
To ensure that you are using the documentation that corresponds to the software you are licensed to use, compare this version number with the software version shown in “About MicroStrategy...” in the Help menu of your software.

Document number: 09530921
Copyright © 2011 by MicroStrategy Incorporated. All rights reserved.

If you have not executed a written or electronic agreement with MicroStrategy or any authorized MicroStrategy distributor, the following terms apply:

This software and documentation are the proprietary and confidential information of MicroStrategy Incorporated and may not be provided to any other person. Copyright © 2001-2011 by MicroStrategy Incorporated. All rights reserved.

THIS SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" AND WITHOUT EXPRESS OR IMPLIED WARRANTY OF ANY KIND EITHER BY MICROSTRATEGY INCORPORATED OR ANYONE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DISTRIBUTION OF THE SOFTWARE OR DOCUMENTATION, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, GOOD TITLE AND NONINFRINGEMENT, QUALITY OR ACCURACY, THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE SOFTWARE AND DOCUMENTATION IS WITH YOU. SHOULD THE SOFTWARE OR DOCUMENTATION PROVE DEFECTIVE, YOU (AND NOT MICROSTRATEGY, INC. OR ANYONE ELSE WHO HAS BEEN INVOLVED WITH THE CREATION, PRODUCTION, OR DISTRIBUTION OF THE SOFTWARE OR DOCUMENTATION) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

In no event will MicroStrategy, Inc. or any other person involved with the creation, production, or distribution of the Software be liable to you on account of any claim for damage, including any lost profits, lost savings, or other special, incidental, consequential, or exemplary damages, including but not limited to any damages assessed against or paid by you to any third party, arising from the use, inability to use, quality, or performance of such Software and Documentation, or any other person involved in the creation, production, or distribution of the Software shall not be liable for any claim by you or any other party for damages arising from the use, inability to use, quality, or performance of such Software and Documentation, based upon contract, warranty, negligence, strict liability for the negligence of indemnity or contribution, the failure of any remedy to achieve its essential purpose, or otherwise. The entire liability of MicroStrategy, Inc. and your exclusive remedy shall not exceed, at the option of MicroStrategy, Inc., either a full refund of the price paid, or replacement of the Software. No oral or written information given out explains the liability of MicroStrategy, Inc. beyond that specified in the above limitation of liability. Some states do not allow the limitation or exclusion of liability for incidental or consequential damages, so the above limitation may not apply to you.

The information contained in this manual (the Documentation) and the Software are copyrighted and all rights are reserved by MicroStrategy, Inc. MicroStrategy reserves the right to make periodic modifications to the Software or the Documentation without obligation to notify any person or entity of such revision. Copying, duplicating, selling, or otherwise distributing any part of the Software or Documentation without prior written consent of an authorized representative of MicroStrategy, Inc. are prohibited. U.S. Government Restricted Rights. It is acknowledged that the Software and Documentation were developed at private expense, that no part is public domain, and that the Software and Documentation are Commercial Computer Software provided with RESTRICTED RIGHTS under Federal Acquisition Regulations and agency supplements to them. Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFAR 252.227-7013 et seq., or subparagraphs (c)(1) and (2) of the Commercial Computer Software—Restricted Rights at FAR 52.227-19, as applicable. Contractor is MicroStrategy, Inc., 1850 Towers Crescent Plaza, Vienna, VA 22182. Rights are reserved under copyright laws of the United States with respect to unpublished portions of the Software.


Patent Information
This product is patented. One or more of the following patents apply to the product sold herein: U.S. Patent Nos. 6,154,766, 6,173,310, 6,260,050, 6,283,051, 6,289,393, 6,279,033, 6,567,796, 6,587,547, 6,606,596, 6,658,093, 6,658,432, 6,662,195, 6,671,715, 6,691,100, 6,694,316, 6,697,808, 6,704,723, 6,741,980, 6,765,997, 6,768,788, 6,772,137, 6,788,788, 6,798,867, 6,801,910, 6,820,073, 6,829,334, 6,836,537, 6,850,603, 6,859,796, 6,873,695, 6,885,734, 6,940,953, 6,964,012, 6,977,992, 6,986,588, 6,996,569, 7,003,512, 7,010,518, 7,016,480, 7,020,251, 7,039,165, 7,082,422, 7,113,993, 7,127,403, 7,174,349, 7,181,417, 7,194,457, 7,197,461, 7,228,303, 7,260,577, 7,266,181, 7,272,212, 7,302,639, 7,324,942, 7,330,847, 7,334,580, 7,356,758, 7,359,486, 7,415,368, 7,240,502, 7,240,502, 7,240,502, 7,240,502, 7,363,760, 7,369,671, 7,359,046, 7,574,376, 7,517,201, 7,725,811, 7,801,967, 7,836,178, 7,861,161, 7,861,253, 7,881,443, 7,925,616, and 7,945,584. Other patent applications are pending.

Various MicroStrategy products contain the copyrighted technology of third parties. This product may contain one or more of the following copyrighted technologies:

- Graph Generation Engine
- XML parser
- Xalan XSLT processor
- ASIHTTPRequest library
- True fonts

Portions of Software, XML parser, and True fonts are provided by the Apache Software Foundation. All rights reserved.

MicroStrategy includes software developed by the Open Source Initiative for use in the Open Source Toolkit. (http://www.opensource.org/)
CONTENTS

Description of Guide ................................................................. ix

About this book ........................................................................ x
  Additional formats ................................................................. xi
  How to find business scenarios and examples ......................... xi
  What’s new in this guide ......................................................... xi
  Prerequisites ........................................................................ xii
  Who should use this guide ................................................... xii

Resources ....................................................................................... xiii
  Documentation ....................................................................... xiii
  Consulting ........................................................................... xxi
  International support .......................................................... xxii
  Technical Support ................................................................ xxii

Feedback ....................................................................................... xxvii

1. About MicroStrategy OLAP Services

  OLAP Services features ............................................................. 1
    Providing MOLAP analysis with Intelligent Cubes .................. 2
    Displaying data on the fly: dynamic aggregation ................. 3
    Creating metrics on-the-fly: derived metrics ....................... 3
    Defining attribute elements on-the-fly: derived elements ........ 4
    Filtering data on the fly: view filters and metric filters .......... 4
    Importing data as an Intelligent Cube .................................. 5

  The benefits of using OLAP Services ....................................... 5

  Privileges required for Desktop and Web users ....................... 7

  Report types: Standard and OLAP reports ............................ 8

  Standard OLAP analysis features .......................................... 10
    Aliasing ............................................................................. 10
Creating derived elements........................................................... 81
   Quickly creating groups, calculations, and sorts.................... 82
   Using the Derived Elements Editor ........................................ 97
   Creating and using stand-alone derived elements............... 117

Defining derived element functionality and formatting .......... 122
   Applying derived element values to subtotals...................... 122
   Displaying derived elements or their attribute elements ...... 125
   Displaying derived elements and their attribute elements simultaneously ........................................................ 127
   Formatting derived elements .............................................. 130

Interaction with other reporting features ............................... 134
   View filters ....................................................................... 135
   Derived metrics .................................................................. 135
   Page-by ............................................................................ 136
   Thresholds ....................................................................... 138
   Drilling ........................................................................... 139
   Subtotals .......................................................................... 141
   Smart metrics .................................................................... 142

5. Dynamic Aggregation
   Displaying Data On The Fly
   Using dynamic aggregation ................................................... 148
   Functions used in dynamic aggregation ................................. 149
      Metrics with default dynamic aggregation functions .......... 150
      Metrics that are not dynamically aggregated by default..... 152
      Changing the default dynamic aggregation function ......... 161
   View filter effect on dynamic aggregation .............................. 163

6. Derived Metrics
   Creating Metrics On-The-Fly
   Creating a derived metric ..................................................... 167
      Best practices .................................................................. 167
      Creating a derived metric using the Input Metric Formula dialog box ................................................................. 169
      Using rank and percent-to-total metric analysis ................. 173
   Editing derived metrics ....................................................... 180
   Formatting derived metrics .................................................. 181
   Deleting derived metrics ..................................................... 181
   View filter effects on derived metrics ................................... 182
   Derived element effects on derived metrics .......................... 183
Description of Guide

This guide provides information on MicroStrategy OLAP Services, which is an extension of MicroStrategy Intelligence Server.

This book assumes that you are familiar with the fundamental concepts about the MicroStrategy product and that you have experience with the various features the product offers. In addition, you should have basic knowledge about Multidimensional Online Analytical Processing (MOLAP) and Relational OLAP (ROLAP) analyses.

This book is divided into the following chapters:

- **Chapter 1, About MicroStrategy OLAP Services**: Includes introductory information on the features available with MicroStrategy OLAP Services.

- **Chapter 2, Sharing Sets of Data Among Reports: Intelligent Cubes**: Includes information on Intelligent Cubes, a key component to implementing MicroStrategy OLAP Services.

- **Chapter 3, Reporting on Intelligent Cubes**: Includes information on how you can report and analyze data from Intelligent Cubes.
• **Chapter 4, Derived Elements**: Includes information on derived elements, which provide a new view of report data for analysis and formatting purposes.

• **Chapter 5, Dynamic Aggregation**: Includes information on dynamic aggregation, which allows you to change the level of report aggregation on-the-fly, while you are reviewing the report results.

• **Chapter 6, Derived Metrics**: Includes information on derived metrics, which can perform column math, that is, calculations on metrics included in the report definition, without regenerating or re-executing SQL against the data warehouse.

• **Chapter 7, View Filters**: Includes information on view filters, which restrict the amount of data displayed on the report, providing you with a different view of the data.

• **Chapter 8, Dynamic Sourcing**: Includes information on dynamic sourcing, which extends the accessibility of Intelligent Cubes by allowing regular reports to automatically access published Intelligent Cubes that can satisfy the data requirements of the report.

---

### About this book

This book is divided into chapters and reference appendices. Each chapter begins with a brief overview of the chapter’s content.

The following sections provide the location of examples, list prerequisites for using this book, and describe the user roles the information in this book was designed for.

Dates in the MicroStrategy Tutorial project are updated to reflect the current year. The sample documents and images in this guide, as well as the procedures, were created with dates that may no longer be available in the Tutorial project. Replace them with the first year of data in your Tutorial project.
Additional formats

This book is also available as an electronic publication in the Apple iBookstore, and can be read on an iPhone or iPad with the iBooks app installed. To download this book, search for the book’s title in the iBookstore search bar, or scan the QR code below using your device’s camera.

How to find business scenarios and examples

Within this guide, many of the concepts discussed are accompanied by business scenarios or other descriptive examples. Many of the examples use the MicroStrategy Tutorial, which is MicroStrategy’s sample warehouse, metadata, and project. Information about the MicroStrategy Tutorial can be found in the MicroStrategy Basic Reporting Guide.

Other examples in this book use the Analytics Modules, which include a set of sample reports, each from a different business area. Sample reports present data for analysis in such business areas as financial reporting, human resources, and customer analysis.

Detailed examples of advanced reporting functionality can be found in the MicroStrategy Advanced Reporting Guide.

What’s new in this guide

MicroStrategy 9.2

- Instructions for the new Incremental Refresh feature, which allows you to update the data in Intelligent Cubes without re-executing the entire Intelligent Cube. For information, see Updating Intelligent Cubes without re-processing: Incremental Refresh, page 45.
• Updated the conditions under which transformation metrics affect whether reports are eligible for dynamic sourcing. For information, see *Features that can be used with dynamic sourcing, page 240.*

• Added conditions for dynamic sourcing to be supported for MDX reports. For information, see *Features that prevent the use of dynamic sourcing, page 241.*

**Prerequisites**

Before working with this document, you should be familiar with:

• MicroStrategy objects, including attributes, metrics, and filters
• Project creation and configuration
• Report creation and manipulation
• MOLAP and ROLAP analyses
• Structured Query Language (SQL)

**Who should use this guide**

This document is designed for:

• Report Designers who will be using the OLAP Services to create advanced reports and reporting objects such as metrics, filters, and so on
• Analysts who will be performing advanced report manipulation
Resources

Documentation

MicroStrategy provides both manuals and online help; these two information sources provide different types of information, as described below:

- **Manuals**: In general, MicroStrategy manuals provide:
  - Introductory information and concepts
  - Examples and images
  - Checklists and high-level procedures to get started

  The steps to access the manuals are described in *Accessing manuals and other documentation sources, page xix*. Most of these manuals are also available printed in a bound, soft cover format. To purchase printed manuals, contact your MicroStrategy account executive or email documentationfeedback@microstrategy.com.

- **Help**: In general, MicroStrategy help provides:
  - Detailed steps to perform procedures
  - Descriptions of each option on every software screen

  Due to translation time, manuals in languages other than English may contain information that is one or more releases behind. You can see the version number on the title page of each manual. For the most up-to-date translations, refer to the MicroStrategy Knowledge Base.

MicroStrategy overview and evaluation

- **Introduction to MicroStrategy: Evaluation Guide**

  Instructions for installing, configuring, and using the MicroStrategy Evaluation Edition of the software. This guide also includes a detailed, step-by-step evaluation process of MicroStrategy features, where you perform reporting with the MicroStrategy Tutorial project and its sample business data.
• **MicroStrategy Evaluation Edition Quick Start Guide**
  Overview of the installation and evaluation process, and additional resources.

• **Evaluate MicroStrategy for Linux Guide: In a Windows or Linux Environment with the MicroStrategy Evaluation Edition Virtual Appliance**
  Evaluate MicroStrategy for Linux, in a Microsoft Windows or Linux environment, with the MicroStrategy Evaluation Edition Virtual Appliance. This guide provides all details to download, activate, and evaluate MicroStrategy software running in a Linux environment.

• **MicroStrategy Reporting Suite: Quick Start Guide**
  Evaluate MicroStrategy as a departmental solution. Provides detailed information to download, install, configure, and use the MicroStrategy Reporting Suite.

• **MicroStrategy Mobile Suite: Quick Start Guide**
  Evaluate MicroStrategy Mobile as a departmental solution. Provides detailed information to download, install, configure, and use the MicroStrategy Mobile Suite.

### Manuals for query, reporting, and analysis

• **MicroStrategy Installation and Configuration Guide**
  Information to install and configure MicroStrategy products on Windows, UNIX, Linux, and HP platforms, as well as basic maintenance guidelines.

• **MicroStrategy Upgrade Guide**
  Instructions to upgrade existing MicroStrategy products.

• **MicroStrategy Project Design Guide**
  Information to create and modify MicroStrategy projects, and understand facts, attributes, hierarchies, transformations, advanced schemas, and project optimization.

• **MicroStrategy Basic Reporting Guide**
  Instructions to get started with MicroStrategy Desktop and MicroStrategy Web, and how to analyze data in a report. Includes the basics for creating reports, metrics, filters, and prompts.
• MicroStrategy Advanced Reporting Guide: Enhancing Your Business Intelligence Application

Instructions for advanced topics in the MicroStrategy system, building on information in the Basic Reporting Guide. Topics include reports, Freeform SQL reports, Query Builder reports, filters, metrics, Data Mining Services, custom groups, consolidations, and prompts.


Instructions for a business analyst to execute and analyze a document in MicroStrategy Desktop and MicroStrategy Web, building on basic concepts about projects and reports presented in the MicroStrategy Basic Reporting Guide.

• MicroStrategy Report Services Document Creation Guide: Creating Boardroom Quality Documents


• MicroStrategy OLAP Services Guide

Information on MicroStrategy OLAP Services, which is an extension of MicroStrategy Intelligence Server. OLAP Services features include Intelligent Cubes, derived metrics, derived elements, dynamic aggregation, view filters, and dynamic sourcing.

• MicroStrategy Office User Guide

Instructions for using MicroStrategy Office to work with MicroStrategy reports and documents in Microsoft® Excel, PowerPoint, Word, and Outlook, to analyze, format, and distribute business data.

• MicroStrategy Mobile Analysis Guide: Analyzing Data with MicroStrategy Mobile

Information and instructions for using MicroStrategy Mobile to view and analyze data, and perform other business tasks with MicroStrategy reports and documents on a mobile device.

• MicroStrategy Mobile Design and Administration Guide: A Platform for Mobile Intelligence

Information and instructions to install and configure MicroStrategy Mobile, as well as instructions for a designer working in MicroStrategy Desktop or MicroStrategy Web to create effective reports and documents for use with MicroStrategy Mobile.
• **MicroStrategy System Administration Guide: Tuning, Monitoring, and Troubleshooting your MicroStrategy Business Intelligence System**

  Concepts and high-level steps to implement, deploy, maintain, tune, and troubleshoot a MicroStrategy business intelligence system.

• **MicroStrategy Supplemental Reference for System Administration: VLDB Properties, Internationalization, User Privileges, and other Supplemental Information for Administrators**

  Information and instructions for MicroStrategy administrative tasks such as configuring VLDB properties and defining data and metadata internationalization, and reference material for other administrative tasks.

• **MicroStrategy Functions Reference**

  Function syntax and formula components; instructions to use functions in metrics, filters, attribute forms; examples of functions in business scenarios.

• **MicroStrategy MDX Cube Reporting Guide**

  Information to integrate MicroStrategy with MDX cube sources. You can integrate data from MDX cube sources into your MicroStrategy projects and applications.

### Manuals for Analytics Modules

• **Analytics Modules Installation and Porting Guide**

• **Customer Analysis Module Reference**

• **Sales Force Analysis Module Reference**

• **Financial Reporting Analysis Module Reference**

• **Sales and Distribution Analysis Module Reference**

• **Human Resources Analysis Module Reference**

### Manuals for Narrowcast Services products

• **MicroStrategy Narrowcast Server Getting Started Guide**

  Instructions to work with the tutorial to learn Narrowcast Server interfaces and features.
- **MicroStrategy Narrowcast Server Installation and Configuration Guide**
  Information to install and configure Narrowcast Server.

- **MicroStrategy Narrowcast Server Application Designer Guide**
  Fundamentals of designing Narrowcast Server applications.

- **MicroStrategy Narrowcast Server System Administrator Guide**
  Concepts and high-level steps to implement, maintain, tune, and troubleshoot Narrowcast Server.

- **MicroStrategy Narrowcast Server Upgrade Guide**
  Instructions to upgrade an existing Narrowcast Server.

**Software Development Kits**

- **MicroStrategy Developer Library (MSDL)**
  Information to understand the MicroStrategy SDK, including details about architecture, object models, customization scenarios, code samples, and so on.

- **MicroStrategy Web SDK**
  The Web SDK is available in the MicroStrategy Developer Library, which is sold as part of the MicroStrategy SDK.

- **Narrowcast Server SDK Guide**
  Instructions to customize Narrowcast Server functionality, integrate Narrowcast Server with other systems, and embed Narrowcast Server functionality within other applications. Documents the Narrowcast Server Delivery Engine and Subscription Portal APIs, and the Narrowcast Server SPI.

**Documentation for MicroStrategy Portlets**

- **Enterprise Portal Integration Help**
  Information to help you implement and deploy MicroStrategy BI within your enterprise portal, including instructions for installing and configuring out-of-the-box MicroStrategy Portlets for several major enterprise portal servers.
This resource can be accessed using the MicroStrategy Product Manuals page, as described in Accessing manuals and other documentation sources, page xix.

Documentation for MicroStrategy GIS Connectors

• GIS Integration Help

Information to help you integrate MicroStrategy with Geospatial Information Systems (GIS), including specific examples for integrating with ESRI mapping services.

This resource can be accessed using the MicroStrategy Product Manuals page, as described in Accessing manuals and other documentation sources, page xix.

Help

Each MicroStrategy product includes an integrated help system to complement the various interfaces of the product as well as the tasks that can be accomplished using the product.

Some of the MicroStrategy help systems require a web browser to be viewed. For supported web browsers, see the MicroStrategy Readme.

MicroStrategy provides several ways to access help:

• Help button: Use the Help button or ? (question mark) icon on most software windows to see help for that window.

• Help menu: From the Help menu or link at the top of any screen, select MicroStrategy Help to see the table of contents, the Search field, and the index for the help system.

• F1 key: Press F1 to see context-sensitive help that describes each option in the software window you are currently viewing.

For MicroStrategy Web, MicroStrategy Web Administrator, and MicroStrategy Mobile Server, pressing the F1 key opens the context-sensitive help for the web browser you are using to access these MicroStrategy interfaces. Use the Help menu or ? (question mark) icon to access help for these MicroStrategy interfaces.
Accessing manuals and other documentation sources

The manuals are available from your MicroStrategy disk or the machine where MicroStrategy was installed.

Adobe Acrobat Reader is required to view these manuals. If you do not have Acrobat Reader installed on your computer, you can download it from http://get.adobe.com/reader/.

The best place for all users to begin is with the MicroStrategy Basic Reporting Guide.

To access the installed manuals and other documentation sources, see the following procedures:

- To access installed manuals and other documentation sources on Windows, page xix
- To access installed manuals and other documentation sources on UNIX and Linux, page xix

To access installed manuals and other documentation sources on Windows

1 From the Windows Start menu, choose Programs (or All Programs), MicroStrategy, then Product Manuals. A page opens in your browser showing a list of available manuals in PDF format and other documentation sources.

2 Click the link for the desired manual or other documentation source.

3 If you click the link for the Narrowcast Services SDK Guide, a File Download dialog box opens. This documentation resource must be downloaded. Select Open this file from its current location, and click OK.

If bookmarks are not visible on the left side of an Acrobat (PDF) manual, from the View menu click Bookmarks and Page. This step varies slightly depending on your version of Adobe Acrobat Reader.

To access installed manuals and other documentation sources on UNIX and Linux

1 Within your UNIX or Linux machine, navigate to the directory where you installed MicroStrategy. The default location is /opt/MicroStrategy,
or $HOME/MicroStrategy/install if you do not have write access to /opt/MicroStrategy.

2 From the MicroStrategy installation directory, open the Documentation folder.

3 Open the Product_Manuals.htm file in a web browser. A page opens in your browser showing a list of available manuals in PDF format and other documentation sources.

4 Click the link for the desired manual or other documentation source.

5 If you click the link for the Narrowcast Services SDK Guide, a File Download dialog box opens. This documentation resource must be downloaded. Select Open this file from its current location, and click OK.

If bookmarks are not visible on the left side of an Acrobat (PDF) manual, from the View menu click Bookmarks and Page. This step varies slightly depending on your version of Adobe Acrobat Reader.

**Documentation standards**

MicroStrategy online help and PDF manuals (available both online and in printed format) use standards to help you identify certain types of content. The following table lists these standards.

These standards may differ depending on the language of this manual; some languages have rules that supersede the table below.

<table>
<thead>
<tr>
<th>Type</th>
<th>Indicates</th>
</tr>
</thead>
</table>
| **bold** | • Button names, check boxes, options, lists, and menus that are the focus of actions or part of a list of such GUI elements and their definitions  
| | • Text to be entered by the user  
| | Example: Click Select Warehouse.  
| | Example: Type cmdmgr -f scriptfile.scp and press Enter.  |
| **italic** | • New terms defined within the text and in the glossary  
| | • Names of other product manuals  
| | • When part of a command syntax, indicates variable information to be replaced by the user  
| | Example: The aggregation level is the level of calculation for the metric.  
| | Example: Type copy c:\filename d:\foldername\filename  |
MicroStrategy Education Services provides a comprehensive curriculum and highly skilled education consultants. Many customers and partners from over 800 different organizations have benefited from MicroStrategy instruction.

Courses that can help you prepare for using this manual or that address some of the information in this manual include:

- MicroStrategy Desktop: Reporting Essentials
- MicroStrategy Desktop: Advanced Reporting

For the most up-to-date and detailed description of education offerings and course curricula, visit http://www.microstrategy.com/Education.

MicroStrategy Consulting Services provides proven methods for delivering leading-edge technology solutions. Offerings include complex security architecture designs, performance and tuning, project and testing strategies and recommendations, strategic planning, and more. For a detailed description of consulting offerings, visit http://www.microstrategy.com/Consulting.
International support

MicroStrategy supports several locales. Support for a locale typically includes native database and operating system support, support for date formats, numeric formats, currency symbols, and availability of translated interfaces and certain documentation.

MicroStrategy is certified in homogeneous configurations (where all the components lie in the same locale) in the following languages—English (US), French, German, Italian, Japanese, Korean, Portuguese (Brazilian), Spanish, Chinese (Simplified), Chinese (Traditional), Danish, and Swedish. A translated user interface is available in each of the above languages. For information on specific languages supported by individual MicroStrategy system components, see the MicroStrategy readme.

MicroStrategy also provides limited support for heterogeneous configurations (where some of the components may lie in different locales). Please contact MicroStrategy Technical Support for more details.

A translated user interface is available in each of the above languages.

Technical Support

If you have questions about a specific MicroStrategy product, you should:

1. Consult the product guides, Help, and readme files. Locations to access each are described above.

2. Consult the MicroStrategy Knowledge Base online at [https://resource.microstrategy.com/support](https://resource.microstrategy.com/support).

   A technical administrator in your organization may be able to help you resolve your issues immediately.

3. If the resources listed in the steps above do not provide a solution, contact MicroStrategy Technical Support directly. To ensure the most productive relationship with MicroStrategy Technical Support, review the Policies and Procedures document in your language, posted at [http://www.microstrategy.com/Support/Policies](http://www.microstrategy.com/Support/Policies). Refer to the terms of your purchase agreement to determine the type of support available to you.

   MicroStrategy Technical Support can be contacted by your company’s Support Liaison. A Support Liaison is a person whom your company has
designated as a point-of-contact with MicroStrategy’s support personnel. All customer inquiries and case communications must come through these named individuals. Your company may designate two employees to serve as their Support Liaisons, and can request to change their Support Liaisons two times per year with prior written notice to MicroStrategy Technical Support.

It is recommended that you designate Support Liaisons who have MicroStrategy Administrator privileges. This can eliminate security conflicts and improve case resolution time. When troubleshooting and researching issues, MicroStrategy Technical Support personnel may make recommendations that require administrative privileges within MicroStrategy, or that assume that the designated Support Liaison has a security level that permits them to fully manipulate the MicroStrategy projects and has access to potentially sensitive project data such as security filter definitions.

### Ensure issues are resolved quickly

Before logging a case with MicroStrategy Technical Support, the Support Liaison may follow the steps below to ensure that issues are resolved quickly:

1. Verify that the issue is with MicroStrategy software and not a third party software.


3. Attempt to reproduce the issue and determine whether it occurs consistently.

4. Minimize the complexity of the system or project object definition to isolate the cause.

5. Determine whether the issue occurs on a local machine or on multiple machines in the customer environment.

6. Discuss the issue with other users by posting a question about the issue on the MicroStrategy Customer Forum at [https://resource.microstrategy.com/forum/](https://resource.microstrategy.com/forum/).

The following table shows where, when, and how to contact MicroStrategy Technical Support. If your Support Liaison is unable to reach MicroStrategy Technical Support by phone during the hours of operation, they can leave a voicemail message, send email or fax, or log a case using the Online Support.
The individual Technical Support Centers are closed on certain public holidays.

