679.262	1-
lusic	Argentina	241,577.909	12
	Australia	2,976,666.105	4
	Brazil	363,025.444	11
	Canada	1,266,314.453	7
	France	1,760,651.286	5
	Germany	1,508,942.386	6
	Ireland	706,185.127	8
	Japan	3,949,849.182	2
	Norway	238,788.625	10
	Sweden	789,847.619	9
	United Kingo	3,017,461.916	3
	USA	10,650,603.589	1

Countries ranked by their Amount sales. — Country is the break column. =Rank (Amount Sales, Country)

Project Sales

This example shows a calculation for a 20% increase in sales projections for each quarter, based on Amount Sales for 1999.



Scalar Function Examples

This section provides examples of some common scalar functions. Each example shows the syntax used and the result of applying the function. The examples that follow do not include all the Interactive Reporting Studio scalar functions. The functions described include:

- Avg
- AvgNonNull
- ColMax
- ColMin

- Count
- CountDistinct
- CountNull
- CountNonNull
- Cume
- Sum
- Median
- Mode
- Percentile
- Rank
- RankAsc
- StdDev
- StdDevp
- Var
- Varp

Avg

The Avg function returns the average (arithmetic mean) of values in a number column.

Avg (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the average is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the average of numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the Avg function is used on the numeric column and break_column. The results are shown in the Computed column.

Avg (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	112
СА	Burbank	240	168.5
СА	Glendale	97	168.5
FL	Palmetto	70	70
MD	Laurel	97	97
MI	Detroit	208	208
MN	Eagan	159	159
NY	New York	490	335
NY	Rochester	180	335

Table 73Avg Example 2

AvgNonNull

The AvgNonNull function returns the average (arithmetic mean) of values in a number column, excluding null values.

Avg (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the average is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the average of non-null numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the AvgNonNull function is used on the numeric column, break_column, and break_value. The results are shown in the Computed column

```
AvgNonNull (Amount, State, 'CA')
```

State	City	Amount	Computed
AZ	Tucson	112	240
СА	Burbank	240	240
СА	Glendale	NULL	240
FL	Palmetto	70	240

Table 74 AvgNonNull Example 3

State	City	Amount	Computed
MD	Laurel	97	240
MI	Detroit	208	240
MN	Eagan	159	240
NY	New York	490	240
NY	Rochester	180	240

ColMax

The ColMax function returns the largest value in a column of numbers.

```
ColMax (numbers, break_col, break_value)
```

where:

numbers references the column that contains the number on which the maximum column value is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the maximum value of numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the ColMax function is used on the numeric column. The results are shown in the Computed column.

ColMax(Amount)

State	City	Amount	Computed
AZ	Tucson	112	490
CA	Burbank	240	490
CA	Glendale	NULL	490
FL	Palmetto	70	490
MD	Laurel	97	490
MI	Detroit	208	490
MN	Eagan	159	490

Table 75	ColMax	Example
----------	--------	---------

State	City	Amount	Computed
NY	New York	490	490
NY	Rochester	180	490

ColMin

The ColMin function returns the smallest value in a column of numbers.

ColMin (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the count of minimum column value is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the minimum value of numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the ColMin function is used on the numeric column and break_column. The results are shown in the Computed column.

ColMin (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	112
СА	Burbank	240	240
СА	Glendale	Null	240
FL	Palmetto	70	70
MD	Laurel	97	97
MI	Detroit	208	208
MN	Eagan	159	159
NY	New York	490	180
NY	Rochester	180	180

Table 76ColMin Example 2

Count

The count function counts the number of rows in a column.

```
Count (numbers, break_col, break_value)
```

where:

numbers references the column that contains the numbers on which the count is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the count of numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the count function is used on the numeric column. The results are shown in the Computed column.

Count (Amount)

State	City	Amount	Computed
AZ	Tucson	112	9
СА	Burbank	240	9
СА	Glendale	97	9
FL	Palmetto	70	9
MD	Laurel	97	9
MI	Detroit	208	9
MN	Eagan	159	9
NY	New York	490	9
NY	Rochester	180	9

Table 77 Count Example

CountDistinct

The CountDistinct function counts the number of values in a column.

CountDistinct (numbers, break_col, break_values)
where:

numbers references the column that contains the numbers on which the count of distinct (unique) values is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns a distinct (unique) count of numbers column where value in break_col equals break_value.

Note: The CountDistinct function differentiates actual values and not the rows. The CountDistinct function counts only the actual rows in a column. For example, if column named "OS Operating Systems" has one hundred rows and shows data by Windows and UNIX systems, the CountDistinct function counts only the three operating systems and not the number of rows.

In this example, the **CountDistinct** function is used on the numeric column and break_column. The results are shown in the Computed column.

CountDistinct (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	1
СА	Burbank	240	1
СА	Glendale	240	1
FL	Palmetto	70	1
MD	Laurel	97	1
MI	Detroit	208	1
MN	Eagan	159	1
NY	New York	490	1
NY	Rochester	180	1

 Table 78
 CountDistinct Example 2

CountNull

The CountNull function counts the number of rows in a column that contains null values.

CountNull (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the count of null values is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the count of null numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the CountNull function is used on the numeric column. The results are shown in the Computed column.

CountNull(Amount)

State	City	Amount	Computed
AZ	Tucson	112	1
СА	Burbank	240	1
СА	Glendale	NULL	1
FL	Palmetto	70	1
MD	Laurel	97	1
MI	Detroit	208	1
MN	Eagan	159	1
NY	New York	490	1
NY	Rochester	180	1

Table 79 CountNull Example

CountNonNull

The CountNonNull function counts the number of rows in a column that do not contain null values.

CountNonNull (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the count of non-null values is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the count of non-null numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the CountNonNull function is used on the numeric column, break_column, and break_value. The results are shown in the Computed column.

CountNonNull (Amount, State, 'CA')

State	City	Amount	Computed
AZ	Tucson	112	1
СА	Burbank	240	1
СА	Glendale	97	1

Table 80 CountNonNull Exa	ample 3
---------------------------	---------

State	City	Amount	Computed
FL	Palmetto	70	1
MD	Laurel	97	1
MI	Detroit	208	1
MN	Eagan	159	1
NY	New York	490	1
NY	Rochester	180	1

Cume

The cume function returns a cumulative running total for each value in a column of numbers.

```
Cume (numbers, break_col)
```

where:

numbers references the column that contains the numbers on which the cume is calculated.

break_col is an optional parameter that references a break column.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the Cume function is used on the numeric column and break_column. The results are shown in the Computed column.

Cume (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	112
СА	Burbank	240	240
СА	Glendale	97	337
FL	Palmetto	70	70
MD	Laurel	97	97
MI	Detroit	208	208
MN	Eagan	159	159
NY	New York	490	490
NY	Rochester	180	670

Table 81	Cume Example 2
----------	----------------

Sum

The sum function computes the total for a column of numbers.

Sum (numbers, break_col, break_value)

where:

numbers references the column that contains the numbers on which the sum is calculated.

break_col is an optional parameter that references a break column.

break_value is an optional parameter that returns the sum of numbers column where value in break_col equals break_value.

Note: If constant values in the break_value column are substituted for data items, dates and text strings must be enclosed in single quotes.

In this example, the Sum function is used on the numeric column. The results are shown in the Computed column.

Sum(Amount)

State	City	Amount	Computed
AZ	Tucson	112	1556
СА	Burbank	240	1556
СА	Glendale	NULL	1556
FL	Palmetto	70	1556
MD	Laurel	97	1556
MI	Detroit	208	1556
MN	Eagan	159	1556
NY	New York	490	1556
NY	Rochester	180	1556

Table 82 Sum Example

Median

The Median function returns the median of a column of numbers. The median is the middle value or number in the middle of a set of numbers (and not the average).

Median (numbers, break_col)

numbers references the column that contains the numbers on which the median is calculated. *break_col* is an optional parameter that references a break column. In this example, the Median function is used on a numeric column that has an odd number of rows:

Median (Amount)

The Median function returns the number in the middle, which in this example is 30.

State	Amount	Computed
AZ	10	30
СА	20	30
FL	30	30
MD	40	30
MI	50	30

Table 83Median Example 1

If the numbers column has an even number of rows, the Median function calculates the average of the two numbers in the middle.

State	Amount	Computed
AZ	10	25
СА	20	25
FL	30	25
MD	40	25

Table 84 Median Example 2

Mode

The Mode function returns the most frequently occurring value in columns of numbers.

Mode (numbers, break_col)

where:

numbers references the column that contains the numbers on which the mode is calculated.

break_col is an optional parameter that references a break column.

Note: Null values in the numbers column are ignored. Zeroes (0) are included. If the numbers column has no duplicate data values, the Mode function returns the value of the first cell in the numbers column.

In this example, the Mode function is used on the numeric column. The results are shown in the Computed column.

Mode(Amount)

Table 8	5 Mo	de Exa	ample
---------	------	--------	-------

State	Amount	Computed
AZ	10	10
СА	10	10
FL	0	10
MD	40	10
MI	50	10

Percentile

The Percentile function returns the n^{th} percentile of values in a column of numbers.

```
Percentile (numbers, n, break_col)
```

where:

numbers references the column that contains the numbers on which the percentile is calculated.

n is the percentile value 0 to 1 inclusive.

break_col is an optional parameter that references a break column.

Note: Percentile can also be used to return quartile values by setting the *n*th percentile to the following: 0.25 for first quartile, 0.5 for second quartile, 0.75 for third quartile.

In this example, two computed value columns have been calculated. In the first computed column, the Percentile function is used on the numeric column and the n^{th} percentile of values (0 in this case).

Percentile (Units, 0)

In the second computed column, the Percentile function is used on the numeric column, and the n^{th} percentile of values (.25 in this case).

Percentile (Units,.25)

State	Units	Computed 1	Computed 2
WI	50	50	96
AZ	70	50	96
CA	96	50	96
СА	98	50	96

Table	86	Percentile	Example
-------	----	------------	---------

State	Units	Computed 1	Computed 2
CA	101	50	96
FL	112	50	96
MD	159	50	96
NY	241	50	96
NY	490	50	96

Rank

The Rank function returns the rank of a number in a column of numbers. It works as if you were to sort the list in descending order.

Rank(numbers, break_col)

where:

numbers references the column that contains the numbers on which the rank is calculated.

break_col is an optional parameter that references a break column.

Note: The Rank function assigns duplicate numbers the same rank. The presence of duplicate numbers affects the ranks of subsequent numbers.

In this example, the Rank function is used for the numeric column (the column that contains the numbers to rank). The results are shown in the Computed column.

Rank	(Amount)

State	City	Amount	Computed
AZ	Tucson	112	6
CA	Burbank	241	2
CA	Glendale	97	7
FL	Palmetto	70	9
MD	Laurel	97	7
MI	Detroit	208	3
MN	Eagan	159	5
NY	New York	490	1
NY	Rochester	180	4

Table	87	Rank	Example
14010	•••		Dumpio

RankAsc

The RankAsc function returns the rank of a number in a column of numbers. It works as if you were to sort the list in ascending order. In this case, the rank of the number would be its position.

RankAsc (numbers, break_col)

where:

numbers references the column that contains the numbers on which the rank is calculated.

break_col is an optional parameter that references a break column.

Note: The RankAsc function assigns duplicate numbers the same rank. The presence of duplicate numbers affects the ranks of subsequent numbers.

In this example, the RankAsc function is used on the numeric column and break_column. The results are shown in the Computed column. RankAsc (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	3
СА	Burbank	240	6
CA	Glendale	97	6
FL	Palmetto	70	1
MD	Laurel	97	2
MI	Detroit	208	5
MN	Eagan	159	4
NY	New York	490	7
NY	Rochester	180	7

Table 88RankAsc Example 2

Note: The RankAsc is calculated on the SUM(col_1) and grouped by col_2.

StdDev

The StdDev function returns a standard deviation based on a sample. It is a measure of how widely values are dispersed from the average value (the mean).