<table>
<thead>
<tr>
<th>Region</th>
<th>Email</th>
<th>Web</th>
<th>Phone</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>North America</td>
<td><a href="mailto:support@microstrategy.com">support@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td>(703) 842–8709</td>
<td>9:00 A.M.–7:00 P.M. Eastern Time, Monday–Friday except holidays</td>
</tr>
<tr>
<td>EMEA: Europe</td>
<td><a href="mailto:eurosupp@microstrategy.com">eurosupp@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td>+44 (0) 208 711 2525</td>
<td>United Kingdom: 9:00 A.M.–6:00 P.M. GMT, Monday-Friday except holidays</td>
</tr>
<tr>
<td>The Middle East</td>
<td></td>
<td></td>
<td></td>
<td>EMEA (except UK): 9:00 A.M.–6:00 P.M. CET, Monday-Friday except holidays</td>
</tr>
<tr>
<td>Africa</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asia Pacific</td>
<td><a href="mailto:apsupport@microstrategy.com">apsupport@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td></td>
<td>Japan and Korea: 9:00 A.M.–6:00 P.M. JST (Tokyo), Monday-Friday except holidays</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Asia Pacific (except Japan and Korea): 7 A.M.–6 P.M. (Singapore) Monday-Friday except holidays</td>
</tr>
<tr>
<td>Latin America</td>
<td><a href="mailto:latamsupport@microstrategy.com">latamsupport@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td></td>
<td>Latin America (except Brazil): 9:00 A.M.–7:00 P.M. (Buenos Aires), Monday-Friday except holidays</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Brazil: 9 A.M. - 6 P.M. (São Paulo), Monday–Friday except holidays</td>
</tr>
</tbody>
</table>
Support Liaisons should contact the Technical Support Center from which they obtained their MicroStrategy software licenses or the Technical Support Center to which they have been designated.

**Required information when calling**

When contacting MicroStrategy Technical Support, please provide the following information:

- **Personal information:**
  - Name (first and last)
  - Company and customer site (if different from company)
  - Contact information (phone and fax numbers, e-mail addresses)

- **Case details:**
  - Configuration information, including MicroStrategy software product(s) and versions
  - Full description of the case including symptoms, error messages(s), and steps taken to troubleshoot the case thus far

- **Business/system impact**

If this is the Support Liaison’s first call, they should also be prepared to provide the following:

- Street address
- Phone number
- Fax number
- Email address

To help the Technical Support representative resolve the problem promptly and effectively, be prepared to provide the following additional information:

- **Case number:** Please keep a record of the number assigned to each case logged with MicroStrategy Technical Support, and be ready to provide it when inquiring about an existing case

- **Software version and product registration numbers of the MicroStrategy software products you are using**
• Case description:
  □ What causes the condition to occur?
  □ Does the condition occur sporadically or each time a certain action is performed?
  □ Does the condition occur on all machines or just on one?
  □ When did the condition first occur?
  □ What events took place immediately prior to the first occurrence of the condition (for example, a major database load, a database move, or a software upgrade)?
  □ If there was an error message, what was its exact wording?
  □ What steps have you taken to isolate and resolve the issue? What were the results?

• System configuration (the information needed depends on the nature of the problem; not all items listed below may be necessary):
  □ Computer hardware specifications (processor speed, RAM, disk space, and so on)
  □ Network protocol used
  □ ODBC driver manufacturer and version
  □ Database gateway software version
  □ (For MicroStrategy Web-related problems) browser manufacturer and version
  □ (For MicroStrategy Web-related problems) Web server manufacturer and version

If the issue requires additional investigation or testing, the Support Liaison and the MicroStrategy Technical Support representative should agree on certain action items to be performed. The Support Liaison should perform any agreed-upon actions before contacting MicroStrategy Technical Support again regarding the issue. If the Technical Support representative is responsible for an action item, the Support Liaison may call MicroStrategy Technical Support at any time to inquire about the status of the issue.
Feedback

Please send any comments or suggestions about user documentation for MicroStrategy products to:

doctorationfeedback@microstrategy.com

Send suggestions for product enhancements to:

support@microstrategy.com

When you provide feedback to us, please include the name and version of the products you are currently using. Your feedback is important to us as we prepare for future releases.
Introduction

This chapter is an overview of MicroStrategy OLAP Services, which extends the capability and functionality of MicroStrategy Intelligence Server.

This chapter covers the following information on OLAP Services:

- **OLAP Services features, page 1**
- **The benefits of using OLAP Services, page 5**
- **Privileges required for Desktop and Web users, page 7**
- **Report types: Standard and OLAP reports, page 8**
- **Standard OLAP analysis features, page 10**

OLAP Services features

As an extension of MicroStrategy Intelligence Server, *OLAP Services* uses Intelligent Cube Technology—an in-memory version of report data that can
be manipulated by the MicroStrategy Analytical Engine. MicroStrategy Desktop, Web, and Office users can slice and dice data in reports within Intelligent Cubes without having to re-execute SQL against the data warehouse. *Chapter 2, Sharing Sets of Data Among Reports: Intelligent Cubes*, describes the role of OLAP Services in the overall MicroStrategy architecture and how you can take advantage of Intelligent Cubes to achieve the best efficiency and performance in your application.

As a MicroStrategy user, you may already take advantage of the many standard OLAP features that MicroStrategy provides out of the box, such as page-by, pivoting, sorting, subtotals, and so on.

With an OLAP Services license, you can perform additional OLAP analysis, using the following features:

- *Displaying data on the fly: dynamic aggregation, page 3*
- *Creating metrics on-the-fly: derived metrics, page 3*
- *Defining attribute elements on-the-fly: derived elements, page 4*
- *Filtering data on the fly: view filters and metric filters, page 4*
- *Importing data as an Intelligent Cube, page 5*

**Providing MOLAP analysis with Intelligent Cubes**

The main feature supporting MicroStrategy OLAP Services is MicroStrategy Intelligent Cube Technology™, which allows you to create multi-dimensional cubes (sets of data) that are stored within MicroStrategy Intelligence Server. On MicroStrategy’s business intelligence platform, you have two unique methods to implement Intelligent Cube Technology:

- **Personal Intelligent Cubes**: You can begin by creating reports in MicroStrategy as usual, and then analyze your reports with OLAP Services features such as view filters, derived metrics, and derived elements. These features are processed on the in-memory copy of data, known as a personal Intelligent Cube, rather than processed on the data warehouse.

- **Intelligent Cubes**: Rather than returning data from the data warehouse for a single report, you can return sets of data, called Intelligent Cubes, from your data warehouse, and save them directly to Intelligence Server’s memory. Intelligent Cubes can be shared as a single in-memory copy, to be used by many different reports created by multiple users.
For detailed information on Intelligent Cubes, refer to *Chapter 2, Sharing Sets of Data Among Reports: Intelligent Cubes*.

**Displaying data on the fly: dynamic aggregation**

When you create a non-OLAP Services report in MicroStrategy, data is retrieved from the data warehouse and aggregated to the logical level determined by the objects on the report.

Dynamic aggregation allows metric values to be aggregated at different levels, depending on the attributes included on the report, without having to re-execute it against the data warehouse. Dynamic aggregation occurs when the attributes included on the report layout changes—for example, when you move an attribute or attribute form off of the report layout to the Report Objects pane, or when you move an attribute or attribute form from the Report Objects pane back onto the report layout. As the attributes included on the report layout change, metric values are dynamically aggregated to the new level of the report.

For example, a report with revenue data and attributes for year and region displays revenue data for each region during each year. You can move the attribute for year from the report layout to the Report Objects pane to display a region’s revenue data for all years rather than display each yearly revenue total separately.

Metrics can also be moved between the report layout and the Report Objects pane, but this does not affect the level of aggregation for the report.

By default, the Analytical Engine selects the best aggregation function to use for each metric. However, you can also specify the function for each metric. You can use any of the standard predefined subtotal functions or define your own functions using user-defined subtotals.

For more detailed information on dynamic aggregation, refer to *Chapter 5, Dynamic Aggregation*.

**Creating metrics on-the-fly: derived metrics**

Derived metrics perform calculations on-the-fly with the data available in a report. They are an easy way to present data already available on the report in different ways, providing further analysis of data. You can use derived
metrics to quickly perform on-the-fly analyses such as margins, contributions, and differences between metrics included on the report.

These metrics are created based on existing metrics in the report. Since derived metrics are evaluated in-memory, their computation does not require any SQL to execute in the database.

Since derived metrics are created within a report, they can only be used for the report in which they are created. They cannot be saved as individual objects in the project, and therefore cannot be applied to other reports in the project.

For more detailed information on derived metrics, refer to *Chapter 6, Derived Metrics*.

**Defining attribute elements on-the-fly: derived elements**

A derived element is a grouping of attribute elements on a report. These groups provide a new view of report data for analysis and formatting purposes. For example, you can group data for the months of December, January, and February into a single element that combines and displays the data for the entire winter season.

Rather than having to define consolidations or custom groups, you can use derived elements to create these groups on-the-fly while viewing a report. Derived elements are evaluated on the report dataset without regenerating or re-executing SQL.

Derived elements are defined by using a list, filter, or calculation to combine attribute element data.

For detailed information on derived elements, refer to *Chapter 4, Derived Elements*, in this guide.

**Filtering data on the fly: view filters and metric filters**

A *view filter* dynamically restricts the data being displayed on the report without re-executing the report against the warehouse. This is different from a report filter, which restricts how much data is retrieved from the data warehouse. View filters improve the response time of reports, and decrease database load.
It is important to note that you can use a report filter and view filter on the same report. The report filter returns a set of data for the report, which the view filter then restricts further. You should therefore avoid defining contradictory filtering criteria in both, otherwise you may encounter situations where no data is displayed.

For detailed information on view filters, refer to Chapter 7, View Filters.

Importing data as an Intelligent Cube

You can use the Import Data feature in MicroStrategy Web to import data from sources such as Microsoft Excel spreadsheets, comma-separated text files, and external databases into your project. This data is stored as Intelligent Cubes.

Some uses for this data include:

- Quickly integrating data into MicroStrategy as part of a proof-of-concept
- Importing and reporting on personalized data from various data sources
- Immediately building reports and documents without having to “model” the data source as part of the entire project
- Modifying the data in your data source, then republishing the Intelligent Cube to quickly update the data in your reports and documents

For detailed information on the Import Data feature, see the MicroStrategy Web Help.

The benefits of using OLAP Services

OLAP Services combines the benefits of both MOLAP and ROLAP analyses.

Generally speaking, MOLAP tools place data in multi-dimensional cubes and perform consolidations in advance, allowing you to run queries fast. ROLAP tools allow you to use complex SQL queries against relational tables to obtain multi-dimensional views of data on-the-fly.
Each type of analysis has its own advantages and disadvantages. However, OLAP Services provides MOLAP and ROLAP analysis on the same report, which offers many distinct benefits, summarized below:

- **Analyze reports at the “speed of thought”, and manipulate them in real time**

  Using OLAP Services, you can get fast response times for reports that use data directly from in-memory Intelligent Cubes, instead of from the data warehouse. You can create and analyze new reports in real time through interactive OLAP Services manipulations.

- **Share Intelligent Cube data securely**

  MicroStrategy’s centralized metadata and Intelligence Server architecture allows Intelligent Cube data to be shared in a secure fashion.

- **Create and synchronize personal Intelligent Cubes automatically**

  Personal Intelligent Cubes are created, analyzed, and populated in the metadata for users across the enterprise to take advantage of OLAP Services. Automatic data synchronization ensures that the data in Intelligent Cubes is always up to date.

- **Schedule Intelligent Cube execution and maintenance**

  To reduce stress on the Intelligence Server, you can schedule when Intelligent Cubes are executed. This allows you to take advantage of Intelligence Server down time to execute Intelligent Cubes without affecting performance for your user community.

  You can also schedule when Intelligent Cubes are re-executed to synchronize their data with changes to the data in your data warehouse.

- **Drill from summary data to transaction-level details**

  You can drill from predefined reports to conduct advanced analysis and take full advantage of the Intelligent Cube feature. Drilling is allowed within an Intelligent Cube for quick-response MOLAP analysis. Drilling can also be enabled outside of an Intelligent Cube for full ROLAP analysis.

- **Use MicroStrategy Desktop, Office, or Web**

  Using OLAP Services, you can perform the same multidimensional analysis whether you use MicroStrategy Desktop, Office, or Web.
• **Apply security restrictions on users and objects**

Reporting with OLAP Services and Intelligent Cubes adhere to the same standards of data access security as the rest of your MicroStrategy project.

• **Increase user self-service and productivity**

Since accessing Intelligent Cubes for OLAP analysis does not require runtime processing on the data warehouse and can use schedules to reduce IT management, users have increased flexibility to create and modify their own reports to suit their unique work environment.

### Privileges required for Desktop and Web users

If you have purchased an OLAP Services license, you can assign related privileges to different user groups on Desktop or Web to perform specific tasks. The following list summarizes which privileges can be assigned for which user groups:

• **Common privileges**: The predefined MicroStrategy Web Reporter and Desktop Analyst user groups are assigned the set of common privileges by default. Within the set of common privileges, the following privilege is specific to OLAP Services:
  - **Drill within Intelligent Cube**: This privilege allows you to drill within an Intelligent Cube, which means that the drill can be resolved through OLAP Services and therefore does not need to generate and execute SQL against the warehouse.

  • **Use Dynamic Sourcing**: This privilege lets you to use Dynamic Sourcing, which allows non-OLAP Services reports use Intelligent Cubes that satisfy their data requirements.

  • **Use OLAP Services**: This privilege lets you create and execute reports and documents that use OLAP Services.

• **Web Analyst**: The predefined MicroStrategy Web Analyst group is assigned the set of Web Analyst privileges by default. This group also inherits all of the privileges assigned to the Web Reporter group. Within the set of Web Analyst and Reporter privileges, The following privileges are specific to OLAP Services users:

  • **Web add/remove units to/from grid in document in View mode**: This privilege allows you to add to or remove report objects from an existing grid report in a Report Services document.
- **Web create derived metrics**: This privilege allows you to create new calculations based on other metrics already on a base report.

- **Web number formatting**: This privilege allows you to change number formats for all metrics on grid reports.

- **Web use Report Objects window**: This privilege allows you to use the Report Objects pane. With the Report Objects pane, you can use dynamic aggregation to change the attributes available on the report layout.

- **Web use View Filter Editor**: This privilege allows you to add or modify a view filter for a report.

- **Desktop Analyst**: The predefined MicroStrategy Desktop Analyst group is assigned this set of privileges by default. Within the set of Desktop Analyst privileges, the following privileges are specific to OLAP Services:
  - **Create derived metrics**: This privilege allows you to create new calculations based on other metrics already on a base report.
  - **Use Report Objects window**: This privilege allows you to use the Report Objects pane. With the Report Objects pane, you can use dynamic aggregation to change the attributes available on the report layout.
  - **Use view filter editor**: This privilege allows you to add or modify a view filter for a report.

For information on all user privileges, refer to the *Supplemental Reference for System Administration*.

**Report types: Standard and OLAP reports**

During the report design process, if none of the OLAP Services features are applied on the report, then the report remains a standard report. This is
indicated by the word **Standard** in the bottom right corner of the Report Editor, as shown below.

In Desktop, the moment you use an OLAP Services feature, whether it is dynamic aggregation, a derived metric, a derived element, or a view filter, the standard report becomes an OLAP report, as shown in the following image. Notice that **Standard** is replaced by **OLAP** after a view filter is applied.

- The report type change from Standard to OLAP is displayed only in Desktop. The same indication is not shown on reports in MicroStrategy Web.

![Report details](image)

![View filter](image)
When a saved OLAP Services report is opened again, it remains an OLAP Services report—while a Standard report can be turned into an OLAP Services report, but an OLAP Services report cannot be turned into a Standard report.

Standard OLAP analysis features

Whether you use MicroStrategy Desktop or Web, once you execute a report, you can perform various OLAP manipulations on the report using features such as banding, sorting, pivoting, page-by, and so on. These features do not cause the report to be re-executed against the warehouse, and therefore have a much faster response time.

These standard OLAP features are different than the OLAP Services features covered in this guide. Each standard OLAP feature is described briefly in the following sections below. These features listed below are covered in detail in the Basic Reporting Guide and Advanced Reporting Guide:

- **Aliasing, page 10**
- **Banding, page 11**
- **Outline mode, page 11**
- **Page-by, page 11**
- **Pivoting, page 12**
- **Sorting, page 12**
- **Subtotals, page 12**
- **Thresholds, page 12**

Aliasing

When displaying a report, you can use the aliasing feature to rename any object on the report grid, such as attribute names, consolidation names, custom group names, and metric names. You can perform this task from both MicroStrategy Desktop and Web.
Banding

Banding allows you to color groups of rows or columns so that they form bands of data that are easy to locate and analyze. Banding can also make it easier to make sense of a very large report, because the large amounts of data are broken up into visual groups. If you need to keep track of values that mean different things in different columns (for example, dollars in one column and inventory quantities in another column), banding can help you avoid reading the wrong number.

Banding is a method of organizing or grouping data values in a grid report according to certain criteria. You can band rows or columns in several ways. You can band based on the number of rows or columns (for example, alternating color every 5 rows). You can also band data based on the row and column headers (for example, sorting the Units Sold column in order, then applying alternating colors to sets of values). For information on applying banding to a report, see the Basic Reporting Guide.

Outline mode

Outline mode allows you to create an indented grouping of related attribute elements. You can collapse and expand sections of related data. This function is particularly useful in instances where the information displayed would otherwise involve repetitive entries. For example, in the case of a grid report showing sales by year, each year is broken down by month. With outline mode enabled, the data is organized into groups, with months of each year nested below the years.

Page-by

Page-by is a way to segment data in a grid report by placing available attributes, consolidations, or metrics on a third axis called the Page axis. Based on the varying objects on the axis, you can view the report data in separate pages. This feature is most useful when you have an extremely long report with many objects, and you need to scroll to see all the data. You can page by many objects, such as attributes, metrics, hierarchies, consolidations, custom groups, and so on.
Pivoting

Pivoting enables you to rearrange the columns and rows in a report to view data from different perspectives. With data pivoting, you can do the following:

- Move an object from the row header to the column header
- Move an object from the column header to the row header
- Change the order of objects in the row header
- Change the order of objects in the column header

Sorting

Sorting allows you to specify an ascending or descending order for a particular row or column to present the report data. You can select what objects you want to sort, the sorting criteria, and the sorting order. MicroStrategy Desktop and Web offer quick sort, advanced sort, and hierarchical sort.

Subtotals

Using the Subtotal feature, you can add, remove, and edit the subtotals at different levels for metrics on the report. The subtotal functions available include sum, count, min, max, average, mean, median, and so on. You might choose to display all subtotals, a grand total only, or subtotals across levels where you select the object to be subtotaled. Additionally, Desktop allows you to construct custom subtotals that, for example, allow you to enable subtotals for selected metrics only.

Thresholds

A threshold highlights data that meets conditions defined by you. Highlighting data can include using different cell formats, symbols, images, or replacement text.
SHARING SETS OF DATA AMONG REPORTS: INTELLIGENT CUBES

Introduction

Intelligent Cubes are multi-dimensional cubes (sets of data) that allow you to use OLAP Services features on reports, as well as share sets of data among multiple reports. MicroStrategy Intelligent Cube Technology™ allows you to create these Intelligent Cubes, which operate within MicroStrategy Intelligence Server. On MicroStrategy’s business intelligence platform, you have two unique methods to implement Intelligent Cube Technology:

- **Personal Intelligent Cubes**: A personal Intelligent Cube is the Intelligent Cube functionality available in all pre-9.0 MicroStrategy releases. You automatically use personal Intelligent Cubes when you create reports in MicroStrategy as usual, and then analyze them with OLAP Services features such as view filters, derived metrics, and dynamic aggregation. These features are processed on the in-memory copy of data, known as a personal Intelligent Cube, rather than being processed on the data warehouse.

- **Intelligent Cubes**: An Intelligent Cube is a set of data that can be shared as a single in-memory copy, among many different reports created by multiple users. Rather than returning data from the data warehouse for a single report, you can return sets of data from your data warehouse and save them directly to Intelligence Server memory. The reports accessing
Intelligent Cubes can use all of the OLAP Services features for analysis and reporting purposes.

Personal Intelligent Cubes and Intelligent Cubes provide two unique methods to improve the performance of your system with the use of Intelligent Cube Technology and OLAP Services. Each method provides this solution for different scenarios. Depending on your business intelligence requirements, use the information in this section below decide when to use each method.

The major differences between the two methods center around how each set of data is shared and the restrictions made in accessing the data warehouse. Personal Intelligent Cubes allow a single report to use OLAP Services while still being able to seamlessly access the data warehouse. In contrast, Intelligent Cubes provide access to a single set of data that can be shared among multiple reports and analyzed with OLAP Services, while reducing access to the data warehouse. The table below outlines the differences between using personal Intelligent Cubes and Intelligent Cubes.

<table>
<thead>
<tr>
<th>Task</th>
<th>Personal Intelligent Cubes</th>
<th>Intelligent Cubes</th>
</tr>
</thead>
</table>
| Creating and managing Intelligent Cubes | All creation and management is handled automatically by Intelligence Server:  
  • Pro: No time is needed for the creation and administration of personal Intelligent Cubes.  
  • Con: You have less control over the set of data in Intelligence Server memory. | You must create, publish, and manage the Intelligent Cube available in Intelligence Server memory:  
  • Pro: You can create and publish the exact set of data required. You can use the Intelligent Cube Monitor to publish, share, and manage your Intelligent Cubes.  
  • Con: Time is required to create and manage your Intelligent Cubes. |
| Accessing the data warehouse (full relational analysis) | Full access to re-execute against the data warehouse is granted:  
  • Pro: You can refine your report with full relational analysis against the data warehouse.  
  • Con: As with any query against the data warehouse, this puts an extra load on the data warehouse and takes time to process. Also, the personal Intelligent Cube must be updated. | Access to re-execute against the data warehouse is only possible by drilling on data:  
  • Pro: The Intelligent Cube remains valid and processing time against the data warehouse is minimal. It is also more difficult or impossible for someone reporting on Intelligent Cube data to unintentionally cause re-execution against the data warehouse, preventing extra load on the data warehouse.  
  • Con: Refining the set of data in your Intelligent Cube published for multiple reports requires you to modify the Intelligent Cube and republish it to Intelligence Server. |
<table>
<thead>
<tr>
<th>Task</th>
<th>Personal Intelligent Cubes</th>
<th>Intelligent Cubes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sharing data</td>
<td>Personal Intelligent Cubes are linked to a single report:</td>
<td>Multiple reports can access a single Intelligent Cube:</td>
</tr>
<tr>
<td></td>
<td>• Pro: The report can use OLAP Services to analyze report data and is always updated with</td>
<td>• Pro: A single set of data in Intelligence Server memory can supply multiple</td>
</tr>
<tr>
<td></td>
<td>the latest version of the personal Intelligent Cube.</td>
<td>reports, reducing access to and processing in the data warehouse.</td>
</tr>
<tr>
<td></td>
<td>• Con: The copy of report data in Intelligence Server memory only applies to a single report.</td>
<td>• Con: Modification of an Intelligent Cube can invalidate the data for reports that</td>
</tr>
<tr>
<td></td>
<td></td>
<td>access the Intelligent Cube.</td>
</tr>
<tr>
<td>Filtering data</td>
<td>Both view filters and report filters can be used:</td>
<td>Only view filters can be used:</td>
</tr>
<tr>
<td></td>
<td>• Pro: You can take advantage of the reporting features available for view filters and</td>
<td>• Pro: Filtering is achievable with view filters and can be processed within an</td>
</tr>
<tr>
<td></td>
<td>report filters in the same report.</td>
<td>Intelligent Cube.</td>
</tr>
<tr>
<td></td>
<td>• Con: Modifying report filters requires re-execution against the data warehouse and</td>
<td>• Con: Some filtering techniques available with report filters are not available</td>
</tr>
<tr>
<td></td>
<td>modification to the personal Intelligent Cube.</td>
<td>with view filters, and thus not available for reports connected to an Intelligent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cube. For a comparison of view filter and report filter features, see</td>
</tr>
<tr>
<td></td>
<td></td>
<td>View filters versus report filters, page 188.</td>
</tr>
<tr>
<td>Using prompts</td>
<td>Prompts can be used just as in any standard report.</td>
<td>Prompts are limited to the objects included in Intelligent Cubes.</td>
</tr>
<tr>
<td></td>
<td>• Pro: You can use any applicable prompts in your project.</td>
<td>• Pro: All variations of prompt answers can be processed within an Intelligent</td>
</tr>
<tr>
<td></td>
<td>• Con: Modifying prompt answers requires re-execution against the data warehouse and</td>
<td>Cube.</td>
</tr>
<tr>
<td></td>
<td>modification to the personal Intelligent Cube.</td>
<td>• Con: Prompts must be created to only access data within an Intelligent Cube.</td>
</tr>
<tr>
<td>Applying security filters</td>
<td>Security filters are automatically detected and enforced.</td>
<td>Security filters are automatically detected and enforced.</td>
</tr>
<tr>
<td>Using consolidations and</td>
<td>Consolitions and custom groups can be used in reports that access personal Intelligent</td>
<td>Consolations and custom groups cannot be used in reports that access Intelligent</td>
</tr>
<tr>
<td>custom groups</td>
<td>Cubes.</td>
<td>Cubes. However, you can use derived elements in reports that access Intelligent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cubes to provide much of the same functionality.</td>
</tr>
</tbody>
</table>
The following sections in this chapter provide you with more insight into how personal Intelligent Cubes and Intelligent Cubes work, and when to use each approach to meet the needs of your business intelligence system:

- *About personal Intelligent Cubes, page 16*
- *Sharing Intelligent Cubes, page 19*
- *Updating Intelligent Cubes without re-processing: Incremental Refresh, page 45*

### About personal Intelligent Cubes

To understand how reports work with a personal Intelligent Cube, a comparison is made between a report cache and a personal Intelligent Cube.

A cache is a recently used result set that is stored on a system to improve response time in future requests. In the MicroStrategy environment, a report cache is a file created by Intelligence Server based on the data definition and the caching properties of the report. The report cache contains pre-processed report data that is saved on disk and in Intelligence Server’s memory. All subsequent requests for the same report are processed using the saved report cache, given that certain criteria are met, such as matching report ID, filter ID, security ID, and so on. For more information on report cache matching, see *Improving Report and Document Response Time: Caching* in the *System Administration Guide*. 

<table>
<thead>
<tr>
<th>Task</th>
<th>Personal Intelligent Cubes</th>
<th>Intelligent Cubes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using derived elements</td>
<td>You cannot use derived elements in a personal Intelligent Cube; they can only be used in a report that accesses a shared Intelligent Cube. However, you can use consolidations and custom groups in reports that access personal Intelligent Cubes to provide much of the same functionality.</td>
<td>Derived elements can be used and created on reports that access an Intelligent Cube. For information on derived elements, see <em>Chapter 4, Derived Elements</em>.</td>
</tr>
<tr>
<td>Using dynamic aggregation, view filters, and derived metrics</td>
<td>All of these OLAP Services features are available for reports accessing a personal Intelligent Cube.</td>
<td>All of these OLAP Services features are available for reports accessing an Intelligent Cube.</td>
</tr>
</tbody>
</table>
It is faster and more efficient to retrieve the same report results from the report cache than re-executing queries against the data warehouse. Report caching shortens the database processing time and also enhances user scalability, resulting in improved throughput. One drawback with any cache is that if the report is manipulated in such a way as to change the data definition, the report cache is invalidated and re-execution against the data warehouse is necessary to update the report cache.

The personal Intelligent Cube is similar to the report cache; however, it acts as a layer between the report cache and the report itself. This copy of report data is stored in Intelligence Server memory only. Whenever a report executes against a data warehouse and returns a new set of data, a personal Intelligent Cube is created automatically; however, Intelligent Cubes are always manually created. For information on creating Intelligent Cubes, see Creating Intelligent Cubes, page 26.

Subsequent requests for the same report take advantage of the personal Intelligent Cube without having to re-execute the report against the data warehouse. The logic to match these requests with the appropriate personal Intelligent Cube is automatically handled within the MicroStrategy platform. Personal Intelligent Cubes are managed automatically, and there is no need to specify which personal Intelligent Cube to connect to. The Analytical Engine then uses this set of data to render different report views. These views are created by processing manipulations using OLAP Services features, such as dynamic aggregation, derived metrics, view filters, derived elements, and
so on. OLAP Services features are handled within the personal Intelligent Cube, and therefore no re-execution against the data warehouse is necessary.

As illustrated above, while a report cache is saved in-memory and on the disk, the personal Intelligent Cube is only kept in Intelligence Server’s memory. A combination of the complete data definition and the view of the data is stored in the personal Intelligent Cube and displayed on the report.

Relational analysis with personal Intelligent Cubes

While personal Intelligent Cubes allow reports to execute against a copy of report data in Intelligence Server memory, you can easily and seamlessly extend your analysis to the complete data warehouse. Unlike MOLAP cubes that have a fully restricted set of data to report against, reports in MicroStrategy can execute within a personal Intelligent Cube, as well as use ROLAP analysis to return new data from the data warehouse. This allows you to include new metrics, attributes, report filter conditions, and other data from your data warehouse into your report.
ROLAP manipulations require re-execution against the data warehouse, and therefore should only be used when necessary. Whenever a report manipulation requires new report SQL to be generated and executed against the data warehouse, a confirmation message will be displayed. Once the data is returned from the data warehouse, the personal Intelligent Cube is automatically updated and you can begin executing against the updated copy of report data.

Intelligent Cubes apply stricter constraints than personal Intelligent Cubes, limiting reporting and analysis to execute against the copy of data within Intelligence Server. Therefore, Intelligent Cubes help to limit the amount of processing done in the data warehouse. (For more information, see Sharing Intelligent Cubes, page 19.) This is in direct contrast to the ability to perform ROLAP analysis with personal Intelligent Cubes as discussed above.

**Maintaining data access security**

Personal Intelligent Cubes automatically inherit all of the security provisions of every object they use. These provisions are dynamically applied to each user’s MicroStrategy security profile when the personal Intelligent Cube is accessed.

For example, if a user has privileges to use four of the five metrics contained in a personal Intelligent Cube, that restriction is automatically applied when the user accesses the personal Intelligent Cube. In other words, the user is given access to the four metrics in the personal Intelligent Cube that he or she has MicroStrategy privileges to use.

For information about object security in general, refer to the System Administration Guide.

**Sharing Intelligent Cubes**

MicroStrategy 9.0 and later enhances the scalability and manageability of OLAP Services with the addition of sharable Intelligent Cubes. Intelligent Cubes allow multiple reports to retrieve data from a single shared in-memory set of data.

Unlike personal Intelligent Cubes that are created automatically, you create Intelligent Cubes and publish them as a shared data source for the users to build reports from. Intelligent Cubes provide the fast response time and
analytic calculations that are often associated with Multidimensional Online Analytic Processing (MOLAP) cubes, while also benefiting from the ability to use Relational Online Analytic Processing (ROLAP) by drilling into the full set of data outside of the Intelligent Cube. In addition, Intelligent Cubes are fully scalable, limiting excessive data consumption and redundant data by allowing you to build only the sets of data you require.

With Intelligent Cubes you can return a specific set of data from your data warehouse. Users can then create reports that display and analyze a subset of the set of data defined in an Intelligent Cube. This process is different than the common approach of creating a report that directly accesses the data warehouse.

Intelligent Cubes act as a layer between your data warehouse and MicroStrategy reports that analyze and display data, as illustrated below.

The abstraction that Intelligent Cubes provide between your data warehouse and reports can improve the performance of your business intelligence application in the following ways:

• Reports that connect to an Intelligent Cube can perform reporting and analysis manipulations within the Intelligent Cube without hitting the data warehouse. These manipulations are executed much faster than running a new query against a data warehouse.

• The data that reports can access is restricted to the data within the Intelligent Cube. Users can still perform a few ROLAP manipulations such as drilling that can access the data warehouse. However, these types
of manipulations that cause re-execution against the data warehouse are not as accessible as they are when using personal Intelligent Cubes.

For example, when using a personal Intelligent Cube, users can access any attribute defined for the project and include it on the report. A manipulation of this type requires re-execution against the data warehouse. Conversely, a user working in a report that connects to an Intelligent Cube can only add attributes to the report grid if the attributes are included in the Intelligent Cube.

Therefore, Intelligent Cubes help to limit the amount of processing done in the data warehouse and improve performance.

• Security filters are applied separately for each user at the level of the report connecting to the Intelligent Cube, rather than having to create multiple Intelligent Cubes for each security filter. For more information, see Maintaining data access security, page 21.

Maintaining data access security

 Reports that connect to Intelligent Cubes adhere to many of the same standards of data access security as the rest of your MicroStrategy project. You can control users’ access to data with security filters. For more information on security filters, see the System Administration Guide.

User and group security filters are applied automatically on reports that connect to an Intelligent Cube, as shown below:
This approach allows a single Intelligent Cube to be used by multiple security filters, rather than having to create separate Intelligent Cubes for each security filter. By using a single Intelligent Cube to support all the security filters for a project, data access security is implemented automatically with minimal burden on Intelligence Server memory.

However, there are some differences in security filter resolution for reports that connect to Intelligent Cubes as compared to reports that directly access the data warehouse. For information on these differences, see Security filter resolution for reports connected to Intelligent Cubes below.

**Security filter resolution for reports connected to Intelligent Cubes**

Security filter resolution for reports that connect to Intelligent Cubes differs as compared to reports that directly access the data warehouse. These differences relate to what attributes are on a user’s security filter, and how these relate to the attributes and fact entry levels of metrics available in the Intelligent Cube used for the report.

If all attributes in a user’s security filter are in the Intelligent Cube that is used for the report, then security filters can be resolved using the standard process. However, if some of the attributes in a user’s security filter are not in the Intelligent Cube used for the report, security filter resolution can differ from the standard process.

While this scenario is uncommon, it can cause users of reports that access Intelligent Cubes to experience one of the following results:

- No data is returned for the report.
  
  This could also be caused by the user creating a view filter that is too restrictive, or by the user’s normal security filter resolution.

- A metric or metrics are displayed with no data returned for the metric.

  Another possible cause of metrics not properly returning data on reports connected to Intelligent Cubes is the use of dynamic aggregation (see Chapter 5, Dynamic Aggregation). However, when this is caused by dynamic aggregation, null values are displayed for the metric rather than
not displaying any information at all. The image below shows the difference between security filter resolution and dynamic aggregation as the cause for metrics not displaying any data.

By default, null values are represented by dashes (--) on reports. For information on changing the display of null values, see *Changing the display of null values, page 160.*

For more specific scenarios and examples of when security filter resolution can occur, see *Security filter resolution when attributes in a user’s security filter are not in the Intelligent Cube used for the report, page 24.*

If a user is experiencing one of the two scenarios listed above due to security filter resolution, the following resolutions can be considered:

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>The user continues to use the report that accesses the Intelligent Cube.</td>
<td>• Data access security is maintained. No additional resources are needed to modify the Intelligent Cube or to create a new report.</td>
<td>• Some data that may be available to the user by directly querying a data source may not be available in the report that accessed an Intelligent Cube.</td>
</tr>
</tbody>
</table>
| The user creates or views a report with the same definition that directly queries a data source rather than accessing an Intelligent Cube. | • The user is able to verify the full results that can be returned for such a report. | • A new report that directly queries a data source must be created.  
  • The new report cannot take advantage of the improved query performance of accessing an Intelligent Cube. |
When attributes in a user’s security filter are not in the Intelligent Cube used for the report, the outcome depends on how the attributes are related to those in the Intelligent cube, as described below:

- **Attributes in the security filter are related to attributes in the Intelligent Cube**: No data is returned, to maintain data access security.

  For example, an Intelligent Cube includes the attributes Year and Region, and the metric Revenue. A user creates a report that connects to this Intelligent Cube, and includes Year and Revenue on the report. The user’s security filter is defined on the attribute Quarter to return data only for the first quarter of 2008.

  By including the Year attribute on the report, this report would return information for all quarters in each year. However, the user is allowed to only see data for the first quarter of 2008. To maintain this data access security, no data is returned for the report.

- **Attributes in the security filter are not related to attributes in the Intelligent Cube**: The data returned depends on whether metrics in the Intelligent Cube report fact data based on attributes related to those in the user’s security filter:
  
  - A metric in the Intelligent Cube reports fact data based on an attribute related to one in the security filter: The user’s security filter prevents any data from being returned.

    For example, an Intelligent Cube includes the attributes Year and Region, and the metric Revenue, which is based on the fact Revenue. This fact, in turn, is reported based on the attributes Item, Day, and Call Center. A user creates a report that connects to this Intelligent Cube and includes Year and Revenue in the report. The user’s security filter is defined on the attribute Month to return data only for the month of January 2008.

    By including the Year attribute on the report, this report would return information for all years and months in each year. However, the user is allowed to only see data for the month of January 2008. To maintain this data access security, the data returned depends on whether metrics in the Intelligent Cube report fact data based on the attributes related to those in the user’s security filter.
Cube, and includes Year and Revenue on the report. However, this
user’s security filter is defined on the attribute Category to return data
only from Books.

Since Revenue is based on Item, Day and Call Center only, it cannot be
reported based on the Category attribute. In such a case, no data will
be reported for the Revenue metric.

- **A metric in the Intelligent Cube reports fact data based on an
  attribute unrelated to those in the security filter:** The security filter
does not apply any restriction, and displays the data for metrics based
on the fact.

- The table below describes the security filter resolution for scenarios listed
above:

<table>
<thead>
<tr>
<th>Attributes in the security filter are related to attributes in the Intelligent Cube</th>
<th>A fact in the Intelligent Cube is reported based on an attribute in the security filter</th>
<th>A fact in the Intelligent Cube is not reported based on any of the attributes in the security filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes in the security filter are not related to attributes in the Intelligent Cube</td>
<td>No data is returned, to maintain data access security.</td>
<td>No data is returned, to maintain data access security.</td>
</tr>
<tr>
<td></td>
<td>Data can be returned using the standard security filter resolution. However, to maintain data access security, no data is displayed for any metrics where fact data is reported based on attributes related to those in the security filter.</td>
<td>Data can be returned using the standard security filter resolution. In this scenario the security filter does not need to restrict any data, and the metric data can also be displayed.</td>
</tr>
</tbody>
</table>

**Data access security with connection mapping**

In MicroStrategy, you can also use connection mapping to control the data
that users have access to when they run reports. You can also apply this
security when users create reports that connect to an Intelligent Cube. For
information on maintaining data access security with connection mapping to
Intelligent Cubes, see the *System Administration Guide*.

**Setting permissions for individual Intelligent Cubes**

You can use an Intelligent Cube’s Access Control List (ACL) to set specific
access permissions for users. For example, you can restrict some users to
only create reports based on an Intelligent Cube, but not re-execute the Intelligent Cube.

You can use the ACL Editor in MicroStrategy Web to assign the following permission groups to users, for each Intelligent Cube:

<table>
<thead>
<tr>
<th>Group</th>
<th>Description</th>
<th>Permissions granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consume</td>
<td>Grants permission to create and execute reports based on this Intelligent Cube.</td>
<td>• Browse • Read • Use</td>
</tr>
<tr>
<td>Add</td>
<td>Grants permission to create and execute reports based on this Intelligent Cube, and republish/re-execute the Intelligent Cube to update the data.</td>
<td>• Browse • Read • Use • Execute</td>
</tr>
<tr>
<td>Collaborate</td>
<td>Grants permission to create and execute reports based on this Intelligent Cube, republish/re-execute the Intelligent Cube to update the data, and modify the Intelligent Cube.</td>
<td>• Browse • Read • Write • Delete • Use • Execute</td>
</tr>
</tbody>
</table>

For information on ACLs and access permissions, see the *System Administration Guide*.

**Creating Intelligent Cubes**

Creating Intelligent Cubes is as easy as creating reports. This section covers the following methods for creating Intelligent Cubes:

- *Prerequisites to creating Intelligent Cubes, page 27*
- *Supporting various features with Intelligent Cubes, page 28*
- *Building an Intelligent Cube, page 29*
- *Converting reports into Intelligent Cubes, page 31*
- *Updating Intelligent Cubes using schedules, page 32*

You need the Use Intelligent Cube Editor privilege to create Intelligent Cubes. This privilege is part of OLAP Services.
After you have created your Intelligent Cube, you can publish its set of data to be shared by multiple reports. For more information on publishing an Intelligent Cube, see *Publishing Intelligent Cubes, page 40*.

**Prerequisites to creating Intelligent Cubes**

Make sure you review the following prerequisites before you create an Intelligent Cube in MicroStrategy. They can save you time and make your Intelligent Cubes more effective for your reporting environment:

While creating Intelligent Cubes, bear in mind that Intelligent Cubes can deplete Intelligence Server’s system resources. Create Intelligent Cubes for logical subsets of your data, rather than using them as a reflection of your entire data warehouse. For information on managing the size of Intelligent Cubes, see *Managing Intelligent Cubes* in the *System Administration Guide*.

- Define your business query: Before you can determine what objects to place on an Intelligent Cube, you need to know what data you want to make available for reports to access directly. To define your Intelligent Cube, make sure you consider the following questions:
  - What subset of business queries does the Intelligent Cube need to provide data for? Intelligent Cubes allow you to create sets of data that can support multiple reports that answer variations to similar business queries.
  - Do you have reports that currently access your data warehouse that could benefit from accessing an Intelligent Cube instead? To support this scenario, you can use dynamic sourcing to connect these reports to Intelligent Cubes that you create. You can also use MicroStrategy Cube Advisor to create Intelligent Cubes that these reports can access. For information on using Cube Advisor to support dynamic sourcing, see *Using Cube Advisor to support dynamic sourcing, page 260*.

- Look for existing Intelligent Cubes: Before you create an Intelligent Cube, search through MicroStrategy to see whether a similar Intelligent Cube already exists that can serve the same purpose as the Intelligent Cube that you intend to create. This can not only save you time, it can help you avoid unnecessary duplication in your MicroStrategy project. You can search a project for Intelligent Cubes, or you can view Intelligent Cubes created for your projects in the Intelligent Cube Monitor. For information on using the Intelligent Cube Monitor, see the *MicroStrategy System Administration Guide*. 
• Review the feature support for Intelligent Cubes: The MicroStrategy features that are available for Intelligent Cubes differ from those available for reports. To review this list, see Supporting various features with Intelligent Cubes below.

Supporting various features with Intelligent Cubes

Intelligent Cubes act as a set of data for other reports to connect to and use OLAP Services to report, analyze, and display data. Since Intelligent Cubes function simply as a set of data, Intelligent Cubes do not need to contain objects that modify the display of data. The display of data is handled by the reports that access the Intelligent Cube.

For these reasons, the following objects and features cannot be included in Intelligent Cubes in the same ways that they can be included for reports:

• **Consolidations and custom groups**: Consolidations and custom groups cannot be included in Intelligent Cubes or in reports that access Intelligent Cubes. However, you can include derived elements in reports that access Intelligent Cubes. Derived elements combine the analysis features of consolidations and custom groups, while executing against the Intelligent Cube rather than the data warehouse. For information on derived elements, see Chapter 4, Derived Elements.

• **OLAP Services features**: View filters, derived metrics, and derived elements cannot be included in Intelligent Cubes, but these features are the main analysis tools for reports that access Intelligent Cubes. For an overview of OLAP Services features and how these can be created in reports, see Chapter 1, About MicroStrategy OLAP Services.

• **Transformation shortcut metrics**: These metrics can be created on reports using a similar workflow as derived metrics that are created as shortcut metrics. For information about the difference between transformation shortcut metrics and shortcut metrics that are also derived metrics, see Using rank and percent-to-total metric analysis, page 173. As an alternative, you can create transformation metrics using the Metric Editor. Metrics that use transformations in this way can be included in Intelligent Cubes. To create transformation metrics using the Metric Editor, see the Advanced Reporting Guide.

• **Prompts**: Prompts cannot be included in Intelligent Cubes. However, reports that access Intelligent Cubes can use prompts that access only data included in the Intelligent Cube. This saves processing time by ensuring that re-execution against the data warehouse is not necessary.
Building an Intelligent Cube

Building an Intelligent Cube is similar to building a report. You can add data for your Intelligent Cube by including objects such as attributes, metrics, filters, and so on as report objects or report filters. For a list of objects and features that cannot be included in the definition of Intelligent Cubes, see Supporting various features with Intelligent Cubes, page 28.

Since Intelligent Cubes are used simply to share a set of data, no data or report results are displayed when you execute an Intelligent Cube. However, executing an Intelligent Cube publishes the Intelligent Cube, which can then be accessed as a set of data for multiple reports (see Publishing Intelligent Cubes, page 40).

Prerequisites

- Review the information provided in Prerequisites to creating Intelligent Cubes, page 27.
• You need the Use Intelligent Cube Editor privilege to create Intelligent Cubes. This privilege is part of OLAP Services.

**To build an Intelligent Cube**

1 In Desktop, from the **File** menu select **New**, and then **Intelligent Cube**. The New Intelligent Cube dialog box opens.

   If the New Intelligent Cube dialog box does not open, from the **Tools** menu, select **Desktop Preferences**. Expand the Object Templates category, select **General**, and from the Show templates for the following objects list, select **Report**. Click **OK** to accept your changes, and then repeat the previous step to open the New Grid dialog box.

2 Select **Empty Intelligent Cube** and click **OK**. The Report Editor opens.

3 Add objects such as attributes, metrics, and so on for the Intelligent Cube, the same way you would add report objects.

4 Create a filter for the Intelligent Cube as needed.

   If you create a filter on an Intelligent Cube, any data that is restricted from the Intelligent Cube is not available for any reports that connect to the Intelligent Cube. While this can help reduce the size of the Intelligent Cube, it also reduces the amount of data available in the Intelligent Cube.

5 Click **Save and close** to save the Intelligent Cube and close the Report Editor.

6 To publish an Intelligent Cube, see *Publishing Intelligent Cubes, page 40*.

**Building Intelligent Cubes from MDX data sources, or using Freeform SQL**

You can create Intelligent Cubes with data from an MDX data source, such as SAP BW, Hyperion Essbase, or Microsoft Analysis services. To create an Intelligent Cube based on MDX data in Desktop, open the **File** menu, select **New**, and select **Intelligent Cube**. In the New Intelligent Cube dialog box, select the **MDX Sources** tab. For instructions on configuring MDX sources, refer to the *MDX Cube Reporting Guide*. 
You can also create Intelligent Cubes by directly querying your data using SQL. You can use Freeform SQL to write your own SQL statements, or Query Builder to create a query using a graphical interface. These options are available in the New Intelligent Cube dialog box, under the ODBC Sources tab. For information on using Freeform SQL and Query Builder to connect to ODBC data sources, refer to the Advanced Reporting Guide.

Converting reports into Intelligent Cubes

If you have an existing report that contains a set of objects you want to use in an Intelligent Cube, you can convert the report into an Intelligent Cube.

Creating an Intelligent Cube by converting a report allows you to view the set of data you are including in your Intelligent Cube. This is an easy way to share a report’s set of data among multiple reports.

Intelligent Cubes do not display report results in the same way as reports can. Instead of building an Intelligent Cube from the start, you can build a report; execute it; view the report data in grid, graph, or grid and graph mode; and then convert the report to an Intelligent Cube. With this method you can verify that you have the set of data you want, rather than just the required report objects, before you publish it as an Intelligent Cube.

When you convert a report to an Intelligent Cube, some parts of the report are not included in the resulting Intelligent Cube. Intelligent Cubes are not used for the same display and analysis purposes as a report. Intelligent Cubes simply act as a sharable set of data. Therefore, when a report is converted into an Intelligent Cube, some of the display and analysis features are no longer necessary. For a list of features that are not included in an Intelligent Cube when converting a report to an Intelligent Cube, see Supporting various features with Intelligent Cubes, page 28.

The procedure below explains the high-level steps for converting a report to an Intelligent Cube. The procedure assumes you have already created a report.

**Prerequisites**

- You need the Use Intelligent Cube Editor privilege to create Intelligent Cubes. This privilege is part of OLAP Services.
To convert a report to an Intelligent Cube

1. Right-click the report and select Edit. The Report Editor opens.

2. From the Data menu, point to Intelligent Cube Options, and select Convert to Intelligent Cube.

3. If the report contains objects that cannot be included in the Intelligent Cube, one of the following messages is displayed:
   - If the report includes OLAP Services features such as view filters, derived metrics, or dynamic aggregation, you are prompted to automatically remove these features as part of the conversion process. Click Yes to have these features automatically removed so that the report can be converted into an Intelligent Cube.
   - If the report includes features such as consolidations, custom groups, or prompts, a warning message is displayed that explains that these objects cannot be included in the Intelligent Cube. Click OK to close the warning message and then manually remove the objects from the report. You can then attempt to convert the report to an Intelligent Cube again.

4. Once the conversion process is completed successfully, save the Intelligent Cube.

5. You must publish an Intelligent Cube to make it available for multiple reports to access and report on its set of data. To publish an Intelligent Cube, see Publishing Intelligent Cubes, page 40.

Updating Intelligent Cubes using schedules

While you are creating or modifying an Intelligent Cube, you can schedule when an Intelligent Cube should be re-executed against the data warehouse to update its data. For example, you can schedule to re-execute and republish an Intelligent Cube when a database load occurs.

The procedure below describes the steps to create a subscription that schedules the re-execution and republication of a single Intelligent Cube. You can also schedule multiple Intelligent Cubes to be updated using a single subscription, which is described in Publishing Intelligent Cubes using a schedule, page 41.
Prerequisites

- An Intelligent Cube has been created.
- You have created the schedule to subscribe the publication of an Intelligent Cube to. For information on creating schedules, see the *System Administration Guide*.

To schedule an Intelligent Cube to be updated

1. In Desktop, browse to an Intelligent Cube.
2. Right-click the Intelligent Cube, point to **Schedule Delivery To**, and then **Refresh Cube**. The Subscription Editor opens.
3. Select a schedule from the Schedule drop-down list.
4. Select other schedule options as required, and click **OK**. The Subscription Editor closes and you are returned to Desktop.

Once the schedule is triggered the Intelligent Cube is re-executed against the data warehouse and published to the Intelligent Cube Monitor.

Enabling ROLAP drilling for reports accessing Intelligent Cubes

Reports that access an Intelligent Cube can be granted full ROLAP access to the data warehouse through drilling. This means that you can enable drilling outside of the Intelligent Cube to access the full set of data available in the data warehouse.

While this extends the analysis and data access capabilities of reports that access Intelligent Cubes, drilling outside of an Intelligent Cube can require additional load on the Intelligence Server and data warehouse. This is because drilling outside of an Intelligent Cube requires a new report to be executed against the data warehouse.

All reports that access Intelligent Cubes can drill within the data included in an Intelligent Cube. This provides ROLAP-type analysis without having to re-execute against the data warehouse. For example, an Intelligent Cube includes Year and Quarter. A report accessing the Intelligent Cube only includes Year on the report. On the report, you can drill down from Year to Quarter, which returns the results without any extra load on the data warehouse or Intelligence Server.
The decision to enable or disable drilling outside of an Intelligent Cube depends on a few factors. You should consider the size and complexity of your Intelligent Cubes when deciding whether to enable drilling outside of an Intelligent Cube. While enabling drilling outside of relatively small Intelligent Cubes can give the benefit of ROLAP analysis through drilling, enabling this analysis on relatively large Intelligent Cubes has the potential to cause increased load on your data warehouse and Intelligence Server.

By enabling drilling outside of an Intelligent Cube, reports that access the Intelligent Cube have ROLAP access to data through drilling. For example, an Intelligent Cube includes Year but not Quarter. A report accessing the Intelligent Cube includes Year and drilling outside of the Intelligent Cube is enabled. On the report, you drill down from Year to Quarter. This causes a new report to be created and executed against the data warehouse.

The benefit of enabling this extra analysis can come with a performance cost. As mentioned above, when you drill outside of an Intelligent Cube a new report is created and executed against the data warehouse. This action does not take advantage of the data stored in the Intelligent Cube.

When this drilled-to report is created, only objects that were on the report layout of the report you drilled from are included in the drilled-to report. Any objects that are only in the Report Objects pane of the report you drilled from are not included in the drilled-to report. This can help reduce the size of the drilled-to report. However, if you are drilling from a report that accesses a large Intelligent Cube, it is possible that a user could include all objects of an Intelligent Cube on a report. Drilling outside of the Intelligent Cube on such a report could cause excessive load on the data warehouse and Intelligence Server.

The procedures below describe the steps to enable drilling outside of a specific Intelligent Cube, or any Intelligent Cubes in a project:

- To enable or disable drilling outside of an Intelligent Cube for reports accessing a specific Intelligent Cube, page 35
- To enable or disable drilling outside of an Intelligent Cube for reports accessing any Intelligent Cubes in a project, page 35

**Prerequisites**

- An Intelligent Cube has been created.
- You need the Use Intelligent Cube Editor privilege. This privilege is part of OLAP Services.
To enable or disable drilling outside of an Intelligent Cube for reports accessing a specific Intelligent Cube

1 In Desktop, browse to an Intelligent Cube.

2 Right-click the Intelligent Cube, and select **Edit**. The Intelligent Cube is opened in the Report Editor.

3 From the **Data** menu, select **Configure Intelligent Cube**. The Intelligent Cube Options dialog box opens.

4 From the **Categories** list, expand **Options**, and then select **General**.

5 In the **Drilling** area, clear the **Use default settings** check box.

6 Select the **Allow reports to drill outside the Intelligent Cube** check box. You can clear this check box to disable drilling outside of an Intelligent Cube.

7 Click **OK**. The Intelligent Cube Options dialog box closes and you are returned to the Intelligent Cube.

8 Click **Save and Close** to save your changes and close the Report Editor.

9 To make your changes available for reports accessing the Intelligent Cube, you must publish the Intelligent Cube, which is described in *Publishing Intelligent Cubes, page 40.*

To enable or disable drilling outside of an Intelligent Cube for reports accessing any Intelligent Cubes in a project

This procedure enables drilling outside of all Intelligent Cubes within a project. Review the considerations in *Enabling ROLAP drilling for reports accessing Intelligent Cubes, page 33* before enabling drilling outside of all Intelligent Cubes for a project.