StdDev (numbers, break_col)

where:

numbers references the column that contains the numbers on which the standard deviation is calculated.

break_col is an optional parameter that references a break column.

- **Note:** stdDev assumes that its arguments are a sample of the population. If you data represents the entire population, then compute the standard deviation using StdDevp.
- **Note:** The standard deviation is calculated using the *nonbiased* or *n*-1 method.
- Note: If a result set contains one row of data or less, the StdDev function should return an error.

In this example, the StdDev function is used on the numeric column. The results are shown in the Computed column.

StdDev (Amount)

State	City	Amount	Computed
AZ	Tucson	112	128.11
СА	Burbank	240	128.11
СА	Glendale	97	128.11
FL	Palmetto	70	128.11
MD	Laurel	97	128.11
MI	Detroit	208	128.11
MN	Eagan	159	128.11
NY	New York	490	128.11
NY	Rochester	180	128.11

 Table 89
 StdDev Example

StdDevp

The stdDevp function calculates and returns a standard deviation based on the entire population given as arguments. The standard deviation is a measure of how widely values are dispersed from the average value (the mean).

StdDevp (numbers, break_col)

where:

numbers references the column that contains the numbers on which the standard deviation is calculated.

break_col is an optional parameter that references a break column.

- **Note:** stdDevp assumes that its arguments are the entire population. If your data represents a sample of the population, then compute the standard deviation using StdDev.
- **Note:** The standard deviation is calculated using the *biased* or *n* method.

In this example, the *stdDevp* function is used for the numeric column. The results are shown in the Computed column.

StdDevp (Amount)

State	City	Amount	Computed
AZ	Tucson	112	120.79
СА	Burbank	240	120.79
СА	Glendale	97	120.79
FL	Palmetto	70	120.79
MD	Laurel	97	120.79
MI	Detroit	208	120.79
MN	Eagan	159	120.79
NY	New York	490	120.79
NY	Rochester	180	120.79

Table 90 StdDevp Example

Var

The var function estimates variance based on a sample.

Var (numbers, break_col)

where:

numbers references the column that contains the numbers on which the variance is calculated.

break_col is an optional parameter that references a break column.

Note: Var assumes that its arguments are a sample of the population. If your data represents the entire population, then compute the variance usingVarp.

In this example, the Var function is used on the numeric column and break_column. The results are shown in the Computed column.

Var (Amount, State)

State	City	Amount	Computed
AZ	Tucson	112	0
СА	Burbank	240	10,224.50
СА	Glendale	97	10,224.50
FL	Palmetto	70	0
MD	Laurel	97	0
MI	Detroit	208	0
MN	Eagan	159	0
NY	New York	490	48,050.00
NY	Rochester	180	48,050.00

Table 91 Var Example 2

Varp

The varp function estimates variance based on the entire population.

```
Varp (numbers, break_col)
```

where:

numbers references the column that contains the numbers on which the variance is calculated.

break_col is an optional parameter that references a break column.

Note: Varp assumes that its arguments are the entire population. If your data represents a sample of the population, then compute the variance using Var.

In this example, the varp function is used on the numeric column. The results are shown in the Computed column.

Varp (Amount)

State	City	Amount	Computed
AZ	Tucson	112	14,589.56
СА	Burbank	240	14,589.56
СА	Glendale	97	14,589.56
FL	Palmetto	70	14,589.56
MD	Laurel	97	14,589.56
MI	Detroit	208	14,589.56

State	City	Amount	Computed
MN	Eagan	159	14,589.56
NY	New York	490	14,589.56
NY	Rochester	180	14,589.56

Trend Functions

Both the Pivot and Results sections offer trend functions, which allow you to perform statistical analysis within a sample data set of variable size. Trend functions are useful for removing data irregularities/fluctuations, analyzing data trends, and smoothing a set of data points. They are often used to reduce noise that exists when visualizing large data sets by providing aggregation capabilities at configurable window sizes.

Review the following sections for information on:

- General Moving Average and Moving Function Functionality
- Simple Moving Averages
- Positioning of Moving Average Results Trailing and Centered Averages
- Weighted Moving Averages
- Exponential Moving Averages
- Moving Diff
- Moving Maximum
- Moving Median
- Moving Sum
- Moving Minimum
- Direction Of Moving Function Calculation

General Moving Average and Moving Function Functionality

The term "Moving" when used with functions such as Moving Averages usually refers to a fixed window or "sub-range" of analysis that moves over a larger range of numeric data values. For each window, a calculation such as an average (mean) is performed. As the window moves over the larger range, one number value is dropped from the window and a new value is added.

To illustrate the effect of Moving Averages as opposed to a simple average (mean) consider the following list of arbitrary "Sales" values over a period of twelve months as shown in a Table or Results Section:

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

The simple mean value of all of the above twelve Sales values is:

```
Sum of all twelve values divided by 12 = 296 / 12 = 24.67
```

While having some virtue, this simple mean value does not offer any insight into analyzing the variation trend of the original values or to provide a more accurate mean value at varying points of the range. Moving Averages, on the other hand, do offer the ability to understand the trend of data by calculating many average (mean) values over the entire range of original data values.

There are several variations to Moving Average calculation, the most common of these are: Simple Moving Averages, Weighted Moving Averages and Exponential Moving Averages. All of these variations are described in detail in the following sections. Additionally, other "Moving Functions": Moving Maximum, Moving Minimum, Moving Median, Moving Sum and Moving Difference whose functionality closely resembles that of Moving Averages are discussed.

Simple Moving Averages

In a Simple Moving Average calculation, the original range of data values is split into smaller "windows" and a simple mean value of each window is calculated. An example of the calculation involved for such Simple Moving Averages is shown below.

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

If the first three Sales values were summed and then this sum was divided by 3 to give a mean value, this value would be:

(10 + 15 + 17) / 3 = 42 / 3 = 14

Next, consider the mean of the sum of the second, third and fourth original values i.e.:

(15 + 17 + 20) / 3 = 52 / 3 = 17.33

This pattern could be repeated to reveal the mean values of the third, fourth and fifth values; fifth, sixth and seventh values and so on until all subsequent windows of three numbers are summed and their mean values obtained.

The Simple Moving Averages of the original range of values for a window of 3 (i.e. in this case, a 3-Month Simple Moving Average) could be evaluated to be:

Month	Original Sales Values	3-Month Simple Moving Average
Jan	10	
Feb	15	
Mar	17	14
Apr	20	17.33
Мау	22	19.66
Jun	20	20.66
Jul	25	22.33
Aug	27	24
Sep	30	27.33
Oct	35	30.66
Nov	37	34
Dec	40	37.33

 Table 93
 Simple Moving Average

Positioning of Moving Average Results - Trailing and Centered Averages

Note that in Simple Moving Average table, the average of numbers n, n+1 and n+2 in the "Original Values" column (where "n" refers to the row position) is placed in row position n+2 of the "3-Month Simple Moving Average" column. This Moving Average display technique is known as "Trailing Averages". An alternative display technique is known as "Centered Averages" which instead positions the Moving Average in the center row of the window. The table following illustrates the difference in these display techniques using the first three values from above:
Table 94 Centered and Trailing Averages

Month	Centered Averages	Trailing Averages
Jan		
Feb	14	
Mar		14

"Centered Averages" display requires further calculations when the window is an even number and it is not available for Simple Moving Averages and other Moving Functions at this time.

All "Moving Functions" in this particular implementation will display data according to the "Trailing Averages" principle.

Note also that from the above two tables, "Trailing Averages" display causes the initial n-1 (where n = window size) rows of result data to have no value (rows 1 and 2 are blank in the above examples). This is the generally accepted standard for the initial "n-1" terms and is the standard adopted for the implementation of most Moving Functions.

Table 95 illustrates the above monthly Sales data Simple Moving Average calculation using "Trailing Averages" display:

The Simple Moving Averages of the original range of values for a window of 3 (i.e. in this case, a 3-Month Simple Moving Average) could be evaluated to be:

Month	Original Sales Values	3-Month Simple Moving Average
Jan	10	
Feb	15	
Mar	17	14
Apr	20	17.33
Мау	22	19.66
Jun	20	20.66
Jul	25	22.33
Aug	27	24
Sep	30	27.33
Oct	35	30.66
Nov	37	34
Dec	40	37.33

Table 953-Month Simple Moving Average

Weighted Moving Averages

With Simple Moving Averages, each data value in the "window" in which the calculation is performed is given an equal significance or weight. It is often the case, especially in financial price data analysis, that more chronologically recent data should carry a greater weight. In these cases, Weighted Moving Average (or Exponential Moving Average - see the following topic) functionality is often preferred.

Consider the same table of Sales data values for twelve months:

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

- > To calculate a Weighted Moving Average:
- **1** Calculate how many intervals of data are participating in the Moving Average calculation (i.e. the size of the calculation "window").

If the calculation window is said to be n, then the most recent data value in the window is multiplied by n, the next most recent multiplied by n-1, the value prior to that multiplied by n-2 and so on for all values in the Window.

- 2 Divide the sum of all of the multiplied values by the sum of the weights to give the Weighted Moving Average over that window.
- 3 Place the Weighted Moving Average value in a new column according to the trailing averages positioning described above.

To illustrate these steps, consider if a 3-month Weighted Moving Average of Sales in December is required (using the above table of Sales values).

The term "3-month" implies that the calculation "window" is 3, therefore the Weighted Moving Average calculation algorithm for this case should be:

```
{(Dec Sales value * 3) + (Nov Sales value * 2) + (Oct Sales value * 1)}
divided by (3 + 2 + 1)
= {(40 * 3) + (37 * 2) + (35 * 1)} / 6
= (120 + 74 + 35) / 6
= 229 / 6
= 38.17
```

Or, if a 3-month Weighted Moving Average were evaluated over the entire original range of data, the results would be:

Month	Orignal Sales Values	3-Month Simple Moving Average
Jan	10	
Feb	15	
Mar	17	15.12
Apr	20	18.17
Мау	22	20.5
Jun	20	20.67
Jul	25	22.83
Aug	27	25.17
Sep	30	28.17
Oct	35	32
Nov	37	35.17
Dec	40	38.17

Table 96 3-month Weighted Moving Average

Exponential Moving Averages

Exponential Moving Averages, similar to Weighted Moving Averages, also assign a greater weight to more recent data values. Unlike Weighted Moving Averages, however, they use the previously calculated Exponential Moving Average value as a basis for calculation rather than the original (non-Averaged) data values. In this way, the calculation method used by Exponential Moving Averages is cumulative, meaning that (unlike Simple Moving Averages or Weighted Moving Averages) all previous data values have some effect on the Exponential Moving Average to be calculated, although this effect diminishes greatly with time.

Exponential Moving Averages tend to be more accurate than the other types of Moving Average when the original data values show a more rapid degree of variation over time (or other variable).

The formula for calculating an Exponential Moving Average (EMA) is:

X = (K * (C - P)) + P

Where:

X = Current EMA (i.e. EMA to be calculated)

- C = Current original data value
- K = Smoothing Constant
- P = Previous EMA

(The first EMA in the range to be calculated is arbitrary and can be the corresponding original data value or, often, a Simple Moving Average value.

Where:

K =Smoothing Constant = 2 / (1 + n)

n = number of periods for EMA i.e. the Window to calculate.

This rather complex calculation is, perhaps, best illustrated by example.

Consider the table of monthly Sales values as shown previously:

	Month	Sales
1	Jan	10
2	Feb	15
3	Mar	17
4	Apr	20
5	May	22
6	Jun	20
7	Jul	25
8	Aug	27
9	Sep	30
10	Oct	35
11	Nov	37
12	Dec	40

If we calculated the Exponential Moving Average in a similar fashion to the 3-Month Simple Moving Average, we would perform the following steps:

4 Calculate the Smoothing Constant according to the 2 / (1 + n) formula.

N = window of values = 3, therefore the Smoothing Constant is:

2 / (1 + 3) = 0.5

- 5 For the first Exponential Moving Average, use the first original data value (in this case, that for the Month of "Jan").
- 6 For subsequent values, follow the calculation according to the above formula.