1 In Desktop, log in to a project with a user account with administrative privileges.

2 Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3 From the Categories list, expand Intelligent Cubes, and then select General.

4 Select the Allow reports to drill outside the Intelligent Cube check box. You can clear this check box to disable drilling outside of Intelligent Cubes.

5 Click OK. The Project Configuration Editor closes.

6 To make your changes available for reports accessing the Intelligent Cubes, you must publish the Intelligent Cubes, which is described in Publishing Intelligent Cubes, page 40.

Localizing Intelligent Cubes in multiple languages

If your MicroStrategy project supports the localization of information into multiple languages, you can also support this localization to provide Intelligent Cubes in multiple languages.

When an Intelligent Cube is defined to support a language, data is available for the Intelligent Cube in that language. When a user logs in to MicroStrategy in the locale for that language and creates or views a report that accesses the Intelligent Cube, all data that has been localized is returned.

For example, a report connected to an Intelligent Cube includes Region, Category, Profit, and Discount. The Intelligent Cube is defined to include information for the English and Italian locales. This report connected to the Intelligent Cube is executed by a user with an English locale. This same
The same report returns data that matches the user’s locale. This is because the Intelligent Cube has been defined include localized data.

The SQL view of these reports shown below demonstrates how the data is returned in different languages:

**SQL generated for English locale**

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Discount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td>$7,813</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td>$162,014</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$16,347</td>
<td>$7,799</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td>$35,082</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$46,461</td>
<td>$6,648</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td>$736,291</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td>$6,508</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td>$6,021</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>Books</td>
<td>$455,066</td>
<td>$62,663</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,500,514</td>
<td>$270,429</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,367</td>
<td>$12,099</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$25,895</td>
<td>$11,363</td>
<td></td>
</tr>
<tr>
<td>Northwest</td>
<td>Books</td>
<td>$19,734</td>
<td>$2,757</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,604,110</td>
<td>$289,165</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$3,639</td>
<td>$2,712</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$5,508</td>
<td>$2,436</td>
<td></td>
</tr>
<tr>
<td>South</td>
<td>Books</td>
<td>$216,663</td>
<td>$34,020</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$739,063</td>
<td>$136,079</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$12,905</td>
<td>$6,155</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,002</td>
<td>$6,145</td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td>Books</td>
<td>$25,009</td>
<td>$3,599</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$419,847</td>
<td>$71,154</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$37,704</td>
<td>$17,212</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$5,965</td>
<td>$3,197</td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td>Books</td>
<td>$41,576</td>
<td>$5,739</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$686,421</td>
<td>$126,344</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td>$29,791</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,294</td>
<td>$5,229</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$20,195</td>
<td>$4,339</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td>$88,912</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td>$20,360</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td>$3,786</td>
<td></td>
</tr>
</tbody>
</table>

**SQL generated for Italian locale**

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Sconto</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlantico centrale</td>
<td>Libri</td>
<td>$46,461</td>
<td>$6,648</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td>$736,291</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$14,351</td>
<td>$6,508</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$13,737</td>
<td>$6,021</td>
<td></td>
</tr>
<tr>
<td>Centrale</td>
<td>Libri</td>
<td>$55,495</td>
<td>$7,813</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td>$736,291</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$16,347</td>
<td>$7,799</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$78,357</td>
<td>$35,082</td>
<td></td>
</tr>
<tr>
<td>Nord-est</td>
<td>Libri</td>
<td>$455,066</td>
<td>$62,663</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,500,514</td>
<td>$270,429</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$27,367</td>
<td>$12,099</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$25,895</td>
<td>$11,363</td>
<td></td>
</tr>
<tr>
<td>Nord-ovest</td>
<td>Libri</td>
<td>$19,734</td>
<td>$2,757</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,604,110</td>
<td>$289,165</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$5,639</td>
<td>$2,712</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$5,508</td>
<td>$2,436</td>
<td></td>
</tr>
<tr>
<td>Sud</td>
<td>Libri</td>
<td>$216,663</td>
<td>$34,020</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronica</td>
<td>$739,063</td>
<td>$136,079</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$12,905</td>
<td>$6,155</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$11,002</td>
<td>$6,145</td>
<td></td>
</tr>
<tr>
<td>Sud-est</td>
<td>Libri</td>
<td>$25,009</td>
<td>$3,599</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electtronica</td>
<td>$419,847</td>
<td>$71,154</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$37,704</td>
<td>$17,212</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$5,965</td>
<td>$3,197</td>
<td></td>
</tr>
<tr>
<td>Sud-ovest</td>
<td>Libri</td>
<td>$41,576</td>
<td>$5,739</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronica</td>
<td>$686,421</td>
<td>$126,344</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$60,787</td>
<td>$29,791</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$11,294</td>
<td>$5,229</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Libri</td>
<td>$29,195</td>
<td>$4,339</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronica</td>
<td>$469,635</td>
<td>$88,912</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Film</td>
<td>$42,611</td>
<td>$20,360</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Musica</td>
<td>$7,952</td>
<td>$3,786</td>
<td></td>
</tr>
</tbody>
</table>
Notice that different columns of data were returned for the report based on the locale used to run the report. This demonstrates a column-based solution to localizing your data. It is recommended that you use a column-based solution rather than a connection mapping-based solution to localize your data for Intelligent Cubes. For information on this recommendation, see the *Supplemental Reference for System Administration*.

Localizing an Intelligent Cube does not localize any data, it only returns data that has already been configured in a MicroStrategy project as part of a localization solution. For information on defining localization rules for your MicroStrategy projects, see the *Project Design Guide*.

Providing localized data causes the Intelligent Cube size to be larger than if it supported only a single locale. However, providing localized data in Intelligent Cubes is necessary if users expect reports that return Intelligent Cube data to return localized data to reflect their locale.

The steps below let you define Intelligent Cubes to support various languages.

**Prerequisites**

- An Intelligent Cube has been created.
- The project for the Intelligent Cube has been configured to support multiple languages. For information on localizing projects, see the *Project Design Guide*.
- You need the Use Intelligent Cube Editor privilege. This privilege is part of OLAP Services.

**To support multiple languages in an Intelligent Cube**

1. In Desktop, browse to an Intelligent Cube.
2. Right-click the Intelligent Cube and select **Edit**. The Intelligent Cube opens in the Report Editor.
3. From the **Data** menu, select **Configure Intelligent Cube**. The Intelligent Cube Options dialog box opens.
4. From the **Categories** list, expand **Options**, and then select **Data Languages**.
5 Select from the following localization options:

- **Use project default data language (Default)**: Select this option to provide data only in the default language for the project. The default language for the project is highlighted in blue.

- **All project data languages**: Select this option to provide data for the languages available for the project.

- **Select specific languages**: Select this option to select from all languages available for the project. Data is provided for the languages you select.

All of these options are based on the language options defined for the project, which can be modified from the Project Configuration Editor. For information on localizing projects, see the *Project Design Guide*.

6 Click **OK**. The Intelligent Cube Options dialog box closes and you are returned to the Intelligent Cube.

7 Click **Save and Close** to save your changes and close the Report Editor.

8 To make your changes available for reports accessing the Intelligent Cube, you must publish the Intelligent Cube, which is described in *Publishing Intelligent Cubes, page 40*.

### Enabling dynamic sourcing of Intelligent Cubes

The general workflow of reporting on data in Intelligent Cubes is to create a report that is linked to a published Intelligent Cube. Dynamic sourcing extends the accessibility of Intelligent Cubes by allowing standard reports to access any published Intelligent Cubes that can satisfy the data requirements of the report. This connection is performed automatically without the user having to explicitly connect their report to a published Intelligent Cube. This allows users performing ROLAP analysis to work completely within a pre-created set of data without having to manually find and select the Intelligent Cube that fits their analysis.

For information on dynamic sourcing, see *Chapter 8, Dynamic Sourcing*.
Publishing Intelligent Cubes

Publishing an Intelligent Cube retrieves data from the data warehouse and stores this data as an Intelligent Cube in Intelligence Server memory. Once an Intelligent Cube is stored in Intelligence Server memory, multiple reports can be created to view and analyze the set of published data.

When an Intelligent Cube is published, the Intelligent Cube Monitor displays the Intelligent Cube. You can manage your published Intelligent Cubes from the Intelligent Cube Monitor. For more information on managing Intelligent Cubes, see the System Administration Guide.

Publishing or re-publishing an Intelligent Cube can require significant memory resources. It is recommended that you use incremental refresh settings to add new data to the Intelligent Cube. For information on setting up incremental refreshes, see Updating Intelligent Cubes without re-processing: Incremental Refresh, page 45.

You can publish Intelligent Cubes manually, or you can schedule the publication of Intelligent Cubes for a specific time, as described in:

- Publishing Intelligent Cubes manually, page 40
- Publishing Intelligent Cubes using a schedule, page 41
- Unpublishing an Intelligent Cube, page 44

Publishing Intelligent Cubes manually

The procedure below provides the steps to manually publish an Intelligent Cube.

The act of publishing an Intelligent Cube can require memory resources approximately two times greater than the size of an Intelligent Cube. This can affect performance of your Intelligence Server as well as the ability to publish the Intelligent Cube. To plan for these memory requirements, see the System Administration Guide.

Prerequisites

- An Intelligent Cube has been created. For information on creating an Intelligent Cubes, see Creating Intelligent Cubes, page 26.
You need the Publish Intelligent Cube (Desktop) and/or Web Publish Intelligent Cube (Web) privileges. These privileges are part of OLAP Services.

To publish an Intelligent Cube manually

The steps below show you how to publish an Intelligent Cube in Desktop. You can follow the same high-level steps in Web to browse to an Intelligent Cube, and then run the Intelligent Cube to publish it.

1. In MicroStrategy Desktop, browse to the location of the Intelligent Cube to publish.

   If you define Intelligent Cubes to use the connection mapping configured for your project, the Intelligent Cubes are published with the connection mapping defined for the user account employed to publish the Intelligent Cubes. For information on defining Intelligent Cubes to support connection mapping, see the System Administration Guide.

2. Right-click the Intelligent Cube, and select Run.

   Rather than displaying report results, a message is displayed indicating that the Intelligent Cube has been published. After the Intelligent Cube is published, the Intelligent Cube appears in The Intelligent Cube Monitor, which you can use to manage your published Intelligent Cubes.

Publishing Intelligent Cubes using a schedule

As an alternative manually publishing Intelligent Cubes, you can schedule the publication of Intelligent Cubes to occur at a time when the load on your Intelligence Server and data warehouse is low. This allows you to publish Intelligent Cubes without affecting performance time of Intelligence Server during peak usage.

You can also schedule the publication of Intelligent Cubes to update Intelligent Cube data when events occur that would make the Intelligent Cube data outdated. For example, you can schedule to republish an Intelligent Cube when a database load occurs. The procedure below creates a subscription that can schedule multiple Intelligent Cubes to be updated. Alternatively, you can define an Intelligent Cube to be updated on a schedule during Intelligent Cube creation, which is described in Updating Intelligent Cubes using schedules, page 32.
You schedule an Intelligent Cube for publication by creating a subscription for the Intelligent Cube. The procedure below provides steps to create a subscription to schedule the publication of an Intelligent Cube.

You cannot create subscriptions for Intelligent Cubes that have been created using the Import Data feature in MicroStrategy Web. For information on the Import Data feature, refer to the MicroStrategy Web Help.

The act of publishing an Intelligent Cube can require memory resources approximately two times greater than the size of an Intelligent Cube. This can affect performance of your Intelligence Server as well as the ability to publish the Intelligent Cube. To plan for these memory requirements, see the System Administration Guide.

**Prerequisites**

- An Intelligent Cube has been created.
- You have created the schedule to subscribe the publication of an Intelligent Cube to. For information on creating schedules, see the System Administration Guide.

**To schedule the publication of Intelligent Cubes**

1. In MicroStrategy Desktop, log in to a project source using administrative privileges.

2. From the Administration menu, point to Scheduling, and then select Subscription Creation Wizard. The Subscription Creation Wizard opens.

3. If the welcome page is shown, review the information and click Next.

4. On the Subscription Wizard - Specify Characters page, begin defining the subscription using the following drop-down lists:

   - **Choose a schedule for this set of subscriptions**: Select a schedule to trigger the publication of Intelligent Cubes. A schedule can be time- or event-triggered.

   - **Choose a project from which reports/documents will be delivered to the recipients**: Select a project that contains Intelligent Cubes to schedule for publication. When publishing Intelligent Cubes,
the Intelligent Cubes are not delivered to a specific recipient, they are instead published to the Intelligent Cube Monitor.

- **Choose a Delivery Type**: Select **Cube Update**. This delivery type publishes Intelligent Cubes to the Intelligent Cube Monitor.

5 Click **Next**. The Subscription Wizard - Choose Reports/Documents page opens.

6 In the **Available objects** pane, browse to and select Intelligent Cubes to schedule for publication. Once selected, click the right arrow (>) to move any selected Intelligent Cubes to the **Selected objects** pane.

7 Click **Next**. The Subscription Wizard - Choose Recipients page opens.

8 Keep the default user that is selected or select a different user with administrative privileges, and click **Next**. The Subscription Wizard - Specify Subscription Properties page opens.

9 You can choose from the following subscription properties:

- **Run subscription immediately**: Select this check box if the Intelligent Cube should be published immediately. This causes any Intelligent Cubes included in the subscription to be published immediately. The Intelligent Cubes will also be re-published based on the schedule used in the subscription.

- **Expire subscription on**: Select this check box if the subscription should expire. If you select an expiration date, the subscription is deleted on that date and Intelligent Cubes included in the subscription are no longer scheduled for publication. The Intelligent Cubes can be published manually.

10 Click **Next**. The Subscription Wizard - Summary page opens.

11 Review the summary information, and click **Finish** to create the subscription. If you selected to run the subscription immediately, the Intelligent Cubes are executed against the data warehouse and published to the Intelligent Cube Monitor.

You can view and edit the new subscription from the Subscription Manager. For example, you can select a different schedule to use for the subscription or modify the expiration date for the subscription. For information on subscriptions and the Subscription Manager, see the *System Administration Guide*.
Unpublishing an Intelligent Cube

You can remove a published Intelligent Cube so that its data is no longer accessible by reports. This action does not delete the Intelligent Cube object saved in a MicroStrategy project, it only removes the Intelligent Cube from the Intelligent Cube Monitor.

Unpublishing an Intelligent Cube prevents reports that access an Intelligent Cube from being able to load the Intelligent Cube into Intelligence Server memory. This gives you more administrative control of when to make an Intelligent Cube available to reports. However, be aware that unpublishing an Intelligent Cube means that any reports that access the Intelligent Cube cannot be executed.

This same administrative control can be enforced by de-activating an Intelligent Cube. De-activating an Intelligent Cube saves the Intelligent Cube to secondary storage, such as a hard disk. When you re-activate the Intelligent Cube, the copy in secondary storage is loaded back into Intelligence Server memory. This option is ideal when an Intelligent Cube should not be used for some length of time, but after that should be available again in its current form. For information on de-activating an Intelligent Cube, see the System Administration Guide.

To unpublish an Intelligent Cube

1. In MicroStrategy Desktop, log in to a project source with administrative privileges.

2. From the Folder List, expand Administration, then expand System Monitors, then expand Caches, and select Intelligent Cubes. The Intelligent Cube Monitor is displayed.

3. Right-click an Intelligent Cube and select Delete. The Intelligent Cube is unpublished from the Intelligent Cube Monitor and its data cannot be accessed by reports.

Reporting and analyzing data with Intelligent Cubes

Intelligent Cube Technology allows reports to use OLAP Services features to display and analyze data using an in-memory copy of data rather than querying the data warehouse. For information on reporting on Intelligent Cubes, see Chapter 3, Reporting on Intelligent Cubes.
Updating Intelligent Cubes without re-processing: Incremental Refresh

By default, if the data in an Intelligent Cube needs to be updated, it is re-executed. All the data for the Intelligent Cube is loaded from the data warehouse into Intelligence Server's memory, and the existing data for the Intelligent Cube is overwritten.

However, if the Intelligent Cube is updated based on one or more attributes, such as Month or Region, you can set up incremental refresh settings to update the Intelligent Cube with only new data. This can reduce the time and system resources necessary to update the Intelligent Cube periodically.

For example, you have an Intelligent Cube that contains weekly sales data. At the end of every week, this Intelligent Cube must be updated with the sales data for that week. You can set up incremental refresh settings so that only data for one week is added to the Intelligent Cube.

Prerequisites

For an Intelligent Cube to qualify for an incremental refresh, it must meet the following requirements:

- The updates to the Intelligent Cube must be according to attributes only.
  
  For example, an Intelligent Cube that must be updated with data for a month or region qualifies for an incremental refresh.

- The Intelligent Cube must not be re-processed based on a metric.
  
  For example, an Intelligent Cube that contains data for the top 200 stores by Revenue does not qualify for an incremental refresh.

Defining an incremental refresh for an Intelligent Cube

Once you have defined an Intelligent Cube, you can use the following methods to define the incremental refresh options, depending on your requirements:

- Define Intelligent Cube republish settings. This is recommended if the Intelligent Cube must be updated along only one attribute, or if updates along multiple attributes must be made simultaneously.
To define Intelligent Cube republish settings, see *Defining Intelligent Cube refresh settings, page 46.*

- Define an incremental refresh filter or report. This is recommended if the Intelligent Cube must be updated along different dimensions at different times, or if you want to use the results of a report to update the Intelligent Cube.

To define an incremental refresh filter or report, see *Defining an incremental refresh filter or report, page 47.*

If you choose to define an incremental refresh filter or report, you should not re-publish the Intelligent Cube by double-clicking, or by publishing it on a schedule. Doing so will overwrite any changes made by the incremental refresh filter or report.

**Defining Intelligent Cube refresh settings**

The refresh settings change the way the Intelligent Cube is updated when you re-execute it manually, or when it is published on a schedule, as described in *Publishing Intelligent Cubes using a schedule, page 41.*

This is recommended if the Intelligent Cube must be updated along only one attribute, or if Intelligent Cube must be updated along multiple attributes simultaneously.

**Prerequisites**

- The Intelligent Cube’s definition must include a filter that qualifies on the attributes for which the Intelligent Cube must be updated.

For example, if the Intelligent Cube must be updated with new data for the Store attribute, the filter must qualify on Store.

---

**To define Intelligent Cube refresh settings**

1. In Desktop, navigate to the Intelligent Cube you want to define republish settings for.

2. Right-click on the Intelligent Cube, and select **Edit.** The Intelligent Cube Editor opens.

3. From the **Data** menu, select **Configure Intelligent Cube.** The Intelligent Cube Options dialog box opens.
4 Navigate to the Data Refresh category.

5 Choose one of the following options:

- **Full Refresh**: This is the default. The Intelligent Cube’s SQL is re-executed, and all the data is loaded from the data warehouse into Intelligence Server’s memory.

  Use this option under the following conditions:

  - If the data in the Intelligent Cube is outdated.
  
  - If the Intelligent Cube must be re-processed based on a metric. For example, an Intelligent Cube that contains data for the top 200 stores by Profit must be re-calculated every time it is updated, and thus must use the Full Refresh option.

- **Dynamic refresh**: The Intelligent Cube's filter is evaluated. If new data is returned, it is added to the Intelligent Cube, and data that no longer meets the filter's criteria is deleted from the Intelligent Cube.

  Select this option for Intelligent Cubes that have a rolling set of data—for example, an Intelligent Cube that always contains data for the past six months.

  If the data to be added and deleted cannot be determined, the **Full Refresh** option is used as a fallback.

- **Update**: The Intelligent Cube’s filter is evaluated. If new data is returned, it is added to the Intelligent Cube, and if the data returned is already in the Intelligent Cube, it is updated where applicable.

  Select this option if your data is updated often—for example, an Intelligent Cube that contains daily sales data, and is updated at the end of every day.

- **Insert**: The Intelligent Cube’s filter is evaluated. If new data is returned, it is added to the Intelligent Cube. Data that is already in the Intelligent Cube is not altered.

  Select this option if old data does not change once it is saved to your data warehouse.

**Defining an incremental refresh filter or report**

For complex update requirements, such as updating an Intelligent Cube for different dimensions at different intervals, you can define an incremental refresh filter or report to update your Intelligent Cube.
For example, an Intelligent Cube contains monthly sales data for 2009 and 2010 for year-on-year comparison. Once the year 2011 begins, you only need to keep the data for 2010, and the data for 2009 can be removed from the Intelligent Cube. You can define one incremental refresh that runs at the end of every month, and adds that month’s data to the Intelligent Cube, and a second incremental refresh that deletes the previous year’s data at the end of every year.

You can also add incremental refreshes to add data for other dimensions, such as Country. For example, you have an Intelligent Cube that contains data for the USA, UK and France, and you want to add data for Germany. You can define an incremental refresh that fetches all the data for Germany, and adds it to the cube.

You can define incremental refreshes using filters or reports, as described below:

- **Filter:** The data returned by a filter is compared to the data that is already in the cube. By default, the filter defined for the Intelligent Cube is used as the filter for the incremental refresh. For information on defining an incremental refresh filter, see *Defining an incremental refresh filter*, page 48.

- **Report:** The results of a report are used to populate the Intelligent Cube. By default, the report template used is the same as the Intelligent Cube’s template. For information on defining an incremental refresh report, see *Defining an incremental refresh report*, page 50.

If you choose to define an incremental refresh filter or report, you should not re-publish the Intelligent Cube by double-clicking it, or by publishing it on a schedule. Doing so will overwrite any changes made by the incremental refresh filter or report.

**Defining an incremental refresh filter**

You can define an incremental refresh filter to update the data in an Intelligent Cube. This is the default option for both ROLAP and MDX Intelligent Cubes, and is unavailable for Intelligent Cubes created using Freeform SQL queries or Query Builder.

**To define an incremental refresh filter**

1. In Desktop, navigate to the Intelligent Cube for which you want to define the incremental refresh.
2 Right-click the Intelligent Cube, and select **Define Incremental Refresh Report**. The Incremental Refresh Options dialog box opens.

3 Under **Refresh type**, select one of the following options. The differences between the options are illustrated in the image below.

- **Update**: The incremental refresh filter or report is evaluated. If new data is returned, it is added to the Intelligent Cube, and if the data returned is already in the Intelligent Cube, it is updated where applicable.

- **Insert**: The incremental refresh filter or report is evaluated. If new data is returned, it is added to the Intelligent Cube. Data that is already in the Intelligent Cube is not altered.

- **Delete**: The incremental refresh filter or report is evaluated. The data that is returned is deleted from the cube. For example, if the Intelligent Cube contains data for 2008, 2009 and 2010, and the filter or report returns data for 2009, all the data for 2009 is deleted from the cube.

- **Update only**: The incremental refresh filter or report is evaluated. If the data returned is already in the Intelligent Cube, it is updated where applicable. No new data is added to the Intelligent Cube.

You can change these options at any time by opening the incremental refresh in the Report Editor, and from the **Data** menu, selecting **Configure incremental refresh options**.
4 Click OK. The Report Editor opens with a new incremental refresh. If the Intelligent Cube’s definition included a filter, it appears in the Report Filter pane.

5 In the Report Filter pane, edit the filter if applicable, or create a new filter.

The filter must only qualify on attributes that are present in the Intelligent Cube.

6 To preview the data that will be updated in the Intelligent Cube, from the View menu, select Preview Data. The data is displayed in a grid view.

If your security filter prevents you from viewing some data, the preview only displays data that you can view. However, when the incremental refresh is executed, all the data is updated in the Intelligent Cube, regardless of security filters.

7 To execute the incremental refresh immediately, click the Run Report button.

8 To save and close the incremental refresh, click Save and Close.

**Defining an incremental refresh report**

You can define a report to update an Intelligent Cube. The results of a report are compared to the data in the Intelligent Cube, and the Intelligent Cube is updated accordingly.

If you are updating an Intelligent Cube based on a Freeform SQL or Query Builder report, this is the only available option.

**Prerequisites**

- The report must use all the attributes, and at least one metric from the Intelligent Cube that is being updated. Note that for metrics that are not on the report’s template, data is not updated.

- All attributes and metrics in the report’s definition should be identical to the attributes and metrics in the Intelligent Cube. You can verify this by checking the Global Unique Identifiers (GUIDs), by right-clicking the attribute or metric, and choosing Properties.
To define an incremental refresh report

1. Navigate to the Intelligent Cube for which you want to define the incremental refresh.

2. Right-click the Intelligent Cube, and select **Define Incremental Refresh Report**. The Incremental Refresh Options dialog box opens.

3. Under Refresh type, select one of the following options. The differences between the options are illustrated in the image below.

   - **Update**: The incremental refresh report is evaluated. If new data is returned, it is added to the Intelligent Cube, and if the data returned is already in the Intelligent Cube, it is updated where applicable.

   - **Insert**: The incremental refresh report is evaluated. If new data is returned, it is added to the Intelligent Cube. Data that is already in the Intelligent Cube is not altered.

   - **Delete**: The incremental refresh report is evaluated. The data that is returned is deleted from the cube. For example, if the Intelligent Cube contains data for 2008, 2009 and 2010, and the filter or report returns data for 2009, all the data for 2009 is deleted from the cube.

   If you are using the **Delete** option, you can use a subset of attributes from the Intelligent Cube on the report’s template.
• **Update only**: The incremental refresh report is evaluated. If the data returned is already in the Intelligent Cube, it is updated where applicable. No new data is added to the Intelligent Cube.

  You can change these options at any time by opening the incremental refresh in the Report Editor, and from the Data menu, selecting **Configure incremental refresh options**.

4 Navigate to the **Advanced** category.

5 Select **Report**, and click **OK**. The Report Editor opens, with a new incremental refresh report. By default, the report’s template contains all the attributes and metrics from the Intelligent Cube.

6 Edit the report, if necessary.

7 To preview the data that will be updated in the Intelligent Cube, from the View menu, select **Preview Data**. The data is displayed in a grid view.

  If your security filter prevents you from viewing some data, the preview only displays data that you can view. However, when the incremental refresh is executed, all the data is updated in the Intelligent Cube, regardless of security filters.

8 To execute the incremental refresh immediately, click the **Run Report** button.

9 To save and close the incremental refresh, click **Save and Close**.
REPORTING ON INTELLIGENT CUBES

Introduction

Intelligent Cube Technology allows reports to use OLAP Services features to report and analyze data using an in-memory copy of data rather than querying the data warehouse. This chapter discusses the various ways you can report on Intelligent Cubes:

- Reporting and analyzing data with personal Intelligent Cubes, page 53
- Reporting and analyzing data with Intelligent Cubes, page 54
- Reporting on Intelligent Cubes with dynamic sourcing, page 66

Reporting and analyzing data with personal Intelligent Cubes

While creating and viewing reports in MicroStrategy, you can use various OLAP Services features to report and analyze data using an in-memory copy of data, rather than querying the data warehouse. When OLAP Services
features are used on the fly in reports while performing full ROLAP analysis, the in-memory copy of data is referred to as a personal Intelligent Cube (see About personal Intelligent Cubes, page 16).