X = (K * (C - P)) + P as indicated below:

Table 97 X = (K * (C - P)) + P Example

Month	Sales	Calculation (K * (C - P)) + P	Exponential Moving Average
Jan	10	= Original value	10
Feb	15	(0.5 * (15 - 10)) + 10	12.5
Mar	17	(0.5 * (17 - 12.5)) + 12.5	14.75
Apr	20	(0.5 * (20 - 14.75)) + 14.75	17.375
May	22	(0.5 * (22 - 17.375)) + 17.375	19.688
Jun	20	(0.5 * (20 - 19.6875)) + 19.6875	19.844

Month	Sales	Calculation (K * (C - P)) + P	Exponential Moving Average
Jul	25	(0.5 * (25 - 19.844)) + 19.844	22.422
Aug	27	(0.5 * (27 - 22.422)) + 22.422	24.711
Sep	30	(0.5 * (30 - 24.711)) + 24.711	27.355
Oct	35	(0.5 * (35 - 27.355)) + 27.355	31.178
Nov	37	(0.5 * (37 - 31.178)) + 31.178	34.089
Dec	40	(0.5 * (40 - 34.089)) + 34.089	37.044

Moving Diff

The MovingDiff function operates over a moving window of values, and returns the difference between the current value and the value at the beginning of the window. This function is similar to the MovingMax function. No "Weighted" or "Exponential" can be calculated for this function.

The MovingDiff function takes the following arguments:

MovingDiff (column, window, break_col)

where:

- column (required) Specify the column that contains the numeric fact on which the Moving Function column value is calculated. If you are applying the function from the Pivot section, you can only use a column that has already been added to the Facts pane of the Data Layout.
- window (optional) Specify a moving "window" of values from the *Column* on which the Moving Function gets calculated The window value must be a positive integer of value equal to or less than the total number of rows in the *Column* (within any *Break Column* value).

If the window value is an interger of greater value than the number of values in the *Column*, then the window value defaults to the number of rows in the *Column* (within any *Break Column* value). If no window value is specified, then the window value defaults to 3.

• break column (optional) – Specify the name of the column on which to perform the Moving Function calculation. The break column refers to the rows of similar valued data cell within the Pivot or Results section.

The following results would occur if you used the Moving Diff function with a Window of 3 (MovingDiff(Sales,3).

Month	Orignal Sales Values	3-Month MovingDiff Average
Jan	10	
Feb	15	
Mar	17	7
Apr	20	5
Мау	22	5
Jun	20	0
Jul	25	3
Aug	27	7
Sep	30	5
Oct	35	5
Nov	37	7
Dec	40	5

Table 98 Moving Difference Example

Moving Maximum

The MovingMax (Moving Maximum) function operates over a moving window of values. For each Window, the MovingMax returns the maximum value found in the Window. This function is similar to the Simple Moving Average, No "Weighted" or "Exponential" can be calculated for this function.

The MovingMax function takes the following arguments:

MovingMax (column, window, break_col)

where:

- column (required) Specify the column that contains the numeric fact on which the Moving Function column value is calculated. If you are applying the function from the Pivot section, you can only use a column that has already been added to the Facts pane of the Data Layout.
- window (optional) Specify a moving "window" of values from the *Column* on which the Moving Function gets calculated The window value must be a positive integer of value equal to or less than the total number of rows in the *Column* (within any *Break Column* value).

If the window value is an interger of greater value than the number of values in the *Column*, then the window value defaults to the number of rows in the *Column* (within any *Break Column* value). If no window value is specified, then the window value defaults to 3.

• break column (optional) – Specify the name of the column on which to perform the Moving Function calculation. The break column refers to the rows of similar valued data cell within the Pivot or Results section.

The following results would occur if you used the MovingMax function with a Window of 3 (MovingMax(Sales,3).

Month	Orignal Sales Values	3-Month MovingMax
Jan	10	
Feb	15	
Mar	17	17
Apr	20	20
Мау	22	22
Jun	20	22
Jul	25	25
Aug	27	27
Sep	30	30
Oct	35	35
Nov	37	37
Dec	40	40

Table 99 MovingMax Function Example

Moving Median

The MovingMed function operates over a moving window of values. For each Window, the MovingMed returns the middle value (in terms of rank) found in the Window. If the Window value is an even number, the simple mean value of the two middle ranking values should be used. No "Weighted" or "Exponential" can be calculated for this function.

The MovingMed function takes the following arguments:

MovingMed (column, window, break_col)

where:

- column (required) Specify the column that contains the numeric fact on which the Moving Function column value is calculated. If you are applying the function from the Pivot section, you can only use a column that has already been added to the Facts pane of the Data Layout.
- window (optional) Specify a moving "window" of values from the *Column* on which the Moving Function gets calculated The window value must be a positive integer of value equal to or less than the total number of rows in the *Column* (within any *Break Column* value).

If the window value is an interger of greater value than the number of values in the *Column*, then the window value defaults to the number of rows in the *Column* (within any *Break Column* value). If no window value is specified, then the window value defaults to 3.

• break column (optional) - Specify the name of the column on which to perform the Moving Function calculation. The break column refers to the rows of similar valued data cell within the Pivot or Results section.

The following results would occur if you used the MovingMed function with a Window of 3 (MovingMed(Sales,3).

Month	Orignal Sales Values	3-Month MovingMed Average
Jan	10	
Feb	15	
Mar	17	15
Apr	20	17
Мау	22	20
Jun	20	20
Jul	25	22
Aug	27	25
Sep	30	27
Oct	35	30
Nov	37	35
Dec	40	37

Tahle	100	MovingMed	Function	Fxample
lavie	100	wowingwieu	FUNCTION	слаттріе

Moving Sum

MovingSum (Moving Sum) function operates over a moving window of values. For each Window, the MovingSum returns the sum of values found in the window. No "Weighted" or "Exponential" can be calculated for this function.

The MovingSum function takes the following arguments:

MovingSum (column, window, break_col)

where:

• column (required) - Specify the column that contains the numeric fact on which the Moving Function column value is calculated. If you are applying the function from the Pivot section, you can only use a column that has already been added to the Facts pane of the Data Layout.

- window (optional) Specify a moving "window" of values from the *Column* on which the Moving Function gets calculated The window value must be a positive integer of value equal to or less than the total number of rows in the *Column* (within any *Break Column* value). If the window value is an interger of greater value than the number of values in the *Column*, then the window value defaults to the number of rows in the *Column* (within any *Break Column*, then the window value is specified, then the window value defaults to 3.
- break column (optional) Specify the name of the column on which to perform the Moving Function calculation. The break column refers to the rows of similar valued data cell within the Pivot or Results section.

The following results would occur if you used the Moving Sum function with a Window value of 3.

Month	Orignal Sales Values	3-Month MovingSum Average
Jan	10	
Feb	15	
Mar	17	42
Apr	20	52
Мау	22	59
Jun	20	62
Jul	25	72
Aug	27	72
Sep	30	82
Oct	35	92
Nov	37	102
Dec	40	112

Table 101Moving Sum Example

Moving Minimum

The MovingMin function operates over a moving window of values. For each Window, the Moving Minimum returns the minimum value found in the Window. This function is similar to the Simple Moving Average. No "Weighted" or "Exponential" can be calculated for this function.

The MovingMin function takes the following arguments:

```
MovingMin (column, window, break_col)
```

where:

- column (required) Specify the column that contains the numeric fact on which the Moving Function column value is calculated. If you are applying the function from the Pivot section, you can only use a column that has already been added to the Facts pane of the Data Layout.
- window (optional) Specify a moving "window" of values from the *Column* on which the Moving Function gets calculated The window value must be a positive integer of value equal to or less than the total number of rows in the *Column* (within any *Break Column* value).

If the window value is an interger of greater value than the number of values in the *Column*, then the window value defaults to the number of rows in the *Column* (within any *Break Column* value). If no window value is specified, then the window value defaults to 3.

• break column (optional) - Specify the name of the column on which to perform the Moving Function calculation. The break column refers to the rows of similar valued data cell within the Pivot or Results section.

The following results would occur if you used the Moving Minimum function with a Window value of 3.

Similar to Simple Moving Averages above, note that the values in the "3-Month Moving Minimum" column are positioned according to the trailing averages.

Month	Orignal Sales Values	3-Month Moving Minimum
Jan	10	
Feb	15	
Mar	17	10
Apr	20	15
Мау	22	17
Jun	20	20
Jul	25	20
Aug	27	20
Sep	30	25
Oct	35	27
Nov	37	30
Dec	40	35

 Table 102
 Moving Minimum Example

Direction Of Moving Function Calculation

In all of the Moving Function calculation examples, it is assumed that the Moving Function calculation proceeds in a downward direction for each Computed Item column. That is to say, the Moving Function calculation window progresses downward, one row at a time, for each

subsequent row of displayed fact data. This downward movement is evident in the Table Section data (which has been used in the examples shown above) since, in a Table Section, each new instance of fact data can only be represented as a new row.

Other sections, however, such as Pivot and Chart allow the possibility for fact data to be represented in more than one directional axis. In the case of Pivot and Chart Sections, therefore, you can specify the direction of Moving Function calculation. To illustrate the effect of changing the direction of Moving Function calculation, consider the previously shown Table data:

Month	Sales
Jan	10
Feb	15
Mar	17
Apr	20
Мау	22
Jun	20
Jul	25
Aug	27
Sep	30
Oct	35
Nov	37
Dec	40

Table 103Moving Function Example

Adding a new Computed Item column to display a 3-Month Moving Difference would yield:

 Table 104
 3-Month Moving Difference Example

Month	Sales	3-Month Moving Difference
Jan	10	
Feb	15	
Mar	17	7
Apr	20	5
Мау	22	5
Jun	20	0
Jul	25	3
Aug	27	7

Sep	30	5
Oct	35	5
Nov	37	7

Consider, however, if the "Month" and "Sales" data were oriented as follows (as could be done in a Pivot Section):

	Jan	Feb	Mar	Apr	Мау	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Sales	10	15	17	20	22	20	25	27	30	35	37	40

If a new Computed Item were added, to represent a 3-Month Moving Difference, and the Moving Function calculation direction was still down each column, this would result in null values being represented for the Computed Item (as below) since there would only be one fact value per column (this would not be enough fact occurrences to satisfy a 3-term Moving Difference).

	Jan	Feb	Mar	Apr	Мау	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Sales	10	15	17	20	22	20	25	27	30	35	37	40
3-Month Moving Difference												

If, instead, the Moving Function calculation direction were specified to be along each row (i.e. horizontally from right to left) the Moving Difference Computed Item would yield the "expected" results since the source "Sales" Fact data cells would be examined in their "correct" left-to-right sequence. That is, the following display would result:

	Jan	Feb	Mar	Apr	Мау	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Sales	10	15	17	20	22	20	25	27	30	35	37	40
3-Month Moving Difference			7	5	5	0	3	7	5	5	7	5

1	3	Applying Sorts
	This chapt It includes and comp	ter discusses features that enable Interactive Reporting to sort data in various reports. Information or sort lines, single and nested sorts in the Query and Results section, lex sorting by values and labels in the report sections.
In This Chapter	Sorting Data Simple Sorts	

Sorting Data

Sorting simplifies the process of data analysis. After data is sorted, the answers to questions are often readily at your fingertips. Sorting is also very useful for ranking data to reveal business trends and margins.