Personal Intelligent Cubes are created automatically for reports that utilize OLAP Services features. This provides both OLAP Services features and full ROLAP analysis on your report with no overhead to create or maintain an Intelligent Cube, or to link your report to an Intelligent Cube.

As you build and modify your report, the personal Intelligent Cube is updated to reflect you ROLAP analysis. While this provides full ROLAP analysis along with OLAP Services analysis, this can require requerying the data warehouse and generating new report SQL. Also, the OLAP Services feature known as derived elements is unavailable for reports that use personal Intelligent Cubes. For a full comparison of personal Intelligent Cubes and Intelligent Cubes, see About personal Intelligent Cubes, page 16.

Along with all of the standard reporting features available in MicroStrategy, the OLAP Services features listed below can be used when reporting on personal Intelligent Cubes:

- Chapter 5, Dynamic Aggregation
- Chapter 6, Derived Metrics
- Chapter 7, View Filters

The chapters listed above describe how to use various OLAP Services features with reports. For information on how to create reports and use all the standard reporting features available in MicroStrategy, see the Basic Reporting Guide and the Advanced Reporting Guide.

### Reporting and analyzing data with Intelligent Cubes

You can perform all of your reporting and data analysis within an Intelligent Cube, without executing any queries against the data warehouse. Once an Intelligent Cube is created and then published so its data is available, multiple reports can be created to connect to the Intelligent Cube (see Sharing Intelligent Cubes, page 19).
Reporting on an Intelligent Cube provides quick access to data, as the data has been pre-aggregated. This returns results much faster than querying the data warehouse. Reporting on Intelligent Cubes also allows you to use all of the OLAP Services features, including derived elements, which allow you to group attribute elements in a report on the fly, to provide a new view of report data for analysis and formatting (see *Chapter 4, Derived Elements*).

Reports that connect to an Intelligent Cube are restricted to only the data available within the Intelligent Cube. This ensures that report results are returned quickly, and it also prevents full ROLAP analysis. Report filters, consolidations, and custom groups are not available for reports that connect to Intelligent Cubes. For a full comparison of personal Intelligent Cubes and Intelligent Cubes, see *Chapter 2, Sharing Sets of Data Among Reports: Intelligent Cubes*.

While reporting on Intelligent Cubes, there are a few scenarios that can produce unexpected results. To troubleshoot these issues, see *Troubleshooting reports connected to Intelligent Cubes, page 63*.

This section discusses the features available for reports that connect to Intelligent Cubes, and how the reports utilize standard reporting features and OLAP Services features to execute reporting and analysis manipulations completely within the Intelligent Cube. This section includes information on the small differences in workflows and standards when using standard reporting features in these reports:

- *Creating reports that connect to Intelligent Cubes, page 55*
- *Analyzing data using standard OLAP Services features, page 58*
- *Run-time reporting with prompts, page 58*
- *Relational analysis with drilling, page 60*
- *Troubleshooting reports connected to Intelligent Cubes, page 63*

**Creating reports that connect to Intelligent Cubes**

To create a report that connects to an Intelligent Cube, you must choose an Intelligent Cube to connect to when creating the report. The procedures below describe how to create a report that connects to an Intelligent Cube:

- *Creating reports that connect to Intelligent Cubes, in Desktop, page 56*
- *Creating reports that connect to Intelligent Cubes, in Web, page 56*
Creating reports that connect to Intelligent Cubes, in Desktop

Prerequisites

- An Intelligent Cube has been created and published.
- You need the Define Intelligent Cube Report privilege. This privilege is part of OLAP Services.

To create a report that connects to an Intelligent Cube, in MicroStrategy Desktop

1. In Desktop, log into a project.
2. From the File menu, point to New, and then select Report. The New Grid dialog box opens.
3. Select the Intelligent Cubes tab.
4. Browse to an Intelligent Cube, select it, and then click OK.

The Report Editor opens with all of the objects of the Intelligent Cube included in the Report Objects pane on the left. You can begin to create your report.

In Desktop, you can also create a report that connects to an Intelligent Cube by right-clicking an Intelligent Cube and selecting Create Report.

Creating reports that connect to Intelligent Cubes, in Web

Prerequisites

- An Intelligent Cube has been created and published.
- You need the Web Define Intelligent Cube Report privilege. This privilege is part of OLAP Services.

To create a report that connects to an Intelligent Cube, in MicroStrategy Web

1. In MicroStrategy Web, log in to a project.
2 Click **Create Report**. A page of report creation options is displayed.

3 Click **Intelligent Cube Report**. The Select Intelligent Cube dialog box opens.

4 Browse to an Intelligent Cube, select it, and then click **OK**.

The report opens with all of the objects of the Intelligent Cube included in the Report Objects pane on the left. You can begin to create your report.

### Connecting to a different Intelligent Cube

You can modify the Intelligent Cube a report connects to for its data requirements. This allows you to switch to an Intelligent Cube that fits your reporting requirements without having to create a new report.

However, the Intelligent Cube you switch to for the report should have the same or at least similar data as was previously available in the report. If the report contained some objects that are not in the Intelligent Cube you switch to, this can cause the objects to be unavailable for the report or have other unintended functionality. A warning message is displayed for any mismatch in data between the report and the Intelligent Cube you want to link the report to.

Changing the Intelligent Cube that a report points to is possible in Desktop only.

### Prerequisites

- A report has been created that connects to a published Intelligent Cube.
- A second Intelligent Cube has been created and published that has the same or similar data as the report you wish to link to it.
- You need the Define Intelligent Cube Report privilege. This privilege is part of OLAP Services.

### To connect to a different Intelligent Cube

1 In Desktop, browse to a report that is connected to an Intelligent Cube, right-click the report, and select **Edit**. The report opens in the Report Editor.
2 From the **Data** menu, point to **Intelligent Cube Options**, and then select **Point grid to Intelligent Cube**. The Select a cube dialog box opens.

3 Browse to the new Intelligent Cube you want to link the report to, select it, and then click **Open**.

4 If there are data mismatches between the report and the Intelligent Cube you are attempting to link it to, a warning message is displayed that lists the differences. Attributes on the report that are not contained in the new Intelligent Cube will have to be removed from the report to display valid report results. Missing metrics are converted into derived metrics and may be able to display correct data, but they should be removed if no data is returned.

   If the differences are acceptable, click **OK**.

5 The report is updated to connect to the Intelligent Cube selected. You should remove any attributes and metrics if they no longer return correct results.

### Analyzing data using standard OLAP Services features

Reports that connect to Intelligent Cubes make use of OLAP Services features to provide a broad range of analysis capabilities. All OLAP Services analysis is executed against the Intelligent Cube. For more information on the different OLAP Services features available for reports that connect to Intelligent Cubes, see the following chapters in this guide:

- **Chapter 4, Derived Elements**
- **Chapter 5, Dynamic Aggregation**
- **Chapter 6, Derived Metrics**
- **Chapter 7, View Filters**

### Run-time reporting with prompts

In addition to including base attributes, metrics, and other objects to be displayed on your reports, you can also determine how much of the report to display at run time. Prompts allow users to choose which objects and filtering criteria to apply to a report during report execution.
Prompts serve the same purpose in any report, including reports that connect to Intelligent Cubes. However, instead of modifying SQL at report run time, prompts allow reports to select data within the Intelligent Cube, as illustrated below.

![Image showing standard run-time reporting with prompts, while using OLAP Services to execute against the Intelligent Cube rather than against the data warehouse.](image)

The image above shows standard run-time reporting with prompts, while using OLAP Services to execute against the Intelligent Cube rather than against the data warehouse. The performance of your business intelligence application is improved by reducing execution against your data warehouse and maintaining only a single Intelligent Cube for multiple prompted reports.

Prompts on reports that connect to Intelligent Cubes can only access data that is available within the Intelligent Cube. These restrictions are applied automatically when creating prompts. For example, the attributes Year and Region and the metrics Cost and Revenue are included in the Intelligent Cube shown in the image above. If you create an object prompt in your report that connects to this Intelligent Cube, then you can only create prompts based on one of Year, Region, Cost, and Revenue.

You cannot use prompts that include objects or data that are not part of the Intelligent Cube, or prompts that use hierarchies. If you try to use such prompts, an error message is displayed.

You can provide full ROLAP analysis with prompts and OLAP Services by using personal Intelligent Cubes. Prompts in reports that access personal Intelligent Cubes can use the complete ROLAP schema of a project. However, if a prompt retrieves data from outside the personal Intelligent Cube, re-execution against the data warehouse is necessary. For more information on personal Intelligent Cubes, see "About personal Intelligent Cubes."
Relational analysis with drilling

As with any MicroStrategy report, you can drill on reports that connect to Intelligent Cubes to analyze data at different logical levels.

A report connected to an Intelligent Cube can drill within the data available in the Intelligent Cube it is connected to. This means that you can drill from an attribute on the report grid to an attribute that is not on the report grid, but available in the Report Objects pane. If the attribute is not available in the Report Objects pane, it is not an available drilling option by default. However, Intelligent Cubes can be defined to allow drilling outside of an Intelligent Cube to the full relational data warehouse.

For example, your report includes the attribute Year. After analyzing data at the Year level, you want to analyze data for each quarter. You can drill down from Year to the attribute Quarter to view and analyze data at the new logical level. This drilling action is performed within an Intelligent Cube. Following this example scenario, you want to drill from year 2007 to quarters for that year. You have a report connected to an Intelligent Cube that is defined as shown below:

Notice that Quarter is not on the report, but it is included in the Report Objects pane on the left as it is a part of the Intelligent Cube that the report is connected to. As shown in the report above, you right-click the 2007
attribute element for Year and drill down to Quarter. The drilled-to report is shown below.

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit Margin</th>
<th>Profit</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007 Q1</td>
<td>Electronics</td>
<td>Books</td>
<td>24.53%</td>
<td>$85,473</td>
<td>$348,483</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
<td>9.72%</td>
<td>$29,123</td>
<td>$299,531</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
<td>9.69%</td>
<td>$20,590</td>
<td>$212,438</td>
</tr>
<tr>
<td>2007 Q2</td>
<td>Electronics</td>
<td>Books</td>
<td>21.91%</td>
<td>$84,984</td>
<td>$387,849</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
<td>6.61%</td>
<td>$21,578</td>
<td>$326,270</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
<td>6.62%</td>
<td>$15,112</td>
<td>$223,289</td>
</tr>
<tr>
<td>2007 Q3</td>
<td>Electronics</td>
<td>Books</td>
<td>22.24%</td>
<td>$90,618</td>
<td>$407,392</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
<td>7.06%</td>
<td>$23,580</td>
<td>$334,143</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
<td>7.02%</td>
<td>$16,797</td>
<td>$239,112</td>
</tr>
<tr>
<td>2007 Q4</td>
<td>Electronics</td>
<td>Books</td>
<td>19.90%</td>
<td>$83,486</td>
<td>$419,563</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
<td>4.64%</td>
<td>$17,321</td>
<td>$373,182</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
<td>4.68%</td>
<td>$12,182</td>
<td>$260,298</td>
</tr>
</tbody>
</table>

This drilled-to report is executed within and connected to the same Intelligent Cube as the original report. This is verifiable by looking at the Report Objects pane, which shows that the report objects are being returned from the same Intelligent Cube (named Drilling I Cube) as the original report. This provides relational analysis without having to execute the report against the data warehouse.

In the scenario above, drilling is performed within the Intelligent Cube, which is achievable through any report connected to an Intelligent Cube. However, if the Intelligent Cube is defined to allow drilling outside it (see *Enabling ROLAP drilling for reports accessing Intelligent Cubes, page 33*), you can also drill to any object not included in the Intelligent Cube. While drilling outside of an Intelligent Cube requires execution against the data warehouse, it provides access to the full ROLAP schema of the project outside of the Intelligent Cube.

In the next example, the same scenario of drilling from Year to Quarter is used, except that the Intelligent Cube does not contain the Quarter attribute.
Reporting on Intelligent Cubes OLAP Services Guide

As shown in the report below, you right-click the 2007 attribute element for Year and drill down to Quarter.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit Margin</th>
<th>Profit</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Books</td>
<td>22.04%</td>
<td>$251,345</td>
<td>$1,140,363</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>17.43%</td>
<td>$2,760,984</td>
<td>$15,083,893</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>6.69%</td>
<td>$55,041</td>
<td>$823,228</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>6.90%</td>
<td>$43,243</td>
<td>$625,592</td>
<td></td>
</tr>
<tr>
<td>2006</td>
<td>Books</td>
<td>22.32%</td>
<td>$294,789</td>
<td>$1,320,585</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>17.56%</td>
<td>$3,389,108</td>
<td>$19,299,871</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>6.88%</td>
<td>$71,067</td>
<td>$1,032,391</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>7.17%</td>
<td>$53,666</td>
<td>$743,966</td>
<td></td>
</tr>
</tbody>
</table>

Notice in the report shown above that all the attributes in the Time hierarchy are available drilling options even though they are not all included in the Intelligent Cube. These attributes are available drilling options because the Intelligent Cube is defined to enable drilling outside of the Intelligent Cube. As shown in the report above, you right-click the 2007 attribute element for Year and drill down to Quarter. The drilled-to report is shown below.

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit Margin</th>
<th>Profit</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007 Q1</td>
<td>Books</td>
<td>24.53%</td>
<td>$85,473</td>
<td>$348,483</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>20.04%</td>
<td>$1,084,237</td>
<td>$5,411,499</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>9.72%</td>
<td>$29,123</td>
<td>$299,531</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>9.69%</td>
<td>$20,590</td>
<td>$212,438</td>
<td></td>
</tr>
<tr>
<td>2007 Q2</td>
<td>Books</td>
<td>21.91%</td>
<td>$84,984</td>
<td>$387,849</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>17.21%</td>
<td>$983,376</td>
<td>$5,714,783</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>6.61%</td>
<td>$21,570</td>
<td>$325,270</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>6.62%</td>
<td>$15,112</td>
<td>$228,299</td>
<td></td>
</tr>
<tr>
<td>2007 Q3</td>
<td>Books</td>
<td>22.24%</td>
<td>$90,618</td>
<td>$407,392</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>17.62%</td>
<td>$1,057,109</td>
<td>$5,999,174</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>7.06%</td>
<td>$23,580</td>
<td>$334,143</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>7.02%</td>
<td>$16,797</td>
<td>$239,112</td>
<td></td>
</tr>
<tr>
<td>2007 Q4</td>
<td>Books</td>
<td>19.90%</td>
<td>$83,486</td>
<td>$419,563</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>15.39%</td>
<td>$1,004,932</td>
<td>$6,529,576</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>4.64%</td>
<td>$17,321</td>
<td>$373,182</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>4.68%</td>
<td>$12,182</td>
<td>$260,298</td>
<td></td>
</tr>
</tbody>
</table>
This drilled-to report is executed against the data warehouse, and it allows you to access data outside of the Intelligent Cube for further relational analysis. Notice also that all report objects that were not on the report grid are now removed from the Report Objects pane, because this new, drilled-to report is not connected to the Intelligent Cube.

You should consider the execution time requirements for a report before drilling outside of an Intelligent Cube.

For basics on how to drill on data, see the *Basic Reporting Guide*. For information on creating drill maps, which are used to enable drilling techniques, see *Chapter 6, Drill Maps* in the *Advanced Reporting Guide*.

**Troubleshooting reports connected to Intelligent Cubes**

While reporting on Intelligent Cubes, a few scenarios can produce unexpected results. To troubleshoot these issues, review the following sections:

- *No data is returned for the report*, page 63
- *No data is returned for a metric or metrics on the report*, page 64
- *The report fails due to the unavailability of Intelligent Cubes*, page 65

**No data is returned for the report**

If no data is returned for a report that is connected to an Intelligent Cube, this can be caused by the following scenarios:

- The view filter for the report is too restrictive (see *Chapter 7, View Filters*). Modify or remove the view filter to attempt to return data for the report. If removing the view filter does not allow the report to return data, the cause may be due to your security filter, as described below.

- Your security filter does not allow you to see the data available on the Intelligent Cube. For information on how to resolve data availability due to security filter resolution, see *Resolving security filter resolution* below.

**Resolving security filter resolution**

If security filters are used to maintain data access security, this can cause data to not be displayed for a report connected to an Intelligent Cube.
While this may be due to normal data access security, the data available on the Intelligent Cube may prevent you from viewing all possible data in a data source (see *Security filter resolution for reports connected to Intelligent Cubes, page 22*). To verify whether there is additional data that is not being returned, you can create or view a report that connects directly to a data source rather than an Intelligent Cube. You can also contact the designers of the Intelligent Cube to review whether additional data can be added to the Intelligent Cube so that it supports your report.

### No data is returned for a metric or metrics on the report

If no data is returned for a metric or metrics on a report connected to an Intelligent Cube, this can be caused by either dynamic sourcing or your security filter.

When this is caused by your security filter resolution, no data is displayed for the metric or metrics. However, when this is caused by dynamic aggregation, null values are displayed for the metric rather than no information at all. The image below shows the difference between security filter resolution and dynamic aggregation as the cause for metrics not displaying any data.

By default, null values are represented by dashes (--) on reports. For information on changing the display of null values, see *Changing the display of null values, page 160*.

If dynamic aggregation is the cause for null values being displayed, resolutions for this issue are described in *Metrics that are not dynamically aggregated by default, page 152*. 
If your security filter is the cause for metric data not being displayed, see *Resolving security filter resolution, page 63.*

**The report fails due to the unavailability of Intelligent Cubes**

An unavailable Intelligent Cube can cause your reports connected to this Intelligent Cube to fail. This is because the report depends on the Intelligent Cube to provide the data for the report.

The scenarios listed below cause reports to return error messages when Intelligent Cubes are unavailable:

- The Intelligent Cube is not published. This scenario can occur when an Intelligent Cube is removed to:
  - Update it with new data available in the data warehouse.
  - Release its resources for more frequently used Intelligent Cubes.
  - Provide space for a new Intelligent Cube because Intelligent Cube storage limits have been exceeded.

  Contact your administrator to determine why an Intelligent Cube is no longer published and whether there are plans to publish it again.

- The Intelligent Cube is currently offline. In this scenario, the Intelligent Cube is published but has been made unavailable for reporting. Contact the administrator of Intelligent Cubes to determine why the Intelligent Cube has been made unavailable, as well as if and when the Intelligent Cube will be made available again.

- The Intelligent Cube is being published, but the publishing process is not finished yet. This scenario can occur when an Intelligent Cube is republished to return up-to-date information from the data warehouse. If you wait for the Intelligent Cube to publish, you should then be able to run your report without experiencing this error.

To avoid this error, ask the administrator of Intelligent Cubes when Intelligent Cubes are published and updated to reflect up-to-date data. This information can help you avoid accessing Intelligent Cubes during times of planned maintenance.
Reporting on Intelligent Cubes with dynamic sourcing

You can manually connect a report to an Intelligent Cube to perform all reporting and analysis within the shared in-memory copy of data, as described in Reporting and analyzing data with Intelligent Cubes, page 54. Even if you don’t manually connect your reports to an Intelligent Cube, the reports you create can also access Intelligent Cube data rather than querying the data warehouse automatically. Enabling this automatic link between reports and Intelligent Cubes is referred to as dynamic sourcing.

Dynamic sourcing extends the accessibility of Intelligent Cubes by allowing regular reports to access published Intelligent Cubes, as long as the Intelligent Cubes can satisfy the requirements of the report. For a detailed explanation of dynamic sourcing and procedures to perform the project-wide configurations necessary to enable and support dynamic sourcing, see Chapter 5, Dynamic Aggregation.

While reporting on Intelligent Cubes, there are a few scenarios that can produce unexpected results. To troubleshoot these issues, see Troubleshooting reports connected to Intelligent Cubes, page 63

Supporting report execution through dynamic sourcing

When creating reports, you can enable and increase your chances that your report utilizes dynamic sourcing by following the recommendations below.

First, you must enable dynamic sourcing for your report. This can add some overhead to the execution of your report to check whether there are any Intelligent Cubes that meet the data requirements of your report. This overhead usually has no effect on performance. If an Intelligent Cube can be used, the report results can be returned quickly from the in-memory copy of data rather than querying the data warehouse. To enable dynamic sourcing for reports, see Enabling or disabling dynamic sourcing for reports, page 248.

The simpler your report is, the more likely it is that an Intelligent Cube satisfies its data requirements. This does not mean that you should create reports with minimal data simply to take advantage of dynamic sourcing.
However, the considerations listed below can help you create a report that meets your reporting requirements and is well-suited for dynamic sourcing:

- Avoid the use of features that prevent the use of dynamic sourcing, which are described in *Features that prevent the use of dynamic sourcing, page 241.*

- Only add objects that are relevant to the analysis required for the report. When creating a report, you should determine whether certain attributes, metrics, advanced filtering techniques, and so on are required for the report or are extraneous to the analysis requirements of a report.
Introduction

A derived element is a grouping of attribute elements on a report. Derived element groups are defined by a list, filter, or calculation. These groups provide a new view of report data for analysis and formatting purposes.

For example, the report on the left, below, contains Region, Category, and Profit. The report on the left does not display any derived elements. The report on the right displays derived elements defined using groups of Region attribute elements:

- East Coast: This derived element combines the data for the Mid-Atlantic, Northeast, and Southeast regions.
- West Coast: This derived element combines the data for the Northwest and Southwest regions.
- Central and South: This derived element combines the data for the Central and South regions.
• Web: This derived element displays the data for the Web attribute element, which is not included in any of the derived elements listed above.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$16,347</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$46,461</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>Books</td>
<td>$455,964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,500,514</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,367</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$25,895</td>
<td></td>
</tr>
<tr>
<td>Northwest</td>
<td>Books</td>
<td>$19,734</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,604,110</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$5,639</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$5,508</td>
<td></td>
</tr>
<tr>
<td>South</td>
<td>Books</td>
<td>$216,663</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$739,064</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$12,905</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,882</td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td>Books</td>
<td>$25,609</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$419,847</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$37,704</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$6,965</td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td>Books</td>
<td>$41,575</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$686,421</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,294</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

You can do more than just simple combinations of attribute elements with derived elements. For example, after you have defined the East Coast derived
element, you can determine the East Coast region’s percent contribution to profit, as shown in the last row of the report below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td></td>
<td>$528,034</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$5,853,393</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$79,423</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$46,597</td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td></td>
<td>$61,310</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$2,290,530</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$66,426</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$16,803</td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td></td>
<td>$272,158</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$1,674,186</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$29,251</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$90,238</td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td></td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$469,635</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$42,611</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$7,952</td>
</tr>
<tr>
<td>East Coast’s % Contribution</td>
<td>Books</td>
<td></td>
<td>59.28%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>56.90%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>36.48%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>28.84%</td>
</tr>
</tbody>
</table>

This demonstrates only a fraction of the analysis and formatting capabilities of derived elements. With derived elements you can also create custom sort orders for attribute elements, use aggregation functions such as Average to combine attribute elements, and perform other analysis and formatting tasks.

Derived elements are evaluated on the report dataset without regenerating or re-executing SQL.

The following sections of this chapter cover derived element concepts, functionality, and procedures:

- *Types of derived elements, page 72*
- *Creating derived elements, page 81*
- *Defining derived element functionality and formatting, page 122*
- *Interaction with other reporting features, page 134*
Types of derived elements

The different types of derived elements provide a wide range of reporting and analysis features. The following sections describe each type of derived element and how they can be used for reporting and analysis requirements:

- *Group derived element, page 72*
- *Filter derived element, page 74*
- *Calculation derived element, page 77*
- *All Other derived element, page 79*

**Group derived element**

A Group derived element is a combination of attribute elements into a single derived element. All attribute element data is added together to form the metric values for the new Group derived element.

For example, you have a report with Region, Category, and Profit displayed on a report. The report shown below includes the following derived elements defined using groups of Region attribute elements:

- **East Coast**: Groups the Mid-Atlantic, Northeast, and Southeast attribute elements.
- **West Coast**: Groups the Northwest and Southwest attribute elements.
• Central and South: Groups the Central and South attribute elements.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td>$523,034</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$5,853,393</td>
<td>$79,423</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$45,597</td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td>$61,310</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,290,530</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$66,426</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$15,803</td>
<td></td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td>$272,158</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,674,186</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$29,251</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$90,230</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

A procedure to create the report shown above is provided in *Grouping attribute elements to create a derived element, page 102*.

Group derived elements are created by selecting attribute elements to include in each derived element. The image below shows how the East Coast Group derived element is created in the Derived Elements Editor:

Group derived elements can only combine attribute elements, they cannot combine other derived elements. If you want to create a derived element that is a combination of other derived elements, you
must use a Calculation derived element (see Calculation derived element, page 77).

You can quickly create Group derived elements using right-click options (see Creating quick groups, page 83), or you can use the Derived Elements Editor to access the full functionality of derived elements (see Using the Derived Elements Editor, page 97).

You can also use this type of derived element to display the attribute elements in a different order. This enables you to do more advanced attribute element sorting than simple ascending or descending sorts. For information on using Group derived elements to sort the display of attribute elements on a report, see Creating quick sorts, page 93.

Filter derived element

A Filter derived element uses a filter qualification to determine the combination of attribute elements for a derived element.

For example, in a report with Region and Category attributes and a Profit metric, you can filter the regions on the report into various geographical groups based on the region names.

The report shown below includes the following derived elements defined using filters for Region attribute elements:

- Southern Regions: Returns attribute elements whose name begins with South.
• Northern Regions: Returns attribute elements whose name begins with North.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Southern Regions</td>
<td>Books</td>
<td>$283,848</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,845,331</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$111,396</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$30,141</td>
</tr>
<tr>
<td>Northern Regions</td>
<td>Books</td>
<td>$475,697</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,104,624</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$33,006</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$31,404</td>
</tr>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$15,347</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$73,357</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$45,461</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
</tr>
</tbody>
</table>

A procedure to create the report shown above is provided in *Filtering attribute elements to create a derived element, page 107.*

Filter derived elements are created by filtering attribute elements to include in each derived element. There are two methods to create Filter derived elements:

• Create a Filter derived element using a filter qualification on a list of attribute elements. This includes using the In list and Not in List operators.

  □ In list: A filter qualification using In list returns data for all the attribute elements you select. An In list filter qualification that returns all the southern regions is shown below.
Not in List: A filter qualification using Not in List returns data for all the attribute elements you do not select, for a given attribute. A Not in List filter qualification that returns all the southern regions is shown below.

Create a Filter derived element using a filter qualification on attribute forms. This enables you to use various logical and mathematical operators to create filter qualifications on attribute forms to return data. An attribute form qualification using a Begins with operator that returns all southern regions is shown below.