Year	State	Item Type	Amounts	Units
1997	NY	The Standard	8,958.6	70
1996	NY	EZ Fax Modem-I	7,203.9	55
1996	MI	The Turbo	7,986.6	45
1996	MI	1024 MB Drive	10,438	40
1996	NY	The Turbo	11,536.2	65
1995	NY	1024 MB Drive	9,133.25	35
1996	CA	EZ Fax Modem-I	8,513.7	65
1995	CA	1024 MB Drive	10,438	40
		Unsorted Results		

1995 CA 1024 MB Drive 7,828.5 30 1995 CA 1024 MB Drive 10,438 40 1995 CA 1024 MB Drive 10,438 40 1995 CA 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 560 MB Drive 9,133.25 35 1996 CA 560 MB Drive 9,138.26 60 1996 CA EZ Fax Modern-I 7,858.8 60 1996 CA EZ Fax Modern-I 7,851.37 65	Year	State	ltem Type	Amounts	Units
1995 CA 1024 MB Drive 10,438 400 1995 CA 1024 MB Drive 10,438 40 1995 NY 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 560 MB Drive 8,428.8 60 1996 CA EZ Fax Modern-I 7,858.8 60 1996 CA EZ Fax Modern-I 8,513.7 65	1995	CA	1024 MB Drive	7,828.5	30
1995 CA 1024 MB Drive 10,438 400 1995 NY 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 560 MB Drive 8,428.8 60 1996 CA EZ Fax Modern-I 7,858.8 60 1996 CA EZ Fax Modern-I 8,513.7 65	1995	CA	1024 MB Drive	10,438	40
1995 NY 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 1024 MB Drive 9,133.25 35 1996 CA 560 MB Drive 9,138.26 60 1996 CA 560 MB Drive 9,428.8 60 1996 CA EZ Fax Modern-I 7,858.8 60 1996 CA EZ Fax Modern-I 8,513.7 65	1995	CA	1024 MB Drive	10,438	40
1996 CA 1024 MB Drive 9,133.25 35 1996 CA 560 MB Drive 8,428.8 60 1996 CA EZ Fax Modern-I 7,858.8 60 1996 CA EZ Fax Modern-I 7,851.37 65	1995	NY	1024 MB Drive	9,133.25	35
1996 CA 560 MB Drive 8,428.8 60 1996 CA EZ Fax Modem-I 7,858.8 60 1996 CA EZ Fax Modem-I 7,858.8 60 1996 CA EZ Fax Modem-I 8,513.7 65	1996	CA	1024 MB Drive	9,133.25	35
1996 CA EZ Fax Modem-I 7,858.8 60 1996 CA EZ Fax Modem-I 8,513.7 65	1996	CA	560 MB Drive	8,428.8	60
1996 CA EZ Fax Modem-I 8,513.7 65	1996	CA	EZ Fax Modem-I	7,858.8	60
	1996	CA	EZ Fax Modem-I	8,513.7	65



You can perform simple and complex sorts depending on your requirements. You can sort individual data items or use the Sort line to sort items in reference to associated data values. These complex sorts can be nested, so that the sorted data reflects the hierarchical relationships between data items.

Simple Sorts

If you only want to sort a single Request item, report row, or column, you can use the Sort buttons on the toolbar to quickly order the data.

If you apply simple sort conditions in the Query section, the database server sorts the data while processing the query before it is retrieved to your desktop. Or you can sort data on your desktop in Results or report sections. The data associated with a selected item is sorted in ascending or descending order as you wish.

- > To select items to sort:
- **1** Select the data item to sort.

In the Query section, select a topic item on the Request line. In other sections, select a corresponding report element in the Content pane.

2 On the Standard toolbar, click the ascending or descending Sort icon.

The data is sorted in the Content pane.

If the Sort line is visible, the item appears on the Sort line. The item name is followed by an up or down arrow to indicate the sort order.

Sort Lines

Interactive Reporting sort lines have two functions:

- To maintain a record of sort conditions that you have applied to the data set.
- To enable you to specify compound and nested sorts.

Sort lines take on a different appearance in each section depending on the data presentation and the types of sorts available.

- In the Query, Results, Table, and Report Designer sections, sort lines are drag and drop command lines similar to Request and Filter lines.
- In the Chart, Pivot, and OLAPQuery sections, sort lines are list driven and include provisions for sorting by aggregate calculations, such as averages and counts.
- **Tip:** You can move, size, dock, or hide the Sort line. To toggle the Sort line, click the Sort button on the Section title bar.

Complex Sorting

In addition to performing simple sorts, you can use Interactive Reporting to perform complex sorting. Review the following sections for information on:

- Complex Sorting in the Query, Results, and Table Sections
- Complex Sorting in Chart, Pivot, and OLAPQuery Reports

Complex Sorting in the Query, Results, and Table Sections

Note: The information discussed here also applies to sorting in the Report Designer section.

The appearance and functionality of the Sort line is nearly identical in the Query, Results, and Table sections. In each section, the Sort line uses a drag-and-drop interface similar to the Request and Filter lines.

Sort Now Units 🛆 Item Type 🛆 Amounts 🛆

Using this feature, you can drag items to the sort line and request them to be sorted in sequence to yield nested sort results. When you sort more than one data item at a time, the left to right order of data on the Sort line dictates the sort order and creates a nested effect. Data is sorted in the order you specify. The leftmost item on the Sort line is the primary sort. Items to the right are sorted in progression, each within the categories of the preceding item.

For example, if the first item is State, the second item City, and the third item Store, States are sorted alphabetically by name. Within each state, cities are sorted by name. Within each city, stores are sorted.

Nested effects are based solely on the placement of items on the Sort line. If the item order on the Request line differs from the order in the Content pane, the sort is still nested, but the visual impact is not as pronounced.

The key difference between the sections lies in where the sorting is performed:

- In the Query section, the database server sorts items placed on the Sort line as the query is processed. The data is returned already presorted to the Results section.
- In the Results and Table sections, items placed on the Sort line are sorted on your desktop.

The bottom line effect is the same whether you apply sort conditions locally in the Results or Table section or on the database server in the Query Section. Depending on the situation and the needs of your business, one method or the other may be preferable.

Complex Sorting in Chart, Pivot, and OLAPQuery Reports

Note: The information discussed here does not apply to sorting in the Report Designer section.

In Chart, Pivot, and OLAPQuery reports, you generally want to override the default sort order and sort dimensional data with reference to other data. In these sections, you can use the Sort line to impose a sort condition for each dimensional data item in your report, and to nest your sort conditions at each hierarchical level of the report from the outside in.

The Sort line includes three drop-down menus used to define the sort conditions. The contents of the menus vary depending on the data items in query.

Tip: Data in Chart, Pivot, and OLAPQuery sections is sorted alphabetically by default. You can use the sort buttons on the Standard toolbar to perform simple sorts on selected report items and reverse the sort order. (See "Simple Sorts" on page 340 for more information.)

Sort Items

The Sort drop-down menu lists the data items that can be sorted. Each dimensional item included in the report (name and date) is listed in this menu.

Reference Items

The By drop-down menu lists items used as a basis for a complex sort condition (for example, sorting Cities by the revenue generated in each). The drop-down menu includes each numerical data item in the report as well as the keyword entry "label." These choices provide two ways to sort the dimensional item specified in the Sort menu:

- Sorting by Label By default Interactive Reporting sorts dimensional data items alphabetically by name when you create your report this is equivalent to sorting by labels. When selected, *label* indicates that the item chosen from the Sort list is sorted by label or name, rather than by reference to corresponding numeric data values in the report.
- Sorting by Value Sorting by a numeric data item orders each value of the target item chosen from the Sort list by its corresponding numeric value in the Value list.

Sorting by values produces an entirely different sort order. For example, your chart may list each state in which your company has made sales revenue and the total cost-of-sales for each. The states are initially listed in alphabetical order. When you sort by cost-of-goods, the states are ranked in order by each corresponding cost-of-sales figure.

Functions

The Using drop-down menu contains aggregate statistical functions that are available when you sort by values. The functions generally duplicate the data functions available in the active section.

The default function for sorting is *Sum*. When you sort by values, Interactive Reporting sorts dimensional data by the corresponding numeric values of the referenced item (for example, sorting states by the sum total of the cost of goods sold in each state).

- > To specify a sort using the features of the Sort line:
- 1 If the Sort line is not already displayed, click **Sort** on the Section title bar.
- 2 Select an item to sort from the Sort drop-down list.
- **3** Select a **value** from the By drop-down list as a sort reference, or select **Label** to sort the item alphabetically.
- 4 If desired, select an aggregate function from the Using drop-down list when sorting by values.

The Using drop-down menu is not available when you sort by labels.

5 If desired, click **Sort** on the Sort line.

The Sort line stores a sort condition for each dimensional item included in the report.

Glossary

Adaptive Report Level The level of privilege that a user can interact with an Interactive Reporting document. The adaptive report levels are View Only; View and Process; Analyze; Analyze and Process; Query and Analyze; and Data Model, Query, and Analyze. When a document is published, groups of users are granted access to it with specific Adaptive Report level privileges.

ADR (also called as sync) See *automatic distributed refresh*.

aggregate filter Filters placed on aggregated request line items or aggregated meta topic items.

API Application Programmer Interface

API Socket (DaAPISock) The API (such as ODBC, SQL Net, and so on) or protocol level information used when connecting to a database that has been abstracted in an API Socket. Interactive Reporting have one API socket for each supported API/communication protocols.

application aerver A panework for developing applications that provides fundamental capabilities required by many applications such as session and resource management, and security.

auditing The monitoring of Interactive Reporting Repository objects to determine usage patterns.

automatic distributed refresh The process of synchronizing locally saved documents and the version in the repository.

auto-join An administrator can configure a connection file to cause joins to occur automatically for users using the Auto-Join feature. Auto-Join can be configured one of three ways: Best Guess, Custom, and Server-Defined.

auto-process The automatic processing of a query.

axes Straight lines on a chart used for measurement and categorization. Typically the X-axis and the Z-axis are both used for categories while the Y-axis is used for quantification (for example, Facts/Values). Pie charts only use the X-axis for its categories and the Y-axis for its Facts/Values.

catalog A collection of database tables and local results. This is the information the user can use in a data model or query.

Catalog pane Shows a list of elements available to the active section. For example, if Query is the active section, the Catalog pane displays a list of database tables. If Pivot is the active section, the Catalog pane displays a list of results columns. If Dashboard is the active section, the Catalog pane displays a list of embeddable sections, graphic tools, and control tools.

categories Groupings by which the data is organized (for example, month).

CGI See Common Gateway Interface.

chart A graphical representation of data. Users create charts to convert raw data into eloquent, visual information.

Chart section With a varied selection of chart types, and a complete arsenal of OLAP tools like group and drilldown, the Chart section is built to support simultaneous graphic reporting and ad-hoc analysis.

client/server A network architecture in which each computer or process on the network is either a client (requests information) or a server (delivers requested information). clustered bar charts Occurs when the categories are viewed side by side within a given category, useful for side by side category analysis. Clustering is only done with vertical bar charts.

Common Facilities One of the four categories of objects defined by the Object Management Architecture (OMA). The Common Facilities is a collection of services that many applications may share, but the services are not as fundamental as Object Services. For example, Print Facility and Mobile Agent Facility.

computed item A virtual column (as opposed to a column that is physically stored in the database or cube) that can be calculated by the database during a query, or by an Interactive Reporting in the Results section. They are calculations of new data based on functions, data items, and operators provided in the dialog box. They can be included in reports or reused to calculate other data.

connection file See *interactive reporting database connection file*.

Controls Folder Contains prebuilt items that can be added to your Dashboard section, such as list boxes, radio buttons, and command buttons. You can then attach scripts to the controls embedded in the Dashboard section to execute actions.

correlated subqueries Subqueries that are evaluated once for every row in the parent query. A correlated sub query is created by joining a topic item in the subquery with one of the topic items in the parent query.

cross join Creates a query where none of the tables is joined. Every row in one table is joined to every row in another table.

cube The data in OLAP databases (MS Plato, Essbase, MetaCube) is stored in *3-dimensional* cubes, which is different from standard relational databases (*2dimensional*). Cubes are made up of dimensions and measures. A cube may have dozens of dimensions.

DaConnect (aka DaConn) An Interactive Reporting's main connectivity class. Contains all the connectivity specific information. Instance of a connection class is made whenever Interactive Reporting needs to connect to any database.

Daemon-mode BQ The Hyperion client/server executable. It is spawned in daemon mode by node JREs. The role of the daemon-mode BQ is simply to execute database queries and return results.