You can use any of the following operators in attribute form qualifications, which are described in detail in Appendix A, Logical and Mathematical Operators for Filtering in the MicroStrategy Advanced Reporting Guide:

<table>
<thead>
<tr>
<th>Operator Type</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison Operators</td>
<td>Exactly</td>
</tr>
<tr>
<td></td>
<td>Different from</td>
</tr>
<tr>
<td></td>
<td>Greater than</td>
</tr>
<tr>
<td></td>
<td>Less than</td>
</tr>
<tr>
<td></td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td></td>
<td>Less than or equal to</td>
</tr>
<tr>
<td></td>
<td>Between</td>
</tr>
</tbody>
</table>
To create Filter derived elements, you must use the Derived Elements Editor (see *Using the Derived Elements Editor, page 97*).

**Calculation derived element**

A Calculation derived element uses operators and functions to combine attribute elements and derived elements into calculations that define a single derived element.
For example, in a report with Region and Category attributes and a Profit metric, you can combine the regions on the report into various groups for profit analysis, as shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Profit</td>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
<tr>
<td>Average Profit</td>
<td>Books</td>
<td>$111,337</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,285,968</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,214</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$20,199</td>
<td></td>
</tr>
<tr>
<td>Greatest Regional Profit</td>
<td>Books</td>
<td>$455,964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
<tr>
<td>Greatest Regional Profit % Contribution</td>
<td>Books</td>
<td>51.19%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>38.23%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>27.92%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>48.49%</td>
<td></td>
</tr>
</tbody>
</table>

A procedure to create the report shown above is provided in *Using calculations to create derived elements, page 111.*

Calculation derived elements are created by defining expressions with valid combinations of operators, functions, attribute elements, and derived elements. An example of a valid expression is shown below.

You can include the following when you create a Calculation derived element expression:

- Attribute elements: You can include attribute elements in your expression by selecting the attribute from the drop-down list, selecting
attribute elements, and dragging and dropping them into the expression area.

- **Derived elements:** You can include other derived elements in your expression by selecting derived elements from the Groups drop-down list, and dragging and dropping them into the expression area. The example below shows a Calculation derived element created by performing a division of two other derived elements.

![Image showing Derived Elements Editor]

- **Operators:** You can include ( ), +, -, *, and /, which are all available on the toolbar.

- **Functions:** You can include Average, Greatest, and Least by clicking $f(x)$ on the toolbar and completing the Insert Function Wizard. For information on these functions, see *Creating quick calculations, page 86*. For steps to use the Insert Function Wizard, click Help in the wizard.

- **Clear:** You can clear the expression to start creating a new expression.

- **Validate:** You can check your expression to see if its syntax is valid. Any errors in syntax are highlighted in red.

To create Calculation derived elements, you can either quickly create derived elements with right-click options (see *Creating quick calculations, page 86*), or you can use the Derived Elements Editor to access the full functionality of derived elements (see *Using the Derived Elements Editor, page 97*).

### All Other derived element

The All Other derived element collects all attribute elements that are not included in other derived elements, and displays them on the report as
individual attribute elements by default. This derived element is created automatically when the first derived element is created for an attribute. An All Other derived element can only be deleted by deleting all derived elements.

Because it is created automatically, the All Other derived element is created and implemented without having to take note of it at all. For example, recall the report created with a Filter derived element (see Filter derived element, page 74) that included Southern Regions and Northern Regions derived elements as shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Southern Regions</td>
<td>Books</td>
<td>$283,848</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,845,331</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$111,396</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$30,141</td>
<td></td>
</tr>
<tr>
<td>Northern Regions</td>
<td>Books</td>
<td>$475,697</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,104,624</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$33,006</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$31,404</td>
<td></td>
</tr>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$15,347</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$45,461</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

Central, Mid-Atlantic, and Web are all attribute elements that are not included in any derived elements. The All Other derived element gathers these remaining attribute elements, and displays them on a report as individual attribute elements.

You can also include all of the attribute elements that are part of the All Other derived element as one consolidated element on a report (see Displaying derived elements or their attribute elements, page 125).

While this is the most common way the All Other derived element is used, you can define attribute elements included in other derived elements to also be included as part of the All Other derived element. For information on including attribute elements in the All Other derived element, see Displaying derived elements and their attribute elements simultaneously, page 127.
Creating derived elements

Derived elements require static report results so that they can be evaluated without regenerating or re-executing SQL. Due to this requirement, derived elements can only be created in the following types of reports and documents:

- Intelligent Cube reports: An Intelligent Cube report is a report that is connected to and retrieves its data from an Intelligent Cube.

- Grid/Graphs in Report Services documents: A Grid/Graph is a control placed in a document that displays information in the same way a MicroStrategy report does. In other words, a Grid/Graph is a report that has been added to a section of a Report Services document.

You can create derived elements in Grid/Graphs regardless of whether its displayed results come from an Intelligent Cube, directly from a data warehouse, or from some other type of data storage system.

If a report containing derived elements is included as a dataset of a Report Services document, derived elements are displayed for any associated attributes added to the details or grouping sections. However, you cannot create or modify derived elements for attributes added to the grouping or details sections.

You can create derived elements with the following methods, which are all described in this section:

- **Quickly creating groups, calculations, and sorts, page 82**: While reviewing the data on a report or document, you can quickly group attribute elements into derived elements for further analysis of your data.

- **Using the Derived Elements Editor, page 97**: Using the Derived Elements Editor provides the full set of derived elements functionality when creating derived elements.

- **Creating and using stand-alone derived elements, page 117**: You can create stand-alone derived elements that can be used in multiple reports and Grid/Graphs. You use the Derived Elements Editor to create stand-alone derived elements, but you do not have to create them from within a report or Grid/Graph.
Quickly creating groups, calculations, and sorts

While reviewing the results on a report or document, you can quickly group attribute elements into derived elements for further analysis of your data.

You can quickly create derived elements based on attribute elements and other derived elements in reports and Grid/Graphs using right-click options. You can create the following types of quick group derived elements:

- **Creating quick groups, page 83**: Creates a simple group of attribute elements.
- **Creating quick calculations, page 86**: Creates a calculation on the attribute elements or derived elements (or a combination of both).
- **Creating quick sorts, page 93**: Creates a derived element that sorts the attribute elements on the report or document in any order you wish. This option is only available if no derived elements are defined for the attribute on the report or document.

These quick group options to create derived elements are quick and easy ways to create derived elements. However, creating derived elements with these quick group techniques is only available with the configurations listed in the table below:

<table>
<thead>
<tr>
<th>Report Object Type</th>
<th>View/Mode</th>
<th>MicroStrategy Desktop/Web</th>
<th>Derived Element Quick Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligent Cube report</td>
<td>Grid View</td>
<td>Desktop and Web</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Grid and Graph View</td>
<td>Desktop and Web</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Graph View</td>
<td>Desktop and Web</td>
<td>No</td>
</tr>
</tbody>
</table>
Creating quick groups

Using right-click options, you can group attribute elements into a single derived element. The derived element created by this action aggregates and displays all the data for the attribute elements selected, into one set of data. For information on Group derived elements, see Group derived element, page 72.

For example, you have a report that displays the number of sales orders, sales order items, and sales order amounts for customer industry sectors. Your report displays this information over a number of months. To get a more summarized view of the data, you decide to group months into seasons.
You group October and November 2008 into a single derived element called Fall 08, by selecting the attribute elements, right-clicking the selection, and selecting Create Group. This process is shown below.

The example report below was created in the Sales and Distribution Analysis Module.

<table>
<thead>
<tr>
<th>Month</th>
<th>Customer Industry Sector</th>
<th>Metrics</th>
<th>Sales Orders</th>
<th>Sales Order Items</th>
<th>Net Sales Order Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 2008</td>
<td>Banking and Insurance</td>
<td></td>
<td>1</td>
<td>3</td>
<td>36000</td>
</tr>
<tr>
<td></td>
<td>Telecomunications</td>
<td></td>
<td>1</td>
<td>2</td>
<td>245000</td>
</tr>
<tr>
<td>Nov 2008</td>
<td>Retail and Consumer Products</td>
<td></td>
<td>2</td>
<td>11</td>
<td>12038000</td>
</tr>
<tr>
<td>Jan 2009</td>
<td>Retail and Consumer Products</td>
<td></td>
<td>2</td>
<td>4</td>
<td>3310000</td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td></td>
<td>1</td>
<td>4</td>
<td>4750000</td>
</tr>
<tr>
<td></td>
<td>Telecomunications</td>
<td></td>
<td>3</td>
<td>14</td>
<td>719000</td>
</tr>
<tr>
<td></td>
<td>e-business</td>
<td></td>
<td>1</td>
<td>3</td>
<td>216000</td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td></td>
<td>1</td>
<td>2</td>
<td>2900000</td>
</tr>
<tr>
<td></td>
<td>Computers</td>
<td></td>
<td>1</td>
<td>1</td>
<td>61500</td>
</tr>
<tr>
<td>Feb 2009</td>
<td>Retail and Consumer Products</td>
<td></td>
<td>1</td>
<td>3</td>
<td>2510000</td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td></td>
<td>1</td>
<td>2</td>
<td>23000</td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td></td>
<td>1</td>
<td>4</td>
<td>26400</td>
</tr>
<tr>
<td></td>
<td>Computers</td>
<td></td>
<td>4</td>
<td>11</td>
<td>2625000</td>
</tr>
<tr>
<td>Mar 2009</td>
<td>Retail and Consumer Products</td>
<td></td>
<td>1</td>
<td>2</td>
<td>470000</td>
</tr>
<tr>
<td></td>
<td>Telecomunications</td>
<td></td>
<td>1</td>
<td>2</td>
<td>142000</td>
</tr>
</tbody>
</table>

You can then type a name for the group, such as Fall 08, and then click OK. A derived element is created for the Fall 08 group, which displays and aggregates the data for October and November 2008.
You can also group attributes elements to create derived elements for the winter, spring and summer months. A report with a derived element for each season group is shown below.

<table>
<thead>
<tr>
<th>Month</th>
<th>Customer Industry Sector</th>
<th>Metrics</th>
<th>Sales Orders</th>
<th>Sales Order Items</th>
<th>Net Sales Order Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 08</td>
<td>Retail and Consumer Products</td>
<td>2</td>
<td>11</td>
<td>12038000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td>2</td>
<td>6</td>
<td>4660000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Telecommunications</td>
<td>1</td>
<td>2</td>
<td>2450000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>e-business</td>
<td>1</td>
<td>2</td>
<td>1800000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td>1</td>
<td>3</td>
<td>48000</td>
<td></td>
</tr>
<tr>
<td>Winter 08/09</td>
<td>Retail and Consumer Products</td>
<td>3</td>
<td>7</td>
<td>5820000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td>2</td>
<td>6</td>
<td>4773000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Telecommunications</td>
<td>4</td>
<td>17</td>
<td>911000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>e-business</td>
<td>2</td>
<td>5</td>
<td>2380000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td>3</td>
<td>9</td>
<td>3142400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Computers</td>
<td>6</td>
<td>16</td>
<td>3261500</td>
<td></td>
</tr>
<tr>
<td>Spring 09</td>
<td>Retail and Consumer Products</td>
<td>4</td>
<td>11</td>
<td>3087000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td>4</td>
<td>10</td>
<td>4308500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Telecommunications</td>
<td>2</td>
<td>5</td>
<td>4942000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td>1</td>
<td>4</td>
<td>650000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Computers</td>
<td>2</td>
<td>11</td>
<td>1509000</td>
<td></td>
</tr>
<tr>
<td>Summer 09</td>
<td>Retail and Consumer Products</td>
<td>5</td>
<td>20</td>
<td>14785000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Banking and Insurance</td>
<td>5</td>
<td>12</td>
<td>2386000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Telecommunications</td>
<td>6</td>
<td>14</td>
<td>2938000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>e-business</td>
<td>1</td>
<td>5</td>
<td>90000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td>3</td>
<td>10</td>
<td>2626400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Computers</td>
<td>2</td>
<td>3</td>
<td>425000</td>
<td></td>
</tr>
</tbody>
</table>

As shown in the report above, data for each customer industry sector is aggregated for the three months of each group. For example, Retail and Consumer Products had one sales order in June 2009, three sales orders in July 2009, and one sales order in August 2009. This data is aggregated to a total of five sales orders for the Summer 09 time period.

**Creating a derived element on reports with a quick group**

The steps below show you how to create a quick group on a report connected to an Intelligent Cube.

**Prerequisites**

- The report on which you create the derived element is connected to an active Intelligent Cube.
You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

### To create a derived element with a quick group

1. In MicroStrategy Desktop or Web, log in to a project that contains Intelligent Cube reports.

2. Open the Intelligent Cube reports and run the report. Then view the report in either Grid View or Grid and Graph View.

3. In the grid display of the report, hold down the CTRL key and select multiple attribute elements within the same attribute.

   Do not select derived elements for the attribute, as you cannot create quick groups based on derived elements. To group derived elements, you must use the Derived Elements Editor.

4. Right-click your selection and select Group. The Create Group dialog box opens.

5. Type a name for the derived element, and click OK.

   The group is created as a derived element and displayed on the report. You can modify the derived element using the Derived Elements Editor (see Using the Derived Elements Editor, page 97).

### Creating a derived element on Grid/Graphs with a quick group

You can create quick groups on Grid/Graphs in documents in Interactive or Editable Mode in MicroStrategy Web. For steps to create quick groups on Grid/Graphs in documents in MicroStrategy Web, see the MicroStrategy Web Help.

### Creating quick calculations

Using right-click options, you can group attribute elements and derived elements into a single derived element using the following quick calculations.
Quick calculations enable you to view various types of data and analysis on the same report.

Note the following:

- If you select exactly two attribute elements, you can choose from all of the calculations listed below.

- If you select more than two attribute elements, Subtract and Divide are not available calculations as they can only accept two operands. If you want to create a subtraction or division including more than two attribute elements, you must use the Derived Elements Editor.

- The Greatest and Least calculations are best used when your report has only one metric. These calculations operate on each metric individually, so if your report has more than one metric, the values for the Greatest derived element will not necessarily correspond to the same attribute element.

- **Sum**: Performs an addition of two or more attribute elements, derived elements, or a combination of both.

- **Subtract**: Performs a subtraction of two attribute elements, derived elements, or a combination of both. The element that is at the higher position on the report is the first operand of the subtraction, and the element at the lower position is the second operand of the subtraction.

- **Average**: Performs an average of two or more attribute elements, derived elements, or a combination of both.

- **Divide**: Performs a division of two attribute elements, derived elements, or a combination of both. The element that is at the higher position is the first operand of the division, and the element at the lower position is the second operand of the division.

- **Greatest**: Calculates and displays the greatest value of each metric for two or more attribute elements, derived elements, or a combination of both.

The two tables below show an example of a creating a greatest calculation on the books and electronics categories.

<table>
<thead>
<tr>
<th>Category</th>
<th>Revenue</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>$1,000</td>
<td>$500</td>
</tr>
<tr>
<td>Electronics</td>
<td>$2,000</td>
<td>$300</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
• **Least**: Calculates and displays the least value of each metric for two or more attribute elements, derived elements, or a combination of both.

Using the same example used to illustrate the Greatest calculation listed above, the table below shows an example of creating a least calculation on the books and electronics categories.

<table>
<thead>
<tr>
<th>Category</th>
<th>Revenue</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greatest of Books and Electronics</td>
<td>$2,000</td>
<td>$500</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

A quick calculation is created as a Calculation derived element. For information on Calculation derived elements, see *Calculation derived element, page 77.*
To illustrate quick calculations, consider the following example. You have a report that displays the unit price, cost, and profit for all items sold in the Action movies subcategory, as shown below.

<table>
<thead>
<tr>
<th>Subcategory</th>
<th>Item</th>
<th>Unit Cost</th>
<th>Unit Price</th>
<th>Unit Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Small Soldiers</td>
<td>$18.05</td>
<td>$20.00</td>
<td>$1.95</td>
</tr>
<tr>
<td></td>
<td>The Mask of Zorro</td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>Vanishing Point</td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>Godzilla</td>
<td>$9.63</td>
<td>$11.00</td>
<td>$1.37</td>
</tr>
<tr>
<td></td>
<td>Apollo 13</td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>Lethal Weapon 4</td>
<td>$14.44</td>
<td>$17.00</td>
<td>$2.56</td>
</tr>
<tr>
<td></td>
<td>Le Mans</td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>The Scarlet Pimpernel</td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>The African Queen</td>
<td>$7.22</td>
<td>$8.00</td>
<td>$0.78</td>
</tr>
<tr>
<td></td>
<td>Armageddon</td>
<td>$14.44</td>
<td>$16.00</td>
<td>$1.56</td>
</tr>
<tr>
<td></td>
<td>The Avengers</td>
<td>$22.87</td>
<td>$26.00</td>
<td>$3.13</td>
</tr>
<tr>
<td></td>
<td>Blade</td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>Everest</td>
<td>$13.24</td>
<td>$14.00</td>
<td>$0.76</td>
</tr>
<tr>
<td></td>
<td>Apocalypse Now</td>
<td>$16.85</td>
<td>$19.00</td>
<td>$2.15</td>
</tr>
<tr>
<td></td>
<td>Manhunter</td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
</tbody>
</table>

You decide to do further analysis based on the unit cost of the various items listed. To provide this analysis you begin to put the items into various groups that perform an average of their data. You create the first group by selecting Vanishing Point, Godzilla, Apollo 13, Le Mans, The African Queen, and
Manhunter, then right-clicking the selection, pointing to Create Calculation, and selecting Average. This process is shown below.

You then type a name for the group, such as Average Unit Costs $7.00-$9.99, and then click OK. A derived element is created for the Average Unit Costs $7.00-$9.99 group, which displays an average of the data for Vanishing Point, Godzilla, Apollo 13, Le Mans, The African Queen, and Manhunter.

When you create the quick calculation, the resulting derived element appears at the top of the report, and its attribute elements remain below it. To hide these elements, use the Derived Elements Editor. To show or hide attribute elements using the Derived Elements Editor, see Displaying derived elements and their attribute elements simultaneously, page 127.
You can then group the rest of the items into two other groups, one for Average Unit Costs $10.00-$13.99 and one for Unit Costs $14.00+. The resulting report is shown below.

<table>
<thead>
<tr>
<th>Subcategory</th>
<th>Item</th>
<th>Metrics</th>
<th>Unit Cost</th>
<th>Unit Price</th>
<th>Unit Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Average Unit Costs $7.00—$9.99</td>
<td></td>
<td>$8.42</td>
<td>$9.17</td>
<td>$0.74</td>
</tr>
<tr>
<td></td>
<td>Average Unit Costs $10.00—$13.99</td>
<td></td>
<td>$12.34</td>
<td>$13.25</td>
<td>$0.91</td>
</tr>
<tr>
<td></td>
<td>Average Unit Costs $14.00+</td>
<td></td>
<td>$17.33</td>
<td>$19.60</td>
<td>$2.27</td>
</tr>
<tr>
<td></td>
<td>Small Soldiers</td>
<td></td>
<td>$10.05</td>
<td>$20.00</td>
<td>$1.95</td>
</tr>
<tr>
<td></td>
<td>The Mask of Zorro</td>
<td></td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>Vanishing Point</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>Godzilla</td>
<td></td>
<td>$9.63</td>
<td>$11.00</td>
<td>$1.37</td>
</tr>
<tr>
<td></td>
<td>Apollo 13</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>Lethal Weapon 4</td>
<td></td>
<td>$14.44</td>
<td>$17.00</td>
<td>$2.56</td>
</tr>
<tr>
<td></td>
<td>Le Mans</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
<tr>
<td></td>
<td>The Scarlet Pimpernel</td>
<td></td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>The African Queen</td>
<td></td>
<td>$7.22</td>
<td>$8.00</td>
<td>$0.78</td>
</tr>
<tr>
<td></td>
<td>Armageddon</td>
<td></td>
<td>$14.44</td>
<td>$16.00</td>
<td>$1.56</td>
</tr>
<tr>
<td></td>
<td>The Avengers</td>
<td></td>
<td>$22.97</td>
<td>$26.00</td>
<td>$3.13</td>
</tr>
<tr>
<td></td>
<td>Blade</td>
<td></td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
</tr>
<tr>
<td></td>
<td>Everest</td>
<td></td>
<td>$13.24</td>
<td>$14.00</td>
<td>$0.76</td>
</tr>
<tr>
<td></td>
<td>Apocalypse Now</td>
<td></td>
<td>$16.95</td>
<td>$19.00</td>
<td>$2.15</td>
</tr>
<tr>
<td></td>
<td>Manhunter</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
</tr>
</tbody>
</table>

You can now view your data at a new summarized level. The report shows a relatively sizable average unit profit for items with unit costs greater than $14.00. You can continue your analysis to see whether the higher prices are affecting the average number of items sold, as shown below.

<table>
<thead>
<tr>
<th>Subcategory</th>
<th>Item</th>
<th>Metrics</th>
<th>Unit Cost</th>
<th>Unit Price</th>
<th>Unit Profit</th>
<th>Units Sold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Average Unit Costs $7.00—$9.99</td>
<td></td>
<td>$8.42</td>
<td>$9.17</td>
<td>$0.74</td>
<td>3,671</td>
</tr>
<tr>
<td></td>
<td>Average Unit Costs $10.00—$13.99</td>
<td></td>
<td>$12.34</td>
<td>$13.25</td>
<td>$0.91</td>
<td>3,773</td>
</tr>
<tr>
<td></td>
<td>Average Unit Costs $14.00+</td>
<td></td>
<td>$17.33</td>
<td>$19.60</td>
<td>$2.27</td>
<td>2,438</td>
</tr>
<tr>
<td></td>
<td>Small Soldiers</td>
<td></td>
<td>$18.05</td>
<td>$20.00</td>
<td>$1.95</td>
<td>2,278</td>
</tr>
<tr>
<td></td>
<td>The Mask of Zorro</td>
<td></td>
<td>$12.03</td>
<td>$13.00</td>
<td>$0.97</td>
<td>3,777</td>
</tr>
<tr>
<td></td>
<td>Vanishing Point</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
<td>3,778</td>
</tr>
<tr>
<td></td>
<td>Godzilla</td>
<td></td>
<td>$9.63</td>
<td>$11.00</td>
<td>$1.37</td>
<td>3,007</td>
</tr>
<tr>
<td></td>
<td>Apollo 13</td>
<td></td>
<td>$8.42</td>
<td>$9.00</td>
<td>$0.58</td>
<td>3,815</td>
</tr>
<tr>
<td></td>
<td>Lethal Weapon 4</td>
<td></td>
<td>$14.44</td>
<td>$17.00</td>
<td>$2.56</td>
<td>2,275</td>
</tr>
</tbody>
</table>
As shown in the report above, the higher prices have no negative effect on the number of items sold for the items you expect a higher profit margin on. This type of analysis can lead you to update your pricing guidelines to maximize profits for items of varying values.

**Creating a derived element on a report with a quick calculation**

The steps below describe how to create a quick calculation on a report connected to an Intelligent Cube.

**Prerequisites**

- The report on which you create the derived element is connected to an active Intelligent Cube.
- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

**To create a derived element with a quick calculation**

1. In MicroStrategy Desktop or Web, log in to a project that contains reports connected to active Intelligent Cubes.
2. Open the view report and run the report. Then view the report in either Grid View or Grid and Graph View.
3. In the grid display of the report, press the CTRL key and select multiple attribute elements within the same attribute.
4. Right-click your selection, point to Create Calculation, and then select one of the following calculations. Each is described in Creating quick calculations, page 86.

   - Note the following:
     - If you selected exactly two attribute elements, you can choose from all of the calculations listed below.
     - If you select more than two attribute elements, Subtract and Divide are not available calculations as they can only accept two operands.
     - If you want to create a subtraction or division including more than two attribute elements, you must use the Derived Elements Editor.
- Sum
- Subtract
- Average
- Greatest
- Least
- Divide

The Defining Group dialog box opens.

5 Type a name for the derived element and click OK.

The calculation is created as a derived element and displayed on the report. You can modify the derived element using the Derived Elements Editor (see Using the Derived Elements Editor, page 97).

Creating a derived element on Grid/Graphs with a quick calculation

You can create quick calculations on Grid/Graphs in documents in Interactive or Editable Mode in MicroStrategy Web. For steps to create quick calculations on Grid/Graphs in documents in MicroStrategy Web, see the MicroStrategy Web Help.

Creating quick sorts

Using right-click options, you can group attribute elements into a single derived element that sorts the attribute elements in any order you choose. This enables you to do more advanced attribute element sorting than simple ascending or descending sorts.
For example, you have a report that lists the vendors with the ten largest open accounts payable amounts, as shown below.

The example report below was created in the Financial Reporting Analysis Module.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Metrics</th>
<th>Open Payable Amount</th>
<th>Gross Payable Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best Insurance</td>
<td></td>
<td>$231,162</td>
<td>$4,827,296</td>
</tr>
<tr>
<td>J.D. Soft and Company</td>
<td></td>
<td>$182,003</td>
<td>$845,007</td>
</tr>
<tr>
<td>Grocers Market</td>
<td></td>
<td>$166,114</td>
<td>$1,592,643</td>
</tr>
<tr>
<td>Care Pharmacy</td>
<td></td>
<td>$147,501</td>
<td>$2,114,454</td>
</tr>
<tr>
<td>Highcard Corporation</td>
<td></td>
<td>$139,892</td>
<td>$1,855,994</td>
</tr>
<tr>
<td>Jupiter Networks</td>
<td></td>
<td>$96,014</td>
<td>$3,997,077</td>
</tr>
<tr>
<td>Fiber Optics Networks</td>
<td></td>
<td>$91,865</td>
<td>$2,415,128</td>
</tr>
<tr>
<td>MEICO Insurance</td>
<td></td>
<td>$84,886</td>
<td>$353,385</td>
</tr>
<tr>
<td>Digivision Systems Corp.</td>
<td></td>
<td>$84,323</td>
<td>$4,105,835</td>
</tr>
<tr>
<td>Money Market Credit Corp.</td>
<td></td>
<td>$79,177</td>
<td>$207,078</td>
</tr>
</tbody>
</table>

After reviewing the list of vendors with the largest open payable amounts from highest to lowest, you can sort the vendors into any order that meets your requirements. You can achieve this by right-clicking the Vendor attribute, pointing to Sort, and then selecting List (custom). The Derived Elements Editor opens with all of the attribute elements listed as selected objects.

A quick sort is created as a Group derived element. For information on Group derived elements, see Group derived element, page 72.

You can move the elements up and down in the Selected objects list to re-order them on the report. When you are ready, click OK to accept the
changes and return to the report. Notice the new order of Vendor attribute
elements in the report shown below.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Metrics</th>
<th>Open Payable Amount</th>
<th>Gross Payable Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fiber Optics Networks</td>
<td>$91,865</td>
<td>$2,415,128</td>
<td></td>
</tr>
<tr>
<td>Jupiter Networks</td>
<td>$96,014</td>
<td>$3,997,077</td>
<td></td>
</tr>
<tr>
<td>Best Insurance</td>
<td>$231,162</td>
<td>$4,827,296</td>
<td></td>
</tr>
<tr>
<td>MEICO Insurance</td>
<td>$84,886</td>
<td>$358,385</td>
<td></td>
</tr>
<tr>
<td>Care Pharmacy</td>
<td>$147,501</td>
<td>$2,114,454</td>
<td></td>
</tr>
<tr>
<td>Grocers Market</td>
<td>$166,114</td>
<td>$1,592,643</td>
<td></td>
</tr>
<tr>
<td>J.D. Soft and Company</td>
<td>$182,003</td>
<td>$845,007</td>
<td></td>
</tr>
<tr>
<td>Highcard Corporation</td>
<td>$139,892</td>
<td>$1,855,994</td>
<td></td>
</tr>
<tr>
<td>Digivision Systems Corp.</td>
<td>$84,323</td>
<td>$4,105,835</td>
<td></td>
</tr>
<tr>
<td>Money Market Credit Corp.</td>
<td>$75,177</td>
<td>$207,078</td>
<td></td>
</tr>
</tbody>
</table>

Creating a derived element on a report with a quick sort

The steps below show you how to create a quick sort on a report connected to
an Intelligent Cube.