Dashboard Allows you to build and deploy analytic applications. Analytical applications focus on delivering significant prepackaged business content that is encapsulated within an application. With Interactive Reporting, customers and independent software vendors can leverage their knowledge of industry specific best practices and assemble their own analytical applications for deployment to end users using the Web. Combining embeddable Report sections with enhanced Dashboard controls into a visual workspace and hooking up interactivity between these controls and the native Brio Software application, an applications.

Dashboard Home A button that returns you to the Dashboard section designated as the Dashboard Home section. If you have only one Dashboard section, Dashboard Home returns to that section. If you have several Dashboard sections, the default Dashboard Home is the top Dashboard section in the Catalog pane. In Design mode you can specify another Dashboard section to be the Dashboard Home section.

DaSession When a client needs to perform an operation such as select, insert, update on a database, a session is created. The life time of a session synchronizes with the operation's duration. Sessions are created when a client needs to perform operations on the database and are destroyed when the operation is considered complete.

data function Computes aggregate values, including averages, maximums, counts and other statistics, which summarize groupings of data. You can use data functions to aggregate and compute data from the server before it reaches the Results section, or compute different statistics for aggregated totals and items in the other analysis sections.

data model A representation of a subset of actual database tables that acts as a menu for the query-builder. Data models are the sources for building the Request, Filter, and Sort lines in a query section. You create a data model by dragging database tables from the Catalog pane into the Content pane. The data model displays database tables graphically as topics when they are in the Content pane. Data models can be distributed through the Hyperion System 9 BI + repository and used by end users to create their own queries.

database function A predefined formula in a database.

database server A computer that stores database management system software (DBMS, for example, Oracle, Sybase, Essbase), and a database shared by a network of computer clients. Most databases are used in a client/server environment.

By storing data on a single, powerful machine on a network, the data is centralized and accessible to many users. The server ensures that the data is maintained correctly and serves as a traffic cop to regulate client machine access to the data. The server's computing power is also useful in computing and filtering data from the database before it reaches your workstation. For small or localized databases, your own computer can act as both database server and client.

datatype The type of data stored in a specific column in a database. For example, data can be stored as a numeric datatype.

date group A feature in the Results and Table sections that separates a date into Year, Quarter and Month columns. The display format for the new Month column is automatically set to *mmm* so that the month names sort chronologically (as opposed to alphabetically) in the report sections. Quarters are based on the calendar year, beginning January 1.

designer A client/server-based tool that delivers query, analysis, and reporting capabilities and centralized solution administration for developers, database administrators, and system administrators.

design guide Design guides are similar to grids in that objects automatically snap to line up with the design guides. Design guides are placed at user-specified locations in the work area. There are horizontal and vertical design guides. The design guides will draw as a blue line across the work area, with the ability to drag the guide to any new location.

design mode A toggled environment used to build and modify Dashboard sections. In Design mode, you build Dashboard sections with an array of prebuilt control items and graphic items. Designers can switch out of design mode to test Dashboard features and deploy them to end users. Design mode is only available in Dashboard sections. Interactive Reporting Navigator users cannot switch to Dashboard Design mode.

detail view Displays a topic as a database table. When you select Detail view, the database returns ten sample rows from the associated table. Each topic item displays as a database field. Detail view enables users to browse a sample of the raw data, which is useful when unfamiliar with the data model or the underlying data. Users cannot view a meta topic in Detail view.

dimension In an OLAP database cube, categories of information are called dimensions. Examples of dimensions may be Location, Products, Stores, and Time. In an Interactive Reporting, related, nonquantifiable items in a topic are also referred to as dimensions, such as Contact or Store Name.

dimension (legend dimension) The current axis categorization or grouping method. This can be set to the X, Y, or Z-axis for most chart types. For line and area charts it can only be set to the Y or Z-axis. For pie charts it is always set to the X-axis. When you change the Legend Dimension the color is distributed along the new axis you change to.

dimension tab In the Pivot section, the tab that enables you to pivot data between rows and columns.

dimension table Consists of numerous attributes about a specific business process. Each row in a dimension table is unique.

drill anywhere Allows you to drill into and add items to pivot reports residing in the Results section, without having to return to the Query section or trying to locate the item in the Catalog pane. Drill Anywhere items are broken out as new pivot label items.

drill to detail Allows you to retrieve items from a data model that are not in the Results section, without having to rerun the original query. This feature provides the ability to interactively query the database and filter the data that is returned. Drill to Detail sets a filter on the query based on the user's selection and adds the returned value as a new pivot label item automatically.

drilldown Allows you to progressively narrow your focus on a selected chart category. Very useful when you have too many categories on a particular axis. embedded section A Interactive Reporting section that is embedded in a Smart report or other Interactive Reporting section. All embedded sections maintain live data content and formatting from the original section. Reprocessing the query, or modifying the original section automatically updates the embedded section display.

Explorer A client/server-based tool that delivers query, analysis, and reporting capabilities for power users who need to directly access data sources—or to explore the information organized in prebuilt data models stored in the repository.

Expression Line Displays displays the JavaScript syntax for each item displayed in a report. Use this line to build equations in the Report section. For ease of use, it can be undocked and resized.

extranet An intranet that is partially accessible to authorized outsiders. Extranets are secured by user names and passwords.

fact table A table that stores business activity measures. Most fact tables are extremely large. Each row in a fact table contains numeric measures (fully additive measures, nonadditive measures and/or semiadditive measures) and foreign keys to each dimension table.

facts The numeric values that are broken up in the body of the Pivot section. To add facts to the Pivot, put Results columns from the Catalog pane into the Outliner's Facts pane. Facts are the numeric values in a relational database that are available to analyze. In an OLAP Query, they are called measures.

facts/values The data that is being visually represented, usually a numeric amount (for example, \$15,000)

file server A computer and storage device dedicated to storing files.

filter Constraints placed on topic items or request line items to filter them to a certain set of values. Filters appear on the filter line in Interactive Reporting. For example, although the database may display worldwide sales figures for all stores, you may only want to see sales for stores in Germany. Filters make data sets retrieved through a query more efficient and manageable by filtering out unnecessary information.

filters In the OLAPQuery section, filters enable you to define and apply filters to the query once Top or Side Labels have been added to the query. You set a filter by applying comparison operators on the values for a specific member. Additional server-specific functions are available based on the selected OLAP database cube. Filters are built to include or exclude data according to specific criteria. Filters can be set on any level in a dimension. However, they must be one of the labels in the Outliner. Filters can be made of members of one of the dimensions or selected by a measure. OLE DB for OLAP has other filter operator types (Top N, Top Sum, Top N %, Bottom N, Bottom N %, Bottom Sum.) Essbase and MetaCube have their own operator types. In Essbase and MetaCube, filters can be done on measures. In OLE DB for OLAP, filters cannot be placed on measures.

foreign key A database column or set of columns included in the definition of a referential integrity constraint.

fully additive measure Attributes in a table that can have their values added together across any dimension.

grain The level of detail at which measures in a table are recorded is referred to as the grain.

grouping columns This feature, in the Results and Table sections, creates a new column in a dataset by grouping data from an already existing column. Grouping columns consolidate nonnumeric data values into more general group values and map the group values to a new column in the dataset.

hardwire mode In hardwire mode whenever the OLAP Query is changed, the database is queried to fetch new cube data. In contrast, process mode is manually controlled. You add or remove several items to the Outliner, and then press Process to query the database. When determining whether to use hardwire mode or process mode, consider the size of the cube in which you are working.

hierarchy In an OLAP database cube, a hierarchy organizes a dimension's levels and corresponding members into parent and child relationships. For the levels in a Location dimension, the hierarchy would have Country as the parent of the child City and City as the parent of the child Address. **home** Abstract base interface that some Lightning interfaces derive from. Provides basic methods to manipulate and create Bean objects.

HTML See Hypertext Markup Language.

Hypertext Markup Language A programming language used to create World Wide Web pages, with hyperlinks and tags that explain how to format the information on the screen.

icon view Shrinks a selected topic to an icon in the Content pane. The topic remains part of the data model, but is deactivated and can not be accessed by the query. Associated items are removed from the Request line when a topic is made into an icon, and the topic is not recognized as joined to other topics. Icon view is helpful in restricting the use of server time when a topic is infrequently used, and does not have to be active at all times.

imported Files Excel, Text, or CSV files imported into Interactive Reporting. The information in the file goes into a table or a results object. Imported results can be used in other queries (like regular results) for local join purposes.

Indexes Indexes are created in a database to increase the performance of data retrieval. Just as book indexes help to locate specific information faster, database indexes provide a faster access path to table data. Indexes are created on one or more columns of a table.

integrity constraints Constraints on tables that guarantee the data adheres to certain business rules. Integrity constraints are defined with a table and are stored as part of the table definition, central to the database data dictionary, so that all database applications adhere to the same set of rules.

internal function Built-in formulas, defined in the Interactive Reporting application.

interactive reporting database connection file Encapsulate and store connection information used to connect Hyperion applications to a database. Interactive Reporting Database Connection file specify the database API (ODBC, SQL*Net, etc.), database software, the network address of the database server, and your database user name. Once created, a user can specify the Interactive Reporting Database Connection file and database password and logon. An Interactive Reporting Database connection file is required for a Interactive Reporting document to use a database. the file extension is .oce.

Interactive Reporting Repository A specific group of tables created using Interactive Reporting on a database server and used to store Interactive Reporting document objects, including data models and standard queries. Administrators can use the Designer client to upload these objects to the repository for end-users to download, providing a template for query and report building. Documents built from repository objects can be versioncontrolled or audited through a link to the repository.

internet A global network connecting millions of computers. Unlike online services, which are centrally controlled, the Internet is decentralized by design. Each Internet computer, called a host, is independent.

interval Equal subdivisions within a given scale. The interval can be set manually or to best fit.

intranet A network belonging to an organization, usually a corporation, accessible only by organization members, employees or other authorization users. Intranet Web sites look and act just like any other Web site, but the firewall surrounding an intranet fends off unauthorized access.

ISAPI This is a shared library that implements HTTP using Microsoft's Web server plug-in API (ISAPI). Thus it is appropriate for customers with Microsoft Web servers. Functionally it is equivalent to our CGI. Its advantage is that it is far more scalable than a CGI executable.

item A visual representation of a database column that is a member of a topic in the Query section. Items are used to create queries and reports. For example, the Customer Topic may have items including Name, Address, and Phone. You select items from data model topics to build the Request, Filter, and Sort lines in the query section. JavaScript The scripting language for Interactive Reporting products. Interactive Reporting 8.0 include the Netscape JavaScript interpreter (version 1.4). JavaScript and Dashboard's Object Model allow application developers to use the full functionality of the industrystandard scripting language to control Interactive Reporting applications.

job repository A set of database tables that store a queue of scheduled jobs. There can be multiple job repositories in an organization.

join A relational database concept indicating a link between two topics. A join typically occurs between identical or similar items within different topics. Joins allow row records in different tables to be linked on the basis of shared information in a column field. For example, a row record in the Customer table is joined to a related record in the Orders table when the Customer ID value for the record is the same in each table. This allows the order record to be linked with the record of the customer who placed the order. If you request items from unjoined topics, the database server has no way to correlate the information between the two tables and leads to awkward datasets and run-on queries.

Interactive Reporting display joins visually in the workspace between topics to indicate joins between database tables. Users can also create new joins that are not already specified in the database.

join path A predetermined join configuration for a data model. Administrators create join paths for users to select the type of data model needed in a user-friendly prompt upon processing a query. Join paths ensure that the correct tables in a complex data model are used in a query.