Prerequisites

- The report on which you create the derived element is connected to an
  active Intelligent Cube.
- The attribute you select to create a quick sort for cannot have any existing
  derived elements defined on it for the report.
- You need the Define Derived Elements (Desktop) and/or the Web Define
  Derived Elements (Web) privileges. These privileges are part of OLAP
  Services.

To create a derived element with a quick sort

1 In MicroStrategy Desktop or Web, log in to a project that contains reports
   connected to active Intelligent Cubes.

2 Open the view report and run the report. Then view the report in either
   Grid View or Grid and Graph View.
3 Right-click an attribute in the grid display, point to Sort, and then select Custom. The Derived Elements Editor opens with all of the attribute elements listed as selected objects.

If Custom is not an option in the Sort options, this means that the attribute has a derived element defined for it on the report. You must remove any derived elements defined for the attribute before you can create a quick sort to re-order the attribute elements.

4 In the Selected objects list, move the attribute elements to the order you want them displayed on the report.

5 Once the attribute elements are ordered appropriately, click OK. The Derived Elements Editor closes and you are returned to the report.

The report displays the attribute elements in the new order. You can modify the derived element using the Derived Elements Editor (see Using the Derived Elements Editor, page 97).

Creating a derived element on Grid/Graphs with a quick sort

The steps below show you how to create a quick sort on a Grid/Graph included in a Report Services document from MicroStrategy Desktop.

You can create quick sorts on Grid/Graphs in documents in Interactive or Editable Mode in MicroStrategy Web. For steps to create quick calculations on Grid/Graphs in documents in MicroStrategy Web, see the MicroStrategy online help.

Prerequisites

• The Report Services document with a Grid/Graph included in one of the document sections, on which you create the derived element.

• The attribute you select to create a quick sort for cannot have any existing derived elements defined on it for the report.

• You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

To create a derived element with a quick sort

1 In MicroStrategy Desktop, log in to a project that contains Grid/Graphs in Report Services documents.
2. Open the document in Design View and expand the document section that contains the Grid/Graph.

3. Right-click the Grid/Graph and select **Edit Grid**.

4. Right-click an attribute, point to **Sort**, and then select **List (custom)**. The Derived Elements Editor opens with all of the attribute elements listed as selected objects.

   - **If List (custom) is not an option in the Sort options, this means that the attribute has a derived element defined for it on the source report.** You must remove any derived elements defined for the attribute before you can create a quick sort to re-order the attribute elements.

5. In the **Selected objects** list, re-order the attribute elements as you want to view them on the report.

6. Once the attribute elements are ordered appropriately, click **OK**. The Derived Elements Editor closes and you are returned to the report.

   The document displays the attribute elements in the new order. You can modify the derived element using the Derived Elements Editor (see *Using the Derived Elements Editor, page 97*).

---

**Using the Derived Elements Editor**

While the quick groups features described in *Quickly creating groups, calculations, and sorts, page 82* enable you to quickly create derived elements using right-click menu options while reviewing report results, the Derived Elements Editor provides the full set of derived elements.
functionality when creating derived elements. The Derived Elements Editor is shown in the image below.

For example, if you use right-click menu options create a quick calculation on a report connected to an Intelligent Cube, the expression can only include one type of function or operand such as +, /, and Average. However, in the Derived Elements Editor, you can create expressions with a valid combination of different functions. You can create a derived element from the Derived Elements Editor with an expression of the following form:

\[
\frac{(\text{AttributeElement1} + \text{AttributeElement2})}{\text{Sum(AllAttributeElements)}},
\]

You can modify derived elements in the following ways using the Derived Elements Editor:

- *Applying derived element values to subtotals, page 122*
- *Displaying derived elements or their attribute elements, page 125*
- *Displaying derived elements and their attribute elements simultaneously, page 127*
- *Formatting derived elements, page 130*
• Creating and using stand-alone derived elements, page 117

• Creating Filter derived elements with Not in List and Where filter qualifications (see Filtering attribute elements to create a derived element, page 107)

• Creating advanced Calculation derived elements (see Using calculations to create derived elements, page 111)

• Deleting derived elements
Sample report

The report shown below is used in the procedures and examples that follow for creating derived elements with the Derived Elements Editor.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$16,347</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$46,461</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>Books</td>
<td>$455,964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,500,514</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,367</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$25,895</td>
<td></td>
</tr>
<tr>
<td>Northwest</td>
<td>Books</td>
<td>$19,734</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,604,110</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$5,639</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$5,508</td>
<td></td>
</tr>
<tr>
<td>South</td>
<td>Books</td>
<td>$216,663</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$739,064</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$12,905</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,882</td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td>Books</td>
<td>$25,609</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$419,847</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$37,704</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$6,965</td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td>Books</td>
<td>$41,576</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$686,421</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,294</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

Accessing the Derived Elements Editor

Using the Derived Elements Editor, you can create derived elements with groups, filters, and calculations. These derived elements can be created from
the Derived Elements Editor in MicroStrategy Desktop and Web in the reporting objects listed below:

- A report which is connected to an active Intelligent Cube. The table below lists the views you can access the Derived Elements Editor from, in MicroStrategy Desktop and Web. To access the Derived Elements Editor from reports, see *To access the Derived Elements Editor in reports connected to an active Intelligent Cube, page 102.*

- A Grid/Graph in a Report Services document. The table below lists the modes you can access the Derived Elements Editor from, in MicroStrategy Desktop and Web. To access the Derived Elements Editor from Grid/Graphs, see *To access the Derived Elements Editor in Grid/Graphs, page 102.*

<table>
<thead>
<tr>
<th>Report Object Type</th>
<th>View/Mode</th>
<th>MicroStrategy Desktop/Web</th>
<th>Can Access Derived Element Editor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report connected to an active Intelligent Cube</td>
<td>Grid View</td>
<td>Desktop and Web</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Grid and Graph View</td>
<td>Desktop and Web</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Design View</td>
<td>Desktop</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Design View</td>
<td>Web</td>
<td>No</td>
</tr>
<tr>
<td>Grid/Graph in a Report Services document</td>
<td>Design View</td>
<td>Desktop</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Design Mode</td>
<td>Web</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Interactive Mode</td>
<td>Web</td>
<td>Yes</td>
</tr>
<tr>
<td>(Grid/Graph must be displayed as a Grid or a Grid and Graph)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Editable Mode</td>
<td>Web</td>
<td>Yes</td>
</tr>
<tr>
<td>(Grid/Graph must be displayed as a Grid or a Grid and Graph)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You can also create stand-alone derived elements by accessing the Derived Elements Editor from outside reports or Grid/Graphs. Stand-alone derived elements can be used by multiple reports and Grid/Graphs. For information on using a derived element in multiple reports and accessing the Derived Elements Editor to create stand-alone derived elements, see *Creating and using stand-alone derived elements, page 117.*
To access the Derived Elements Editor in reports connected to an active Intelligent Cube

1. In MicroStrategy Desktop or Web, log in to a project that contains reports connected to active Intelligent Cubes.

2. Open a report and run it. View it in either Grid View, Grid and Graph View, or Design View.

3. In the grid display of the report, right-click the attribute you want to create or modify derived elements for, and click Derived Elements.

The Derived Elements Editor opens.

To access the Derived Elements Editor in Grid/Graphs

1. In MicroStrategy Desktop or Web, log in to a project that contains Grid/Graphs in Report Services documents.

2. Open the document. Accessing the Derived Elements Editor depends on what mode you are viewing the document in. Follow the steps below depending on which view or mode you are using:

   • Design View in MicroStrategy Desktop only:
     a. Expand the document section that contains the Grid/Graph.
     b. Right-click the Grid/Graph and select Edit Grid.
     c. Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

   • Interactive Mode or Editable Mode in MicroStrategy Web only:
     a. In the grid display of the Grid/Graph, right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

Grouping attribute elements to create a derived element

When viewing a report you can combine attribute elements into a single group of aggregated data. This combined set of attribute elements is called a
Group derived element. This derived element enables you to view and analyze the attribute elements’ data as a single, distinct group.

For example, in a report with Region and Category attributes and a Profit metric (see the image in Sample report, page 100), you can combine the regions on the report into various geographical groups. The final report you create with attribute elements grouped based on geography is shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td>$523,034</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$5,853,393</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$79,423</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$45,597</td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td>$61,310</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,290,530</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$66,426</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$15,803</td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td>$272,158</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,674,186</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$29,251</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$90,238</td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
</tr>
</tbody>
</table>

The steps below show you how to create a Group derived element on a report, as well as specific instructions to create the sample report shown above.

**Prerequisites**

- The report on which you create the derived element is connected to an active Intelligent Cube, or the Report Services document with a Grid/Graph included in one of the document sections, on which you create the derived element.

- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.
To create a Group derived element

While this procedure creates only Group derived elements, you can create any combination of Group, Filter, and Calculation derived elements on a report or Grid/Graph.

1 Log in to a project in MicroStrategy Desktop. For steps to use the Derived Elements Editor to create derived elements in MicroStrategy Web, see the MicroStrategy Web Help.

To access the Derived Elements Editor in reports and Grid/Graphs

2 Open the report or Grid/Graph (the example scenario uses a report connected to an Intelligent Cube), as follows:

- To access the Derived Elements Editor for a report connected to an Intelligent Cube:
  a Execute the report and run it in Grid View or Grid and Graph View.
  b Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

    For the example scenario, right-click the Region attribute, and click Derived Elements.

- To access the Derived Elements Editor for a Grid/Graph in a Report Services document:
  a Expand the document section that contains the Grid/Graph.
  b Double-click the Grid/Graph to edit it.
  c Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

    For the example scenario, right-click the Region attribute, and click Derived Elements.

To create a Group derived element

3 To create a new Group derived element, from the New drop-down list select Group.

Two new derived elements are created, a blank Group derived element and an All Other derived element. The All Other derived element is a collection of all attribute elements that are not included in any of the
other derived elements for the attribute. For further explanation of the All Other derived element, see *All Other derived element, page 79.*

4 Select the new Group derived element. This displays the available attribute elements in the Definition tab.

For the example scenario, the Region attribute elements are displayed.

5 From the left pane, select attribute elements to include in the derived element, and then click the right arrow (>) to add your selections to the Selected objects pane.

For the example scenario, select the Mid-Atlantic, Northeast, and Southeast attribute elements.

6 To rename the Group derived element, from the Change Group drop-down list, select Rename Group. Type a name for the derived element.

For the example scenario, rename the group as East Coast.

**To make derived element modifications**

7 From the Property tab, you can make various modifications to the new derived element such as:

- *Displaying derived elements or their attribute elements, page 125*
- *Applying derived element values to subtotals, page 122*
- *Displaying derived elements and their attribute elements simultaneously, page 127*

8 From the Change Group drop-down list, you can format derived element headers and values. For information on these formatting techniques, see *Formatting derived elements, page 130.*

9 You can change the order in which the derived elements are displayed on the report using the up (▲) and down (▼) arrows.

**To complete derived element creation**

10 You can continue to create more derived elements, or you can click OK to close the Derived Elements Editor and return to the report. The steps below continue the example scenario.

11 From the New drop-down list, select Group. A blank group is created.
12 Select the new derived element. This displays the available attribute elements in the Definition tab.

13 From the left pane, select the Northeast and Southeast attribute elements, and then click the right arrow (>) to add your selections to the Selected objects pane.

14 From the Change Element drop-down list, select Rename Group. Type West Coast to rename the Group derived element.

15 From the New drop-down list, select Group. A blank group is created.

16 Select the new derived element. This displays the available attribute elements in the Definition tab.

17 From the left pane, select the Central and South attribute elements, and then click the right arrow (>) to add your selections to the Selected objects pane.

18 From the Change Group drop-down list, select Rename Group. Type Central and South to rename the derived element group.

19 You can save your derived element for the report or Grid/Graph, or save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs:

- To save the derived element for the report or Grid/Graph, click OK. The Derived Elements Editor closes and you are returned to the report or document.

- To save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs, click Save Groups. Choose a location to save the derived element to, type a name, and click Save. Click OK. The Derived Elements Editor closes and you are returned to the report or document.

Stand-alone derived elements can only be modified by editing the stand-alone object; you cannot modify them from within reports or Grid/Graphs. For information stand-alone derived elements, see Creating and using stand-alone derived elements, page 117.

If you used the steps above to create the sample report, the report is displayed with the regions grouped into East Coast, West Coast, Central and South, and Web.
Filtering attribute elements to create a derived element

While viewing a report, you can combine attribute elements into a single group of aggregated data using filter qualifications. This group of attribute elements is called a Filter derived element. This derived element enables you to view and analyze the attribute elements’ data as a single distinct group by utilizing various filter qualifications.

For example, in a report with Region and Category attributes and a Profit metric (see the image in Sample report, page 100), you can filter the regions on the report into various geographical groups based on the region names. The final report you create with Filter derived elements is shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Southern Regions</td>
<td>Books</td>
<td>$283,848</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,845,331</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$111,396</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$30,141</td>
<td></td>
</tr>
<tr>
<td>Northern Regions</td>
<td>Books</td>
<td>$475,697</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,104,624</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$33,006</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$31,404</td>
<td></td>
</tr>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$15,347</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$73,357</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$45,461</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

The steps below show you how to create a Filter derived element on a report, as well as specific instructions to create the report shown above.

Prerequisites

- The report on which you create the derived element is connected to an active Intelligent Cube, or the Report Services document with a Grid/Graph included in one of the document sections, on which you create the derived element.
• You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

To create a Filter derived element

While this procedure creates only Filter derived elements, you can create any combination of Group, Filter, and Calculation derived elements on a report or Grid/Graph.

1 Log in to a project in MicroStrategy Desktop. For steps to use the Derived Elements Editor to create derived elements in MicroStrategy Web, see the MicroStrategy Web Help.

To access the Derived Elements Editor in reports and Grid/Graphs

2 Open the report or Grid/Graph (the example scenario uses a report connected to an Intelligent Cube), as follows:

• To access the Derived Elements Editor for a report connected to an Intelligent Cube:
  a Execute the report. Then view it in either Grid View or Grid and Graph View.
  b Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

For the example scenario, right-click the Region attribute, and click Derived Elements.

• To access the Derived Elements Editor for a Grid/Graph in a Report Services document:
  a Expand the document section that contains the Grid/Graph.
  b Double-click the Grid/Graph to edit it.
  c Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.

For the example scenario, right-click the Region attribute, and click Derived Elements.
To create a Filter derived element

3 To create a new Filter derived element, from the New drop-down list, select Filter.

Two new derived elements are created, a blank Filter derived element and an All Other derived element. The All Other derived element is a collection of all attribute elements that are not included in any of the other derived elements for the attribute. For further explanation of the All Other group, see All Other derived element, page 79.

4 Select the new Filter derived element.

5 From the Definition tab, click Click here to start a new qualification.

6 Click Field, and then select an attribute.

   For the example scenario, select Region.

7 Click Operator, and then select one of the following operators to create a filter qualification (for the example scenario, select Where):

   • In list: Returns attribute data for the list of attribute elements you select. Click Value, and then select the attribute elements to return data for.

   • Not in List: Returns attribute data for the list of attribute elements that are not in the list of attribute elements you select. Click Value, and then select the attribute elements to exclude data for.

   • Where: Returns attribute data based on a filter qualification of an attribute form. Proceed to the next step to select an attribute form and complete the filter qualification.

8 For filter qualifications that use the operator Where, new Field, Operator, and Value fields appear. Follow the steps below to complete the filter qualification:

   a Click Field, and then select an attribute form. For the example scenario, select DESC.

   b Click Operator, and then select the operator for the filter qualification on the attribute form. For the example scenario, select Begins with.

   c Click Value, and use one of the options to enter in the required value. For the example scenario, select Type a value, and then type South.
9 To rename the derived element group, from the **Change Group** drop-down list, and select **Rename Group**. Type a name for the Filter derived element.

For the example scenario, rename the derived element as **Southern Regions**.

**To make derived element modifications**

10 From the **Property** tab, you can make various modifications to the new derived element such as:

- *Displaying derived elements or their attribute elements, page 125*
- *Applying derived element values to subtotals, page 122*
- *Displaying derived elements and their attribute elements simultaneously, page 127*

11 From the **Change Group** drop-down list, you can format derived element headers and values. For information on these formatting techniques, see *Formatting derived elements, page 130*.

12 You can change the order in which the derived elements are displayed on the report using the up (↑) and down (↓) arrows.

**To complete derived element creation**

13 You can continue to create more derived elements, or you can click **OK** to close the Derived Elements Editor and return to the report. The steps below continue the example scenario.

14 From the **New** drop-down list, select **Filter**. A blank Filter derived element is created.

15 Select the new derived element.

16 From the **Definition** tab, click **Click here to start a new qualification**.

17 Click **Field**, and then select **Region**.

18 Click **Operator**, and then select **Where**.

19 Click **Field**, and then select the **DESC** attribute form.

20 Click **Operator**, and then select **Begins with**.
21 Click **Value**, and then select **Type a value**, and then type **North**.

22 To rename the derived element group, from the **Change Group** drop-down list, and select **Rename Group**. Type **Northern Regions** to rename the derived element.

23 You can save your derived element for the report or Grid/Graph, or save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs:

- To save the derived element for the report or Grid/Graph, click **OK**. The Derived Elements Editor closes and you are returned to the report or document.

- To save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs, click **Save Groups**. Choose a location to save the derived element to, type a name, and click **Save**. Click **OK**. The Derived Elements Editor closes and you are returned to the report or document.

Stand-alone derived elements can only be modified by editing the stand-alone object; you cannot modify them from within reports or Grid/Graphs. For information on using derived elements, see *Creating and using stand-alone derived elements, page 117*.

If you used the steps above to create the sample report, the report is displayed with the regions grouped into Southern Regions and Northern Regions, along with the Central, Mid-Atlantic, and Web regions.

**Using calculations to create derived elements**

When viewing a report you can combine attribute elements into a single set of aggregated data using calculations. This combined set of attribute elements is called a Calculation derived element. This derived element enables you to view and analyze the attribute elements’ data as a single, distinct group by utilizing various calculation functions and operators.

For example, in a report with Region and Category attributes and a Profit metric (see the image in *Sample report, page 100*), you can combine the
regions on the report into various groups for profit analysis. The final report you create with Calculation derived elements is shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Profit</td>
<td>Books</td>
<td></td>
<td>$890,696</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$10,287,744</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$161,590</td>
</tr>
<tr>
<td>Average Profit</td>
<td>Books</td>
<td></td>
<td>$111,337</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$1,285,968</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$27,214</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$20,199</td>
</tr>
<tr>
<td>Greatest Regional Profit</td>
<td>Books</td>
<td></td>
<td>$455,964</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$3,933,032</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$60,787</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$79,357</td>
</tr>
<tr>
<td>Greatest Regional Profit % Contribution</td>
<td>Books</td>
<td></td>
<td>51.19%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>38.23%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>27.92%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>48.49%</td>
</tr>
</tbody>
</table>

The steps below show you how to create a Calculation derived element on a report, as well as specific instructions to create the sample report shown above.

**Prerequisites**

- The report on which you create the derived element is connected to an active Intelligent Cube, or the Report Services document with a Grid/Graph included in one of the document sections, on which you create the derived element.

- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

**To create a Calculation derived element**

While this procedure creates only Calculation derived elements, you can create any combination of Group, Filter, and Calculation derived elements on a report or Grid/Graph.

1. Log in to a project in MicroStrategy Desktop. For steps to use the Derived Elements Editor to create derived elements in MicroStrategy Web, see the MicroStrategy Web Help.
To access the Derived Elements Editor in reports and Grid/Graphs

2 Open the report or Grid/Graph (the example scenario uses a report connected to an Intelligent Cube), as follows:

• To access the Derived Elements Editor for a report connected to an Intelligent Cube:
  a Execute the report. The view it in either Grid View or Grid and Graph View.
  b Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.
    For the example scenario, right click the Region attribute, and click Derived Elements.

• To access the Derived Elements Editor for a Grid/Graph in a Report Services document:
  a Expand the document section that contains the Grid/Graph.
  b Double-click the Grid/Graph to edit it.
  c Right-click the attribute to create or modify derived elements for, and click Derived Elements. The Derived Elements Editor opens.
    For the example scenario, right click the Region attribute, and click Derived Elements.

To create a Calculation derived element

3 To create a new Calculation derived element, from the New drop-down list, select Calculation.

Two new derived elements are created, a blank Calculation derived element and an All Other derived element. The All Other derived element is a collection of all attribute elements that are not included in any of the other derived elements for the attribute. For further explanation of the All Other derived element, see All Other derived element, page 79.

4 Select the new Calculation derived element. This displays the available attribute elements in the Definition tab.

For the example scenario, the Region attribute elements are displayed.

5 From the left pane, select attribute elements to include in the Calculation derived element, and add them to the expression area on the right (shown
below). You can also use the toolbar above the expression to include operators and functions, as well as validate or clear the expression.

For the example scenario, select all the attribute elements available and drag the entire selection to the expression area. This adds all the attribute elements to the expression area, and automatically use the addition operator to combine the attribute elements, as shown below.

To rename the derived element, from the Change Group drop-down list, select Rename Group. Type a name for the derived element.

For the example scenario, rename the group as Total Profit.

**To modify derived elements**

7 From the Property tab, you can make various modifications to the new derived element, such as:

- *Displaying derived elements or their attribute elements, page 125*
- *Applying derived element values to subtotals, page 122*
- *Displaying derived elements and their attribute elements simultaneously, page 127*

8 From the Change Group drop-down list, you can format derived element headers and values. For information on these formatting techniques, see *Formatting derived elements, page 130*.

9 You can change the order in which the derived elements are displayed on the report using the up (↑) and down (↓) arrows.
To complete creating a derived element

10 You can continue to create more derived elements, or you can click OK to close the Derived Elements Editor and return to the report. The steps below continue the example scenario.

To create an average calculation of attribute elements

11 From the New drop-down list, select Calculation. A blank Calculation derived element is created.

12 Select the new derived element.

13 On the Definition tab, click f(x). The Insert Function Wizard opens.

14 Select the Average function, and click Next. The Arguments page opens.

15 Click ... (browse) next to Argument 1, and in the Select an Object dialog box, select Central. Then click OK. Repeat this step for each successive argument, selecting the following regions:

• Mid-Atlantic
• Northeast
• Northwest
• South
• Southeast
• Southwest
• Web

16 Click Finish.

17 From the Change Group drop-down list, select Rename Group. Type Average Profit to rename the derived element.

To create a greatest calculation of attribute elements

18 From the New drop-down list, select Calculation. A blank Calculation derived element is created.

19 Select the new derived element.

20 On the Definition tab, click f(x). The Insert Function Wizard opens.
21 Select the **Greatest** function, and click **Next**. The Arguments page opens.

22 Click ... (browse) next to Argument 1, and in the Select an Object dialog box, select **Central**. Then click **OK**. Repeat this step for each successive argument, selecting the following regions:

- Mid-Atlantic
- Northeast
- Northwest
- South
- Southeast
- Southwest
- Web

23 From the **Change Group** drop-down list, select **Rename Group**. Type **Greatest Regional Profit** to rename the derived element.

   **To create a percent contribution from a division of derived elements**

24 From the **New** drop-down list, select **Calculation**. A blank Calculation derived element is created.

25 Select the new derived element.

26 On the **Definition** tab, from the drop-down list select **Groups**.

27 From the left pane, select **Greatest Regional Profit** and drag and drop it into the expression area on the right.

28 Click /.

29 From the left pane, select **Total Profit** and drag and drop it into the expression area on the right. Your final expression should appear as shown below:
30 From the **Change Group** drop-down list, select **Rename Group**. Type **Greatest Regional Profit % Contribution** to rename the derived element.

31 You can save your derived element for the report or Grid/Graph, or save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs:

- To save the derived element for the report or Grid/Graph, click **OK**. The Derived Elements Editor closes and you are returned to the report or document.

- To save the derived element as a stand-alone object that can be used by multiple reports and Grid/Graphs, click **Save Groups**. Choose a location to save the derived element to, type a name, and click **Save**. Click **OK**. The Derived Elements Editor closes and you are returned to the report or document.

Stand-alone derived elements can only be modified by editing the stand-alone object; you cannot modify them from within reports or Grid/Graphs. For information on sharing derived elements, see **Creating and using stand-alone derived elements, page 117**.

If you used the steps above to create the sample report, the report is displayed with the regions grouped into Total Profit, Average Profit, Greatest Regional Profit, and Greatest Regional Profit % Contribution.

**Creating and using stand-alone derived elements**

A stand-alone derived element can be connected to matching attributes in reports connected to Intelligent Cubes and Grid/Graphs. This enables you to use a derived element in multiple reports and Grid/Graphs.

To use a derived element in multiple reports and Grid/Graphs, you must create a stand-alone derived element. There are two methods with to create a stand-alone derived element:

- Create a derived element from within a report or Grid/Graph and then save it as a stand-alone object. You can use the Derived Elements Editor to create and save the derived element as a stand-alone derived element. You can use this method in MicroStrategy Desktop and Web. For information on creating a derived element within a report or Grid/Graph using the Derived Elements Editor, see **Using the Derived Elements Editor, page 97**.
Once you save a derived element as a stand-alone derived element, it can no longer be modified from within the report or Grid/Graph. For information on how to modify a stand-alone derived element, see *Editing stand-alone derived elements, page 121*.

- Create a stand-alone derived element outside of any report or Grid/Graph. You can use this method only in MicroStrategy Desktop. To used this method, see *Creating a stand-alone derived element, page 118*.

Once created, a stand-alone derived element has the following functionality:

- All Group, Calculation, Filter, and All Other derived elements are saved as part of the derived element. You cannot select a subset of the derived elements; you must save and share the entire collection of derived elements.

- A stand-alone derived element can only be connected to the attribute that was used to define the derived element. For information on how to connect derived elements to attributes in other reports and Grid/Graphs, see *Using a derived element in multiple reports or Grid/Graphs, page 119*.

- The stand-alone derived element itself can be modified, but you cannot modify it from within a report or Grid/Graph. Any modifications for the derived element are applied to the derived element in all of the reports and Grid/Graphs that it is used in. For information on how to modify a stand-alone derived element, see *Editing stand-alone derived elements, page 121*.

- The stand-alone derived element can only be deleted if it is not used in any report or Grid/Graph. A list of reports and Grid/Graphs that use the derived element is displayed when you attempt to delete a stand-alone derived element.