JRE (Java Runtime Environment) This is the Java interpreter used to run the Java Server.

legend box An informative box containing color-keyed labels to identify the data categories of a given dimension.

layer Stacks a single object in relative position (sends back and front, or brings forward or backward) to other objects.

level Similar types of members in an OLAP database cube are grouped at the same level. For example, using the members listed in a Location dimension, France, the USA, and Japan belong to the Country level. Paris, Palo Alto, and Tokyo belong to the City level. 35 Main Street belongs to the Address level.

limit joins Joins between a database table and a local results object. The topic item being joined is limited by the values of the column being joined in of the local results object. A limit join is one of the options that you can choose in a Modify Join operation between a topic item and a local results item.

linked data model Documents that are linked to a master copy in a repository. When changes are made to the master, users are automatically updated with the changes when they connect their duplicate copy to the database.

local computed meta topic items Computed item definitions evaluated by the Interactive Reporting results engine. Local computed items are created to be meta topic items. They can be dragged to the request line like regular topic items. The only difference is that the results engine evaluates these items as opposed to the database.

local joins A join between a local results object and a database table or another local results object. Interactive Reporting perform the actual join in this case.

local filters Filters placed on the local dataset in the Results section, as opposed to the Query section. Filters in the Query section restrict the data retrieved by the query to the desktop. Local filters screen data from view in the Results set; although it is still there, you cannot see the data that has been excluded or use it in reports unless the filter is removed.

local Results Results of other queries within the same data model. These results can be dragged into the data model to be used in local joins. Local results are displayed in the catalog when requested.

locked data model Data Models that are locked cannot be modified by a user.

Manager See Server.

master data model A Data Model that exists independently and has multiple queries that reference it as a source. When using a master data model, the text "Locked Data Model" appears in the Content pane of the Query section. This means that the data model is linked to the master data model displayed in the Data Model section, which may be hidden by an administrator.

MDX (Multi Dimensional eXpression) The language used to give instructions to OLE DB for OLAP- compliant databases (MS Plato), as SQL is the language used for relational databases. When you build the OLAPQuery section's Outliner, Interactive Reporting translate your requests into MDX instructions. When you process the query, MDX is sent to the database server. The server returns a collection of records to your desktop that answer your query. *See* SQL.

measures Numeric values in an OLAP database cube that are available for analysis. Measures may be margin, cost of goods sold, unit sales, budget amount, and so on. *See* Facts.

members In an OLAP database cube, members are the content values for a dimension. In the location dimension, they could be Palo Alto, Paris, Tokyo, 35 Main Street, USA, France, Japan, and so on. These are all member values for the location dimension.

metatopic A customized, virtual topic, built from regular topics that reflects the exact topic and item structure of database tables. Metatopics allow items from disparate topics to be consolidated in a single topic, simplifying its appearance and reducing its conceptual resemblance to the underlying database structure. You can choose to view a data model in terms of its original topics, metatopics, or a combination of both.

metadata Data about data. Stored in database tables, metadata describes the history, content, and function of database tables, columns and joins in understandable business terms. Metadata can overcome the awkward names or ambiguous abbreviations often used in a database. For example, in a table named CUST_OLD, metadata may use a descriptive business name, such as Inactive Customers. **mime type** A browser mapping of a file type to either a helper application or a plug-in. When a browser attempts to open a file of a particular mime type, it either loads the associated plug-in or launches the associated helper application. A file's mime type is determined either by a) the file extension or b) the HTTP header. Plug-ins tell browsers what mime types they support and what file extensions correspond to that mime type.

Hyperion's web based products support the following mime types: *application/x-brioquery* mime type (for .bqy files). This is the default mime type our web based products support and are ordinary Hyperion files. *application/x-brioquerydata* (for .bqd files). These are data files in text or Excel format, whose extension has been changed to .bqd. When a Hyperion Web client is launched to open a BQD file, it imports the data and executes any Interactive Reporting Studio or Interactive Reporting Web Client script or JavaScript the file contains.

morphing Mechanism by which Hyperion's web based products provide document security.

multidimensional database A database that stores data in a format often referred to as a cube, such Essbase, MS OLAP, MetaCube, and so on. *See also* Relational database and OLAP database.

nonadditive measure Attributes in a table that cannot be added across any dimension, such as a percentage value (for example, margin rate).

NSAPI A shared library that implements HTTP using Netscape's Web server plug-in API (NSAPI). It is equivalent to our CGI. Its advantage is that it is far more scalable than a CGI executable.

Nesting Nesting means that one set of data values appears as a sub-division within each of the data values at a higher level of data. For example in the Pivot section, if you place more than one data label in an Outliner panel, the Pivot report displays the second set of labels inside each of the labels of the first data items. The second labels are nested within the first. This means they represent sub-divisions with another "higher level" category.

null value A null value is absent of data. Null values are not equal to zero.

OCE See interactive reporting database connection file.

OCE Wizard Wizard or set of screens used to create a new OCE or modify an existing OCE.

OLAP database A database that stores its information in cubes. Cubes contain dimensions and measures. A cube can have dozens of dimensions. Cubes are built to hold aggregated data, which anticipate how users think about business models. Cubes deliver this information efficiently and quickly.

OLAPQuery section Analyzes and interacts with data stored in an OLAP cube. When you use Interactive Reporting to connect to an OLAP cube, the document immediately opens an OLAPQuery section. The OLAPQuery section displays the structure of the cube as a hierarchical tree in the Catalog pane. Queries are built by dragging measures and dimension levels or members directly into the Outliner panes.

OOA Object Oriented Analysis.

OOA&D Object-Oriented Analysis & Design.

IOpen Client DBLib API to connect to Sybase, Redbrick, SQL Server, and so on.

Open Metadata Interpreter The Tables, Columns, Joins, Lookups and Remarks tabs available in the connection wizard when you edit a custom metadata source. These tabs allow Hyperion administrators to specify a customer source of metadata that can be accessed through SQL statements, and provided to end users with data models.

Outliners Drag-and-drop command lines used in the Pivot, Chart, OLAPQuery and Report sections. Each Outliner pane corresponds to a specific layout element of the report. When an item is dragged to an Outliner pane, the item assumes the layout attributes of the respective report element. Data appears simultaneously in the Content pane with the appropriate formatting.

pivot dimension row or column of labels that corresponds to an item in the Catalog pane.

Pivot section The section used to create crosstab reports and analyze data.

pivot table Analytical tools that resemble spreadsheets or crosstabular reports. A pivot table overlays a dynamic datacube, which allows data to be sliced and diced for adhoc, interactive, and multidimensional analysis.

pivoting In the Pivot section, the ability to change a label from a top to a side (or a side to a top) orientation with a simple click and swing of the label's Dimension tab.

plot area The area bounded by the X-, Y-, and Z-axis. For pie charts, it is the rectangular area immediately surrounding the pie.

predefined drill paths Enables a user to drill directly to the next level of detail, as defined in the data model.

primary key A database column or set of columns included in the table definition of the PRIMARY KEY constraint. Primary key values uniquely identify the rows in a table. Only one primary key is defined per table.

query A set of database instructions to return an answer set to a specific question. Each row returned in the Results section of a document is an answer to the question posed in the Query section.

query computed items Item definitions created by the user. This can include other request line items or topic items and or database functions. The definition is sent to the database and the database evaluates them.

Query Log Log of all SQL statements sent to the database (also referred to as SQL Log).

relational database A database that stores its information in tables related or joined to each other by common pieces of information called keys. Tables are subdivided into column fields that contain related information. Column fields have parents and children. For example, the Customer table may have columns including Name, Address, and ID number. Each table contains row records that describe information about a singular entity, object, or event, such as a person, product, or transaction. Row records are segmented by column fields. Rows contain the data that you retrieve from the database. Database tables are linked by Joins. (*See also join*)

report Group In the report section, embedded reports and tables are grouped by other data items. Items placed in the Groups Outliner break information into these dimensional groupings. For example, your table may include the name, contact information, and sales for each of your distributors. This table gains in clarity when broken into groupings that classify the stores by geographical region, year, or both. **Report section** A dynamic, analytical report writer, that provides users with complex report layouts and easy to use report building tools. Pivot tables, tables, and charts can be embedded in a report. The report structure is divided into group headers and body areas, with each body area containing a table of data. Tables are created with dimension columns and fact columns. These tables are elastic structures. Multiple tables can be ported into each band, each originating from the same or different result sets.

repository Central location used to store data models, queries and queries with reports. Repository is usually a database chosen by the user.

Request Line Holds the list of items requested from the database server and that will appear in the user's results.

Request Line items Columns listed in the request line.

Results section A section in an Interactive Reporting document that contains the dataset derived from a query. Data is massaged in the Results section for use in the report sections.

run mode A toggled environment used to test Dashboard sections. It simulates a Navigator user's view of the section. In Run mode, you can not add any features, but you can use features that are part of the deployed Dashboard sections. *See also Dashboard* and *design mode*.

scalar function Scalar functions (in contrast to data functions) do not aggregate data or compute aggregate statistics. Instead, scalar functions compute and substitute a new data value for each value associated with a data item.

scale The range of values that allow you to gauge how much each category represents. This range can be either at equal intervals or at logarithmic interval. The scale can be set manually or to best fit.

script A series of instructions for a computer. Scripts are activated when an event occurs, such as clicking a button or selecting an item from a drop down list. Interactive Reporting's scripting language is JavaScript.

Section pane Lists all the sections that are available in the current Interactive Reporting document.

Section Title Bar A navigational aid under the toolbars that provides a means of moving between sections and toggling section-specific tools and gadgets, such as the outliner. **semiadditive measure** Attributes in a table that can be summarized across some dimensions, but not all.

server computed meta topic items Metatopic item definitions created by the user. These items can use any of the database functions available, or any of the other topic items in the data model. These items are evaluated by the database.

Session Socket (DaSessionSocket) Abstracts session information specific to each database or API.

simple join Retrieves rows to create a query where the values in joined columns match.

slicer An axis that filters the data in an OLAPQuery. Only individual members can be used in a slicer. A slicer can be thought of as a third axis in a OLAP Query. The other axis are the Side Labels and the Top Labels. Every dimension folder contains a members subfolder named "Values for..." that dimension. This subfolder contains the members that are eligible for the slicer.

snapshot A a local (readonly) copy of table data that originates from one or more remote master tables.

spring A tool which allows you to maintain O relative verticalspacingbetweendynamic

objects. That is, you can "spring" one object to another so that if the first object is moved, increased or diminished, the second object moves in the same flow.

sort Conditions placed on request line items to sort the results in ascending or descending order. These are displayed in the sort line in an Interactive Reporting.

SQL See Structured Query Language.

SQL Net Oracle's native API to connect to an Oracle database.

stacked charts A chart where the categories are viewed on top of one another for visual comparison. This type of chart is useful for subcategorizing within the current category. Stacking can be used from the Y- and Z-axis in all chart types except pie and line. When stacking charts the Z-axis is used as the Fact/Values axis.

Stored Procedure Precoded queries in languages other than SQL. This is a feature available in some database software. Interactive Reporting can run stored procedures and retrieve the Results.

Structure View A view that displays a topic as a list of component items allowing users to see and quickly select individual data items. Structure view is the default view setting.

Structured Query Language The language used to give instructions to relational databases. When you build the query section's Request, Filter, and Sort lines, Interactive Reporting translate your requests into SQL instructions. When you process the query, the SQL instructions are sent to the database server. The server returns a collection of records to your desktop that answers your query. This reply is displayed as the Results section. You can look at the SQL generated by a query in the Query Log, or code a query directly in SQL using the Custom SQL window.

subquery A query embedded within another query.

surface values A setting in the Pivot section to base aggregate calculations on the values in the report, rather than the values in the Results section.

synonym An alias for a database table or view. It is a direct reference to a table view.

table The basic unit of data storage in a database. Database tables hold all of the user-accessible data. Table data is stored in rows and columns.

Table CatalogDisplays tables, views, and synonyms towhich users have access.Users drag tables from the Tablecatalog to the Content pane to create data models in theQuery section.

Table section Used to create tabular-style reports. It is identical in functionality to the Results section, including grain level (table reports are not aggregated). Other reports can stem from a Table section.

top and side labels In the Pivot section, labels are the column and row headings on the top and sides of the pivot. These define categories by which the numeric values are organized.

topic A a visual representation of a database table in the Content pane. Topics are part of data models displayed in the Query section and can contain one or more items.

topic items Individual items in a topic or metatopic.

topics Visual representation of tables in the database, related by joins that link certain items in each topic. Each topic title bar displays the topic's name. The topic shows a list of items, one for each column in the database table.

underlying values Another name for Results values. When Use Surface Values is disabled in a Pivot section, aggregate calculations are based on values in the Results section.

Union/Intersection/Minus Queries Queries created to perform set operations such as Union, Intersection, and Minus. These queries are created by the Append Query option.

variable filters Filters that prompt users to enter or select filter values before the queries are processed on the database.

view A custom-tailored presentation of the data in one or more database tables. Views do not actually contain or store data; rather, they derive their data from the tables on which they are based, referred to as the base tables of the views.

web client A Web tool that delivers query, analysis, and reporting functionality for intranet, Internet, or extranet access to information. Based on user profiles or reportlevel security, the client environment adapts in six stages from full query, analysis, and reporting with data refresh to static report viewing.

Weighted averages Weighted averages can also be used to apply different levels of importance to a given item. To calculate a weight you take the expected amount divided by the actual amount.

World Wide Web (WWW) A system of Internet servers that support specially formatted documents. The documents are formatted in a language called HTML (HyperText Markup Language) that supports links to other documents, as well as graphics, audio, and video files.

zoom Sets the magnification of a report. Sets the magnification of the report. The report can be magnified to fit the whole page, page width or a percentage of magnification based on 100%.

Index

Symbols

```
% Increase function, 178
% of Category function, 256
% of Column function, 177
% of Grand function, 177, 214
% of Row function, 177
(), 269
.bqtools.ini, 93
.bqy, 73
.csv, 48, 55
.sql, 72
.txt, 48, 55
.xls, 55
<, 273
<=, 273
<>, 273
=, 273
>, 273
>=, 273
```

Numerics

2-D bar charts, 2082-D bar charts, using to analyze data, 2083-D bar charts, 2093-D View, 208

A

Abs function, 295 Add Computed Item command, 122, 132, 141, 197, 237, 264 Add Cume command, 197 Add Date Group command, 132, 141 Add Fact/Measure command, 169 Add Filter(s) command, 122 Add Grouping Column command, 132, 141

Add Request Item(s) command, 122 Add Selected Items command, 197 Add Side Label command, 169 Add Sort(s) command, 122 Add Top Label command, 169 Add Totals command, 197 adding computed items, 289 footers, 84 headers, 84 sections, 82 adding computed items, 137, 181, 215, 255 adding cumulative totals, 176 adding date groups, 139 adding grouping columns, 137 adding lines to labels, 207 adding other report elements, 247 adding report groups, 245 adding totals and subtotals, 175 AddMonths function, 294 Administer Repository command, 94 aggregation, 27, 113 analyzing data based on day of week data, 303 AND, 269 Append Query command, 122 applying data functions to tables, 139 area charts, 210 to 211 ascending sorts, 141, 237, 264 Ascii function, 297 Atn function, 295 AutoAdd Columns command, 132 Automatic data type, 285 automatically adding columns (Results section), 128 Average function, 139, 177, 214, 256 Avg function, 308 Avg function, 296

AvgNonNull function, 309 AvgNonNull function, 296 axes, 201 axis grid lines, showing, 220

B

bar, 201 bar charts 2-D, 208 3-D, 209 clustered, properties, 232 properties, 231 stacked, 210 three-dimensional, 209 two-dimensional, 208 bar-line charts, properties, 232 **BEGINS WITH, 273** BETWEEN, 273 Border and Background command, 78 bgtbls5.ini,112 bqtools.ini, 93 BQY files, 73 Break Total command, 132, 141 break total data functions, 139 break totals, 127, 140 browser restrictions and limitations, 71 Byte data type, 285

C

Catalog pane, 44 Ceil function, 295 changing chart color schemes and fill patterns, 233 changing chart data labels, 233 changing chart legends, 220 changing color of chart elements, lines, and text, 236 changing column names, 244 changing label nesting levels, 174 Chart, 200 chart angle and elevation, changing, 223 chart dimensions, 202 chart elements, working with, 207 chart format, determining what to use, 205 chart label properties, 229 Chart menu commands, 237 chart properties, changing, 227 Chart section applying sorts in, 216

computed items, 215 Outliner, 205 Chart This Pivot command, 173 charting a pivot table, 173 charting basics, 200 charts adding computed items, 215 adding lines to labels, 207 area, 210 building, 205 combination, 212 creating, 205 creating from pivot tables, 173 Data Layout, 205 inserting text, 220 line charts, 212 multidimensional, 208 pie charts, 206 pivoting, 217 ribbon charts, 211 sorting, 37 stacked bar, 210 terminology, 200 three-dimensional, 208 three-dimensional bar, 209 two-dimensional, 206 two-dimensional bar, 208 chr function, 296 Clear command, 76 clustered bar charts properties, 232 understanding, 209 ColMax function, 310 ColMax function, 296 ColMin function, 311 ColMin function, 296 color, adding, 36 color, changing in charts, 236 Column command, 78 Column commands Add Grouping, 141 Hide, 142, 237, 264 Modify, 141 Unhide, 142, 237, 264 Column Name function, 256 column names changing, 244

hiding, 244 Column Titles command, 78 column totals, 140 columns, 32 to 129, 138 to 141, 244 to 245, 253 combination charts, 212, 232 comma-delimited files, 48, 55 Command lines, 43 commands Chart menu, 237 Edit menu, 75 File menu, 75 Format menu, 78 formatting, 129 Pivot menu, 197, 311 Report menu, 264 Results menu, 132 Table menu, 141 Tools menu, 93 View menu, 76 commands, Help menu, xv comparing related values, 213 complex sorting in OLAPQuery section, 342 in Query, Results, and Table sections, 341 components, report, 243 compressing documents, 46 computed fields, 248 to 249 computed items adding, 141, 237, 264, 289 Chart section, 215 compared to data functions, 289 data types and specifications, 285 examples, 304 in Interactive Reporting sections, 284 in OLAP queries, 153 in Query section, 285 in Results and report sections, 286 Report section, 255 Table section, 137 computing new data, 284 Concat function, 297 concatenating report element syntax, 242 Connection commands, 93 connection files, 24 Connections Manager command, 93 Console Window command, 76

consulting services, xvii CONTAINS, 273 Content pane, 44 contents, document, 23 converting detail reports, 262 to 264 Copy command, 75 correlated subqueries, 104 Cos function, 295 Cosh function, 295 Count function, 139, 177, 214, 256 Count function, 295 to 296 CountDistinct function, 296 CountNonNull function, 296 CountNull function, 296 creating charts, 205 custom reports, 242 pivot tables, 172 pivot tables from charts, 217 smart reports, 261 three-dimensional bar charts, 209 creating documents, 73 CSV files, 48, 55 Cume function definition, 296 cumulative filters, 176 cumulative totals, adding, 176 custom functions, 139 custom menu items, 92 Custom SQL command, 77 custom totals, inserting, 126, 139 custom values, 272 Customize command, 94 Customize Filter command, 122 customizing chart patterns, colors, and labels, 232 customizing chart properties and labels, 227 customizing filter options, 277 Cut command, 75

D

data calculating, 33 charting, 36 exporting, 55 filtering, 33 hiding, 36

importing, 48 pivoting, 34 refining in OLAPQuery, 151 Data Function command, 197, 237, 264 data functions compared to computed items, 289 in charts, 214 in custom reports, 256 in OLAP queries, 165 in pivot tables, 177 in queries, 113 Data Functions command, 122, 169 Data Layout Chart section, 205 Pivot section, 173 Table section, 134 data layout, 43 data models, 25 data source connections, 24 data sources, working with, 48 data types, 124 data types and specifications, 285 data, computing, 284 data, pivoting, 173 database definition, 23 displaying remarks, 113 viewing tables, 31 Date & Time field, 249 Date & Time Now field, 249 Date command, 87 Date data type, 285 Date field, 249 Date Group command, Add, 141 date handling, 89 Date Now field, 249 dates, breakout columns, 128, 139 day of week, returning, 303 DayOfMonth function, 294 Decode function, 294 default formats, 86 default number formats, 59 Delete Section command, 76 deleting columns, 141 deleting sections, 85 descending sorts, 141

design guides, 251 Design Guides command, 264 Detail reports, converting, 262 dimensions, definition, 124 display differences, 262 display options Essbase, 147 displaying axis grid lines, 220 displaying rulers, 251 document sections. See sections. documents compressing, 46 contents of, 23 conventions used, xvi exporting as Web pages, 70 feedback, xviii password protecting, 46 saving, 45 structure of, xiv documents, accessing Hyperion Download Center, xv Hyperion Solutions Web site, xv Information Map, xv online help, xv Download To Results command, 169 Drill Anywhere command, 237 drill down, 35, 152 Drill Down command, 169, 197 drill up, 152 Drill Up command, 169, 197, 237 drilling into charts, 218 drilling through, multi-dimensional to relational, 166 Duplicate Section command, 76, 83 duplicate values, suppressing, 244

E

editing footers, 84 editing headers, 84 education services, xvii elements selecting pivot, 173 working with chart, 207 empty rows, suppressing, 148 ENDS WITH, 273 EQUAL, 273 Essbase

applying measure filters, 159 attribute dimensions, 145 command language, 26 display options, 147 query language, 26 substitution variables, 161 Estimate Query Size command, 122 Excel files, 55 Exp function, 295 export documents as Web pages, 69 HTML Wizard, 71 properites, 50 to 51 properties, 78 query logs, 72 sections, 55 SOL, 72 Export commands, 75 Export Properties command, 78 Expression line, 241 expression syntax, 241

F

facts, 172 facts, definition, 124 features, sort line, 216 field definitions, number, 87 fields, 248 file locations, 89 File menu commands, 75 File Name field, 249 files, exporting, 55 Filter command, 132, 141 filter dialog box, 270 filter line, 269 filter line syntax, 270 filter options, customizing, 277 filter, operators, 273 filtering data in a table, 135 filtering queries, 274 filtering results, 275 filters applying in OLAPOuery, 158 member selection, 158 cumulative, 176

measure applying, 159 variable, 160 member variable, 159 removing, 136 tables, 135 variable, 159 in OLAP Query, 159 slicer, 160 filters, applying in OLAPQuery, 158 filters, cumulative, 176 focus on items, 218, 258 Focus On Items command, 197, 237, 264 Font command, 78 font styles, changing, 261 fonts, changing, 261 fonts, default, 86 footers, 84 footers, page in custom reports, 246 Format menu commands, 78 formats default fonts, 86 number, 87 specifying, 86 formatting commands, 129 formatting day of week data, 303 formatting report items, 261 functions data, 165 in sorts, 217, 343 MDX, 155 weighted, 180

G

Go To Section command, 76 Grand Total command, 132, 142 grand total data functions, 139 grand totals, inserting, 126, 139 graphic elements, 247 GREATER OR EQUAL, 273 GREATER THAN, 273 Grid command, 264 grid lines, 201 Grid Lines command, 78 grids, 251 group fact syntax, 242 group headers, report, 245 Group Items command, 197, 237 group label syntax, 242 grouping columns, 141 guides, design, 251

Η

headers, 84 page, in custom reports, 246 report group, 245 Headers and Footers command, 265 Help menu commands, xv Hide Column command, 132, 142, 237, 264 hide items, 219, 259 Hide Items command, 197, 237, 264 Hide Request Item command, 76 Hide Section command, 76 hiding and focusing on charted data, 218 hiding and focusing on reported data, 258 hiding column names, 244 hiding column totals, 244 hiding Request items, 100 hiding sections, 82 horizontal lines, 248 HTML export wizard, 71 restrictions and limitations, 71 HTML file formats, 56 comparison of, 56 Hyperion Consulting Services, xvii Hyperion Download Center accessing documents, xv Hyperion Education Services, xvii Hyperion product information, xvii Hyperion support, xvii Hyperion Technical Support, xviii

import data files, 48 SQL, 49 Import Data File commands, 75 Increase function, 177 INI file, 93 Initcap function, 297 Insert Field command, 264 Insert Graphic command, 264 Insert menu commands, 77 Insert Predefined Field command, 264 Insert Table command, 264 inserting additional tables in a custom report, 245 limit values, 250 page breaks, 247 page headers and footers, 246 report headers and footers, 246 text, 220 text in charts, 220 inserting sections, 77 Instr function, 297 Integer command, 87 Integer data type, 285 Interactive Reporting analysis and reporting, 26 quitting, 73 starting, 30 Interactive Reporting document (.bqy), 73 IS NULL, 273 items, sort, 216

J

JavaScript expressions, 61 Justify commands, 78

K

Keep Together command, 261 Keep With Next command, 261

L

label properties chart, 229 charts, 229 label reference, in sorts, 216, 342 labels, nested, 174 Las Saved field, 249 Last Printed field, 249 LastDay function, 294 layout aids, 241 legacy chart colors, 229, 234 legend, 201 legends, positioning and resizing, 220 Length function, 297 LESS OR EQUAL, 273 LESS THAN, 273 level rules, OLAPQuery member and, 150 LIKE, 273 limit values, in reports, 250 line charts, 212 lines, 248 local filters, Results section, 269 Local Results limitations, 117 processing order, 117 locally saved BQYs, 94 Locked command, 248 logical operators, 273 Lower function, 297 Ltrim function, 297

Μ

Main menu, 42 manipulating chart data, 213 manipulating table data, 135 margins, in Report section, 252 master data models, 25 Max function, 295 Maximum function, 139, 177, 214, 256 MaxL (Multidimensional Access Language), 26 MDSQL (Multidimensional Query Language), 26 MDX (Multidimensional Expression Language), 26 MDX functions, 155 measure filters, applying, 159 measure variable filters, 160 measures, 152 Median function, 296 member and level rules, OLAPQuery, 150 member, OLAP, 351 members, drilling down, 152 menu commands Chart, 237 Edit, 75 File, 75 Format, 78

Pivot, 197, 311 Report, 264 Results, 132 Table, 141 Tools, 93 menus and menu items, customizing, 92 meta topics, 25 metadata, definition, 23 methods for returning the day of the week, 303 Microsoft Office Excel, 58 Min function, 295 Minimum function, 139, 177, 214, 256 Mod function, 295 Mode function, 296 Modify Column command, 132, 141 Modify command, 197 Modify Computed Item command, 237 modifying pivot tables, 173 Month (for Add Date Groups) command, 87 MonthsBetween function, 294 moving between sections, 83 moving pivot table elements, 174 MovingMax, 332 multicolumned reports, 253 Multidimensional Access Language (MaxL), 26 multidimensional charts, 208 multidimensional databases components, 144 definition, 23 querying, 143 supported, 23 Multidimensional Expression Language (MDX), 26 Multidimensional Query Language (MDSQL), 26 multiple data sources, in reports, 259

N

nested labels, 174 nested sorts, 341 Next function, 296 NextDay function, 294 NM function, 294 Non-Null Average function, 178, 214, 256 Non-Null Count function, 178, 214, 256 NOT (with operator), 273 NOT EQUAL, 273 Null command, 87 Null Count function, 178, 214, 256 Number command, 78 number field definitions , 87 number formats, changing, 261 number formats, default, 87 Number of Pages field, 249

0

object names, 248 ODBC, stored procedures and, 118 **OLAP** queries adding totals to, 165 building, 145 OLAP, member, 351 OLAPQuery items, formatting, 166 OLAPQuery member and level rules, 150 OLAPQuery section, 26 accessing offline, 157 diagram, 144 showing as a chart, 165 working offline, 157 OLE DB for OLAP query language, 26 operators logical, 273 options program, 88 Options command, 94 OR, 269 Outliners, 43 ovals, 248

Ρ

Packed Real data type, 285 page breaks, inserting, 77, 247 Page Margins command, 264 Page Number field, 249 Page Setup command, 72, 75 Page X of Y field, 249 pages, setting up in reports, 39 password protect document, 46 Paste command, 75 Path Name field, 249 Percentile function, 296 Picture command, 264 pictures, 248 pie charts

analyzing data, 206 rotating, 207 pie slices, positioning, 207 Pivot, 172 Pivot Data Layout, 173 pivot elements, selecting, 173 Pivot menu commands, 197, 311 Pivot Options, 289 pivot tables calculating totals and subtotals, 175 charting, 173 how to use, 173 modifying, 173 selecting elements, 173 subtotals, 175 to 176 working with pivot items, 173 pivoting data, 173 planes, 201 plot area, 201 positioning pie slices, 207 Power function, 295 primary sort, 341 Print command, 73, 75 Print Preview command, 72, 75 Prior function, 296 process in OLAPQuery, 157 in Query, 100 Process Query command, 93 Process Results To Table command, 122 program options, selecting, 88 Properties command, 77, 237 properties, chart label, 229 properties, export, 50 to 51, 78 publish, 131

Q

```
queries
adding topics to, 31
appending, 116
building OLAP, 145
cancelling, 102
estimating size, 112
example, 31
multiple, 98
processing, 100 to 101, 157
```
sizing, 112 troubleshooting, 112 Query Limit field, 248 Query Log command, 76 query logs, exporting, 72 Query Options command, 122 Query Processing Order, 101 query properties, 119 Query section data functions , 114 diagram, 98 types of, 26 Query SQL field, 249 query, definition, 25, 113

R

Rank function scalar, 296 RankAsc function, 296 Real command, 87 Real data type, 285 rectangles, 248 Redo command, 75 reference items, in sorts, 216, 342 Refresh Data command, 197 relational databases definition, 23 querying, 97 remarks, displaying, 113 Remove command, 132, 141 Remove Selected Items command, 197, 237, 264 Remove Total command, 169 removing columns, 244 removing Request items, 99 removing Request line items, 99 Rename Section command, 76 renaming sections, 83 reordering columns, 245 reordering Request items, 99 reordering Request line items, 99 Replace function, 297 report components, 243 report element syntax, concatenating, 242 report expression syntax, 242 report group headers, 245 Report menu commands, 264

Report Name field, 249 report pages, 251 Report section computed fields, 249 computed items, 255 creating smart reports, 261 data functions, 256 diagram, 240 elements, 240 Expression syntax, 241 fields, 248 formatting items, 261 graphic elements, 248 limit values, 250 toolbar, 241 using multiple data sources, 259 Report Setup command, 265 reporting, 26 reports adding computed items, 255 creating, 39 data functions, 256 designing, 38 inserting page breaks, 247 multicolumned, 253 setting up, 39 smart, 261 Request items, hiding, 100 Restore Name command, 197, 237 Result Limit field, 248 results automatically adding columns, 128 enhancing, 125 exporting, 131 filtering, 125 save options, 131 saving, 46, 129 viewing, 32 working with, 123, 129 Results menu commands, 132 **Results** section automatically creating, 158 data functions, 126 diagram, 124 exporting, 55

Retrieve Dimensions command, 169 returning day of week, 303 ribbon charts, 211 rotating and elevating charts, 223 rotating pie charts, 207 Round function, 295 round rectangles, 248 Row command, 78 Row Numbers command, 78 rows, selecting, 129, 141 Rtrim function, 297 Rulers command, 264

S

Save As command, 75 Save command, 75 Save Connection command, 93 Save Options commands, 75 saving documents, 45 scalar functions, 294 examples, 307 scripts, running from menu items, 92 Section Boundaries command, 264 Section pane, 43 Section title bar, 43 Section Title Bar command, 76 Section/Catalog command, 76 sections adding, 82 deleting, 85 duplicating, 82 to 83 exporting, 55 hiding, 82 inserting, 77 moving between, 83 renaming, 83 viewing, 82 Select A Formatting Locale command, 87 Select All command, 76 selecting chart elements, 219 selecting columns, 244 selecting columns and rows, 141 selecting pivot table elements, 174 server filters in the Query section, 268 server versus local filter processing, 268 setting

bar chart properties, 231 chart label axis properties, 229 chart value axis properties, 230 general chart properties, 227 setting compound filters, 276 setting simple filters, 273 setting up page columns, 253 reports, 252 setting variable filters, 277 Show All Items command, 197, 237, 264 Show As Chart command, 169 Show Hidden Items command, 197, 237, 264 Show Remarks command, 122 show values, 271 showing column names, 244 showing column totals, 244 showing pie percentages, 207 showing positive and negative values, 207 Sign function, 295 simple sorts, 340 Sin function, 295 Sinh function, 295 slice, 201 slicers setting, 151 variable filters, 160 smart reports, 261 smart reports, creating, 39 Sort Ascending command, 132, 141, 237, 264 Sort command, 197, 264 Sort Descending command, 132, 141 sort items, 216, 342 sort line features, 216, 342 sort lines, 340 sort order, 341 sorting chart items, 216 charts, 175 pivot tables, 175 report items, 254 reports, 254 tables, 136 sorting data, 340 sorting, in OLAPQuery section, 342 sorting, in Query, Results, and Table sections, 341 sorts, nested, 341 specifying page margins, 252 page size, 252 Spotlighter command, 78 SOL definition, 25 exporting, 72 importing, 49 SOL files, 72 Sqrt function, 295 stacked bar charts, 210 staging data in tables, 134 standard query languages, 25 Status bar, 44 Status Bar command, 76 StdDev function, 297 StdDevp function, 297 stored procedures, 118 opening, 118 processing, 118 Stored Procedures command, 122 String data type, 285 Structured Query Language (SQL), 25 Style commands, 78 subqueries, building, 103 Substr function, 297 subtotals, 175 subtotals, calculating, 35 Sum function, 139, 177, 214, 256 Sum function, 296, 316 Suppress Duplicates command, 78 suppressing duplicate values, 244 surface values, 286, 289 surface values, in data functions, 179 syntax concatenating, 242 expression, 241 syntax, concatenating, 242 Sysdate function, 294

T

tab-delimited files, 48, 55 table column formatting options, 244 Table Data Layout, 134 table dimension syntax, 242

table fact syntax, 242 Table menu commands, 141 Table section computed items, 137 data functions, 139 diagram, 134 tables adding computed items, 137 filters, 135 in bqtbls5.ini file, 112 modifying pivot, 173 pivot, 173, 175 to 176 working with, 140 tables as a data staging area, 134 Tan function, 295 Tanh function, 295 technical support, xviii Teradata Version 3 OLAP functions, 298 text in charts, inserting, 220 text labels, 248 Text Wrap command, 78 text, formatting, 77 text, wrapping, 244 three-dimensional bar charts, 209 Time command, 87 Time datatype, 285 Time field, 249 Time Now field, 249 Timestamp command, 87 TimeStamp data type, 285 ToCha function, 294 ToDate function, 294 ToMonth function, 294 toolbar, in the Report section, 241 toolbars, 42 Toolbars command, 76 Tools menu commands, 93 topics, adding to query, 31 ToQtr function, 295 Total commands, 141 to 142 totals adding to OLAPQuery, 165 break, 127, 140 calculating, 35 calculating pivot totals, 175 cumulative, 176 to 177

inserting break totals, 140 inserting column totals, 140 inserting in Results section, 126 inserting in Tables section break, 139 custom, 139 grand, 139 surface, 177 surface, 177 ToYear function, 295 Translate function, 297 Trend Functions, 324 True Computed Item Totals, 178 Trunc function, 295 two-dimensional bar charts, 208 TXT files, 48, 55

U

underlying values, 286, 289 Undo command, 75 Ungroup Items command, 197, 237 Unhide Column command, 132, 142, 237, 264 Unhide Request Item command, 77 Unhide Section command, 76 Upper function, 297 using functions, 293 using surface values, 289

V

value reference, in sorts, 216, 342 values, 201, 213, 244 Var filter indicator, 269 Var function, 297 Var limit indicator, 269 Variable Filter command, 122 variable filters, in OLAPQuery, 159 Varp function, 297 VBA functions, 67 vertical lines, 248 View Job List command, 94 Visible command, 248

W

Web browser restrictions and limitations, 71 weighted functions, 180

workspace views, changing, 76 wrapping text, 244

X

XLS files, 55

Ζ

Zoom command, 76