### Creating a stand-alone derived element

The steps below show you how to create a stand-alone derived element.

**Prerequisites**

- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.
To create a stand-alone derived element

1. In MicroStrategy Desktop, log in to a project.

2. From the File menu, point to New, and then select Derived Element. The Select Attribute dialog box opens.

3. Browse to and select the attribute you want to base your derived element on, and then click Open. The Derived Element Editor opens.

4. You can create Group, Filter, and Calculation derived elements to define your stand-alone derived element. With the exception any steps to access or close the Derived Elements Editor, you can use the same processes and techniques described in the following sections to create derived elements for a stand-alone derived element:
   - Grouping attribute elements to create a derived element, page 102
   - Filtering attribute elements to create a derived element, page 107
   - Using calculations to create derived elements, page 111

5. Click Save and Close, the Save Derived Element As dialog box opens.

6. Type a name and click Save. The Derived Elements Editor closes and the derived element is saved as a stand-alone object.

You can now use your derived element in reports connected to Intelligent Cubes and Grid/Graphs in Report Services documents, as described in Using a derived element in multiple reports or Grid/Graphs, page 119.

Using a derived element in multiple reports or Grid/Graphs

You can use a stand-alone derived element in multiple reports and Grid/Graphs in documents. To use a stand-alone derived element, you can connect the derived element to an attribute in a report connected to an Intelligent Cube or to an attribute in a Grid/Graph in a Report Services document. However, a derived element can only be connected to the attribute used to define the derived element.

For example, if a derived element is created on the Year attribute, this derived element can only be connected to the Year attribute in reports and Grid/Graphs. You cannot connect this derived element to any other attributes such as Quarter, Month, Category, Customer, and so on.
The steps below show you how to apply a stand-alone derived element to an attribute in a report or Grid/Graph.

**Prerequisites**

- The report on which you connect the derived element to an attribute is connected to an active Intelligent Cube, or the Report Services document with a Grid/Graph included in one of the document sections, on which you connect the derived element to an attribute.

- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

---

**To connect a stand-alone derived element to an attribute in a report or Grid/Graph**

1. Log in to a project in MicroStrategy Desktop or Web.

2. Open the report or Grid/Graph, as follows:

   - To access the Derived Elements Editor for a report connected to an Intelligent Cube:
     a. Execute the report. Then view it in either Grid View or Grid and Graph View.
     b. Right-click the attribute to apply a stand-alone derived element to, and click **Derived Elements**. The Derived Elements Editor opens.

     For the example scenario, right-click the Region attribute, and click **Derived Elements**.

   - To access the Derived Elements Editor for a Grid/Graph in a Report Services document:
     a. Expand the document section that contains the Grid/Graph.
     b. Double-click the Grid/Graph to edit it.
     c. Right-click the attribute to apply a stand-alone derived element to, and click **Derived Elements**. The Derived Elements Editor opens.

     For the example scenario, right-click the Region attribute, and click **Derived Elements**.
To connect a stand-alone derived element to an attribute

If an attribute already has a derived element defined for it in the report or Grid/Graph, connecting a stand-alone derived element overwrites the existing definition.

3 Click Link Groups. The Select Group dialog box opens.

4 Browse to and select the derived element to connect to the attribute, and then click Open.

5 Click OK to save your changes and close the Derived Elements Editor.

Editing stand-alone derived elements

Editing a stand-alone derived element updates the derived element in all reports and Grid/Graphs it is used in. Since these modifications are applied to multiple reporting objects, you can only update the stand-alone derived element itself, you cannot update it from a report or Grid/Graph that it is used in.

The steps below show you how to modify a stand-alone derived element.

Prerequisites

- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

To edit a stand-alone derived element

1 In MicroStrategy Desktop, log in to a project.

2 Browse to and right-click a stand-alone derived element, and then select Edit. The Derived Element Editor opens.

3 You can create and modify Group, Filter, and Calculation derived elements to define your stand-alone derived element. With the exception of any steps to access or close the Derived Elements Editor, you can use the same processes and techniques described in the following sections to create and modify derived elements for a stand-alone derived element:

- Grouping attribute elements to create a derived element, page 102
Defining derived element functionality and formatting

Derived elements provide a wealth of reporting and analysis functionalities that can be defined to achieve your specific requirements. The sections listed below cover various derived elements functionalities and formatting techniques:

- Applying derived element values to subtotals, page 122
- Displaying derived elements or their attribute elements, page 125
- Displaying derived elements and their attribute elements simultaneously, page 127
- Formatting derived elements, page 130

Applying derived element values to subtotals

You can define how each derived element value is applied to all subtotals included on a report or Grid/Graph. The simple report below is used as an example of how you can apply derived element values to subtotals.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td></td>
<td>$890,696</td>
</tr>
<tr>
<td>Electronics</td>
<td></td>
<td>$10,287,744</td>
</tr>
<tr>
<td>Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td>Music</td>
<td></td>
<td>$151,590</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>$11,557,740</strong></td>
</tr>
</tbody>
</table>
You have the following options in the Derived Elements Editor for applying derived element values to subtotals. The examples below are all based on the basic report above.

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Subtotal Behavior</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use this element when calculating subtotals</td>
<td>(default behavior) The derived element values are applied to all subtotals. This means that a derived element’s aggregated data is applied to all subtotals for the report, rather than the values for the individual items that are part of a derived element.</td>
<td>In a report with Category and Profit, you create a derived element defined as Electronics - Books. Applying the derived element value to the subtotal means that subtotals uses the value resulting from the subtraction of the Electronics and Books attribute elements, as shown below.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Category</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Electronics - Books</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total</strong></td>
</tr>
<tr>
<td>Use the individual items that make up this element when calculating subtotals</td>
<td>The values of the separate items that make up the derived element are applied to all subtotals. For Group and Filter derived elements, only attribute elements can be included as items of derived elements. Calculation derived elements can include attribute elements as well as other derived elements.</td>
<td>In a report with Category and Profit, you create a derived element defined as Electronics - Books. Applying the individual item values to the subtotal means that the subtotal uses the individual values of the Electronics and Books attribute elements, as shown below.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Category</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Electronics - Books</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Movies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Music</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total</strong></td>
</tr>
</tbody>
</table>

Notice that the subtotal in this report is the same value as the original report.
The steps below show you how to define subtotal behavior for derived elements.

**Prerequisites**

- A report connected to an active Intelligent Cube, a Grid/Graph in a Report Services document, or a stand-alone derived element.
- This procedure also assumes you have created derived elements for the report, Grid/Graph, or stand-alone derived element.
- You have the Derived Elements Editor open. For information on accessing the Derived Elements Editor, see *Accessing the Derived Elements Editor, page 100*.
- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.

**To define subtotal behavior for derived elements**

1. In the Derived Elements Editor, select a derived element.
2. Select the **Property** tab.
3. From the **Subtotal behavior** drop-down list, select one of the following options:
   - **Use this element when calculating subtotals**

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Subtotal Behavior</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do not use this element when calculating subtotals</td>
<td>This option excludes the derived element values and separate item values from all subtotals. You can use this option to avoid double counting when an attribute element is included in more than one derived element.</td>
<td>In a report with Category and Profit, you create a derived element defined as Electronics - Books. The derived element values as well as the individual attribute element values are excluded from the subtotal, as shown below.</td>
</tr>
</tbody>
</table>
• Use the individual items that make up this element when calculating subtotals

• Do not use this element when calculating subtotals

These options are described in *Applying derived element values to subtotals*.

4 Click **OK** to save your changes and close the Derived Elements Editor.

### Displaying derived elements or their attribute elements

Derived elements can be displayed on reports and Grid/Graphs as a single, consolidated element with aggregated data, or as the separate attribute elements that are items of the derived element. The simple report shown below is used to demonstrate how you can apply derived elements to subtotals.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

You have the following options for displaying derived elements:

• Consolidate items into one element (default behavior, except for the All Other derived element): The derived element’s components are displayed as a single entry with the data of all of its items combined as determined by the derived element.

For example, in a report with Category and Profit, you create a Group derived element that combines the Books and Electronics attribute elements. By keeping the default option, the derived element is displayed as a single entry on the report, as shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td>$1,118,440</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

• Keep individual items separate (default behavior for the All Other derived element): The attribute elements included as items of a derived element are displayed individually instead of the derived element.
For example, in a report with Category and Profit, you create a Group derived element that combines the Books and Electronics attribute elements. You choose to keep the individual options separate, which displays the separate attribute elements in the report, as shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td></td>
<td>$11,178,440</td>
</tr>
<tr>
<td>Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td>Music</td>
<td></td>
<td>$161,590</td>
</tr>
</tbody>
</table>

This is the default for an All Other derived element because it displays all attribute elements (that were not included in any other derived elements) as separate attribute elements. This gives the appearance on the report that these attribute elements are not part of any derived element.

This option is not recommended for Calculation derived elements. The purpose of Calculation derived elements is to use a calculation expression to combine values of attribute elements or derived elements (or a combination of both). Using this option with a Calculation derived element means that these values are not combined using the calculation expression. This does not reflect the purpose of Calculation derived elements.

The steps below show you how to define whether derived elements are displayed as one consolidated entry, or all of the attribute elements that are items of the derived element are displayed individually.

**Prerequisites**

- A report connected to an active Intelligent Cube, a Grid/Graph in a Report Services document, or a stand-alone derived element.
- This procedure also assumes you have created derived elements for the report, Grid/Graph, or stand-alone derived element,
- You have the Derived Elements Editor open. For information on accessing the Derived Elements Editor, see *Accessing the Derived Elements Editor, page 100*.
- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.
To display derived elements as consolidated or individual attribute elements

1. In the Derived Elements Editor, select a derived element.

2. Select the Property tab.

3. Select one of the following Element behavior options:
   - Consolidate items into one element
   - Keep individual items separate

   These options are described in Displaying derived elements or their attribute elements.

4. Click OK to save your changes and close the Derived Elements Editor.

Displaying derived elements and their attribute elements simultaneously

You can analyze attribute elements with derived elements, while also including the attribute element on the report. You can accomplish this by selecting which attribute elements are included in the All Other derived element.

By default, the All Other derived element only collects attribute elements that are not included in derived elements, and includes them on the report as individual attribute elements. For an introduction to the All Other derived element, see All Other derived element, page 79.

The simple report below is used as an example of how you can display derived elements and their attribute elements simultaneously.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>
You have the following options in the Derived Elements Editor to display derived elements and their attribute elements simultaneously:

- **Do not include individual items in the All Other element (default behavior):** The attribute elements that are used to define the derived element are excluded from the All Other derived element. This means that the attribute elements are not displayed, only their combined data for the derived element is displayed.

  For example, in a report with Category and Profit, you create a Group derived element that combines the Books and Electronics attribute elements. By selecting to keep the default option, the derived element is displayed on the report but the attribute elements that makes up these derived elements are not, as shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td>$11,178,440</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

- **Include individual items in the All Other element:** The attribute elements that are used to define the derived element are also included in the All Other derived element. This means that the attribute elements are displayed along with the derived elements they are a part of.

  For example, in a report with Category and Profit, you create a Group derived element that combines the Books and Electronics attribute elements. You select to include the Books attribute element and the Electronics attribute element in the All Other derived element. This displays the derived element, as well as the individual Books and Electronics attribute elements, as shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td>$11,178,440</td>
<td></td>
</tr>
<tr>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

For an attribute element to be included in the All Other derived element, all derived elements that the attribute element is included in must be defined to include their attribute elements in the All Other derived element. This can cause unexpected behavior when an attribute element is included in multiple derived elements.
For example, in a report with Category and Profit, you create a Group derived element that combines the Books and Electronics attribute elements. You create a second Group derived element that combines Electronics and Movies. The report is shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td>$11,178,440</td>
<td></td>
</tr>
<tr>
<td>Electronics and Movies</td>
<td>$10,505,454</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

You select to include the attribute elements of the Books and Electronics derived element in the All Other derived element. Notice that Books is displayed but Electronics is not displayed, as shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books and Electronics</td>
<td>$11,178,440</td>
<td></td>
</tr>
<tr>
<td>Electronics and Movies</td>
<td>$10,505,454</td>
<td></td>
</tr>
<tr>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
</tbody>
</table>

This is because the Electronics attribute element is also a part of the Electronics and Movies derived element, which is defined to exclude its attribute elements in the All Other calculation. To display the Electronics attribute element along with both derived elements, you must define both derived elements to include their attribute elements in the All Other derived element.

The steps below show how to define whether derived elements are displayed with their attribute elements simultaneously.

**Prerequisites**

- A report connected to an active Intelligent Cube, a Grid/Graph in a Report Services document, or a stand-alone derived element.
- This procedure also assumes you have created derived elements for the report, Grid/Graph, or stand-alone derived element.
- You have the Derived Elements Editor open. For information on accessing the Derived Elements Editor, see *Accessing the Derived Elements Editor, page 100*.
- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.
To include or exclude attribute elements in an All Other derived element

1. In the Derived Elements Editor, select a derived element.

2. Select the Property tab.

3. Select one of the following Apply element to all other calculation options:
   - Do not include individual items in the All Other element
   - Include individual items in the All Other element

   These options are described in Displaying derived elements and their attribute elements simultaneously.

4. Click OK to save your changes and close the Derived Elements Editor.

Formatting derived elements

Formatting derived elements lets you highlight important or unique data as well as correctly reflect value and character formats for your derived elements. For example, you may require the values of a derived element to be displayed as a percentage.

Standard formatting of attributes and metrics lets you format all data for an attribute or metric. Formatting an attribute applies formatting to the attribute name and all of its attribute elements. Formatting a metric applies formatting to the metric name and all of its values across all attribute elements.

Formatting derived elements extends these features to allow you to format individual attribute elements as well format metric values associated with individual attribute elements.
For example, the report shown below has derived elements created for the Region attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Profit</td>
<td>Books</td>
<td></td>
<td>$890,696</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$10,287,744</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$161,590</td>
</tr>
<tr>
<td>Average Profit</td>
<td>Books</td>
<td></td>
<td>$111,337</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$1,285,968</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$27,214</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$20,199</td>
</tr>
<tr>
<td>Greatest Regional Profit</td>
<td>Books</td>
<td></td>
<td>$455,964</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$3,933,032</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$60,787</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$73,357</td>
</tr>
<tr>
<td>Greatest Regional Profit % Contribution</td>
<td>Books</td>
<td></td>
<td>51.19%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>38.23%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>27.92%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>46.49%</td>
</tr>
</tbody>
</table>

A procedure to create the report shown above is provided in *Using calculations to create derived elements, page 111.*

Notice that the derived element Greatest Regional Profit % Contribution uses derived element formatting to apply a percentage format to the profit values for the derived element. If you used metric formatting to apply a percentage format to profit values, the percentage format applies to all profit values across all derived elements. But this formats data incorrectly for the derived elements Total Profit, Average Profit, and Greatest Regional Profit.

In addition to applying formatting to metric values to correctly represent data for different derived elements, you can also format the names of each derived element individually. This gives you more flexibility than formatting
attributes, which applies formatting to all the attribute elements. This is shown in the same sample report below with additional formatting on individual derived elements, and formatting on the Category attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Profit</td>
<td>Books</td>
<td></td>
<td>$890,696</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$10,287,744</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$161,590</td>
</tr>
<tr>
<td>Average Profit</td>
<td>Books</td>
<td></td>
<td>$111,337</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$1,285,968</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$27,214</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$20,199</td>
</tr>
<tr>
<td>Greatest Regional Profit</td>
<td>Books</td>
<td></td>
<td>$455,964</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$3,933,032</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$60,787</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$73,357</td>
</tr>
<tr>
<td>Greatest Regional Profit % Contribution</td>
<td>Books</td>
<td></td>
<td>51.19%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>38.23%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>27.92%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>48.49%</td>
</tr>
</tbody>
</table>

Notice that each derived element name has a different format but the attribute elements for the Category attribute all share the same format.

Formatting derived elements is only available from the Derived Elements Editor. The procedure below describes how to format derived elements.

**Prerequisites**

- A report connected to an active Intelligent Cube, a Grid/Graph in a Report Services document, or a stand-alone derived element.
- This procedure also assumes you have created derived elements for the report, Grid/Graph, or stand-alone derived element.
- You have the Derived Elements Editor open. For information on accessing the Derived Elements Editor, see *Accessing the Derived Elements Editor, page 100*.
- You need the Define Derived Elements (Desktop) and/or the Web Define Derived Elements (Web) privileges. These privileges are part of OLAP Services.
To format derived elements

1. In the Derived Elements Editor, select a derived element.

2. From the toolbar, from the Change Group drop-down list, point to Format, and then select from the following options:

   - **Values**: Formatting the values of a derived element applies formatting to all metric values associated with a derived element. You can use this to apply value formats for derived elements that use formulas that return different types of data such as percentages, fractions, or other unique formats. In the report shown below, the derived element values for the various derived elements are highlighted with a solid red box, for the purpose of this example.

     Be aware that formatting a derived element’s values only applies to the metric values associated with the derived element. For example, the first four profit values are associated with the Total Profit derived element.

   - **Headers**: Formatting the headers of a derived element applies formatting to the derived element name. You can use this to highlight or visually distinguish the various derived elements on the report.

     In the report shown below, the derived element headers for the various derived elements are highlighted with a dashed black box, for the purpose of this example. Be aware that formatting a derived
element’s header only applies to the derived element you are formatting, not all derived elements for an attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Profit</td>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$217,710</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$161,590</td>
<td></td>
</tr>
<tr>
<td>Average Profit</td>
<td>Books</td>
<td>$111,337</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,285,968</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,214</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$20,199</td>
<td></td>
</tr>
<tr>
<td>Greatest Regional Profit</td>
<td>Books</td>
<td>$455,964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$79,357</td>
<td></td>
</tr>
<tr>
<td>Greatest Regional Profit % Contribution</td>
<td>Books</td>
<td>51.19%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>36.23%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>27.92%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>48.49%</td>
<td></td>
</tr>
</tbody>
</table>

The Format Cells dialog box opens.

3. Apply number, alignment, font, border, and background formatting for the derived element values or headers. For steps to use the Format Cells dialog box, click the Help on the Format Cells dialog box.

4. When you are finished formatting derived elements headers or values, click OK. You are returned to the Derived Elements Editor.

5. Click OK to save your changes and close the Derived Elements Editor.

**Interaction with other reporting features**

The derived elements feature is just one of the many reporting and analysis features available with MicroStrategy. How you use derived elements in combination with other reporting and analysis features can affect the behavior and results of your reports and documents. For information on how derived elements interact with other MicroStrategy features, see the following sections:

- View filters, page 135
- Page-by, page 136
• **Thresholds, page 138**

• **Drilling, page 139**

• **Subtotals, page 141**

• **Smart metrics, page 142**

**View filters**

View filters restrict the amount of data displayed on the report, providing you with a different view of the data. For information on the full set of view filter functionality, see *Chapter 7, View Filters*.

View filters restrict the same data which is used by derived elements to analyze and display data to meet your reporting needs. Since both of these OLAP Services features use the same data, it is important to understand how the two features can be used together to return desired report results. For information on view filter effects on derived elements and other reporting features, see *View filter effects on reporting features, page 219*.

**Derived metrics**

A derived metric is a metric based on metric data included in the report results returned from the data warehouse. You can use derived metrics to perform column math, that is, calculations on metrics included in the report results, without regenerating or re-executing SQL.

Most derived metrics are represented correctly by derived elements. However, derived metrics that have the same evaluation order requirements as smart metrics can initially be calculated incorrectly when using derived elements.

For example, a derived metric that uses a division such as Profit/Revenue is initially calculated incorrectly when using derived elements. This is because, by default, derived metrics are evaluated before derived elements. You must change the evaluation order to evaluate derived elements before this type of derived metric.

Steps to change the evaluation order of derived elements, derived metrics, and smart metrics are provided in *Smart metrics, page 142*. 
Page-by

To group data into subsets, you can use the *page-by* feature. The subsets you separate your business data into are called pages, and you then page your way through the report, viewing one data subset at a time. Page-by makes viewing a large report easier than scrolling through long lists of data.

Attributes are one of the most common objects included in the page-by area of a report. When an attribute is included in the page-by area, you can select which attribute element to view data for.

For example, the report below on the left includes the attributes Region and Category, along with the Profit metric. The Region attribute is included in the page-by area, and Central is selected by default. The report below on the right displays all the attribute elements available in the page-by area.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>$16,347</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
</tbody>
</table>

If you create derived elements for an attribute included in the page-by area, the derived elements are available from the page-by field to display their associated data.
For example, the report below shows East Coast, West Coast, and Central and South derived elements based on the Region attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td></td>
<td>$523,034</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$5,853,393</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$79,423</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$45,597</td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td></td>
<td>$61,310</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$2,290,530</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$65,426</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$15,803</td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td></td>
<td>$272,158</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$1,674,186</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$29,251</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$90,230</td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td></td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$469,635</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$42,611</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$7,952</td>
</tr>
</tbody>
</table>

If Region is moved to the page-by area, the derived elements are available for selection from the page-by field, along with the Web attribute element, as shown below.

You must move attributes from the page-by area to the grid of the report to create or modify derived elements for the attribute. You can then move the attribute back to the page-by area once all derived element modifications are complete.

Page-by is covered in greater detail in the *MicroStrategy Basic Reporting Guide*. 
Thresholds

Thresholds highlight particular data in a report by displaying special cell formats, symbols, images, or replacement text.

Thresholds are created using conditions on attributes or metrics. Thresholds created using conditions on attributes are not displayed for derived element data. However, the more commonly used thresholds created using conditions on metrics are applied to derived element data.

For example, the report below shows East Coast, West Coast, and Central and South derived elements based on the Region attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td></td>
<td>$528,034</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$5,853,393</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$79,423</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$45,597</td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td></td>
<td>$61,310</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$2,290,530</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$66,426</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$15,803</td>
</tr>
<tr>
<td>Central and South</td>
<td>Electronics</td>
<td></td>
<td>$1,674,186</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$29,251</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$90,238</td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td></td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td>$469,635</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td></td>
<td>$42,611</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td></td>
<td>$7,952</td>
</tr>
</tbody>
</table>

You create two thresholds, the first defined on a metric and the second defined on an attribute, as shown in the following image.
When the report is executed, only the Profit > 400000 threshold is displayed, as shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$79,423</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$46,597</td>
<td></td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td></td>
<td>$61,310</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$66,426</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$16,803</td>
<td></td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td></td>
<td>$272,158</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$29,251</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$90,238</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td></td>
<td>$29,195</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

The Profit > 400000 threshold uses the derived element data to determine when to display the defined formatting.

However, the formatting for the Central and South threshold is not displayed. This is due to the fact that the separate Central and South attribute elements are not on the report, and are instead replaced with a Central and South derived element. While you cannot display this type of threshold formatting for derived elements, you can still apply formatting to the data for each derived element, as described in Formatting derived elements, page 130.

Thresholds are covered in greater detail in the MicroStrategy Basic Reporting Guide.

**Drilling**

Drilling allows you to view displayed report data at levels other than that returned in the original grid or graph report. You can investigate the data in your report quickly and easily with the help of drilling. It allows you to execute another report based on the original report to get more detailed or supplemental information.
When you drill on attribute elements in a report, the resulting report restricts the results to data only for the attribute elements used when drilling. For example, if you drill from the Year 2007 attribute element down to Quarter, the resulting report only includes quarters that are within 2007.

The same standard applies to drilling on derived elements. Drilling on a derived element restricts the resulting report to only data for the attribute elements used to define the derived element.

For example, the report below shows East Coast, West Coast, and Central and South derived elements based on the Region attribute.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td>Books</td>
<td>$528,034</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$5,853,393</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$79,423</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$45,597</td>
<td></td>
</tr>
<tr>
<td>West Coast</td>
<td>Books</td>
<td>$61,310</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,290,530</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$66,426</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$15,803</td>
<td></td>
</tr>
<tr>
<td>Central and South</td>
<td>Books</td>
<td>$272,158</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,674,186</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$29,251</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$90,230</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>
If you drill down from the East Coast derived element to Call Center, the resulting report shown below returns data for Call Centers within the East Coast regions (Northeast, Mid-Atlantic, Southeast).

<table>
<thead>
<tr>
<th>Call Center</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta</td>
<td>Books</td>
<td>$11,623</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$197,187</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$17,766</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$3,167</td>
<td></td>
</tr>
<tr>
<td>Washington, DC</td>
<td>Books</td>
<td>$32,282</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,760,258</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$10,024</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$9,638</td>
<td></td>
</tr>
<tr>
<td>Miami</td>
<td>Books</td>
<td>$13,985</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$222,660</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$19,938</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$3,797</td>
<td></td>
</tr>
<tr>
<td>Boston</td>
<td>Books</td>
<td>$80,969</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$268,450</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$4,549</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$4,356</td>
<td></td>
</tr>
<tr>
<td>New York</td>
<td>Books</td>
<td>$374,994</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,232,064</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$22,818</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$21,539</td>
<td></td>
</tr>
<tr>
<td>Charleston</td>
<td>Books</td>
<td>$14,179</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,172,774</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$4,327</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$4,099</td>
<td></td>
</tr>
</tbody>
</table>

Drilling is covered in greater detail in the MicroStrategy Basic Reporting Guide.

Subtotals

You can define how derived elements are applied to subtotals, which is described in Applying derived element values to subtotals, page 122.
Smart metrics

Smart metrics are applied to compound metrics to ensure that they evaluate correctly for derived elements. Smart metrics calculate subtotals on the individual elements of the compound metric. For example, a smart metric uses the formula \( \text{Sum(Metric1)}/\text{Sum(Metric2)} \) rather than \( \text{Sum(Metric1/Metric2)} \).

Smart metrics can initially be calculated incorrectly when using derived elements because smart metrics are evaluated before derived elements. You must change the evaluation order to evaluate derived elements before the smart metrics.

This same modification to evaluation order is also required for derived metrics that use a division.

For example, the report shown below includes the Region attribute with East Coast, West Coast, and Central and South derived elements defined on it. The metrics are Revenue, Profit, and Profit Margin.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td></td>
<td>$37,694,553</td>
<td>$6,507,446</td>
<td>50.56%</td>
</tr>
<tr>
<td>West Coast</td>
<td></td>
<td>$14,595,165</td>
<td>$2,435,059</td>
<td>32.81%</td>
</tr>
<tr>
<td>Central and South</td>
<td></td>
<td>$12,495,474</td>
<td>$2,065,833</td>
<td>33.26%</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$3,531,277</td>
<td>$549,392</td>
<td>15.34%</td>
</tr>
</tbody>
</table>

Profit Margin is defined be a smart metric because its definition is \( \text{Profit/Revenue} \). Notice that the values for Profit Margin are much higher than they should be for the derived elements, but the value for the Web attribute element is correct.

By evaluating the derived elements before the Profit Margin smart metric, the correct Profit Margin values are displayed for all derived elements and attribute elements.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Coast</td>
<td></td>
<td>$37,694,553</td>
<td>$6,507,446</td>
<td>17.26%</td>
</tr>
<tr>
<td>West Coast</td>
<td></td>
<td>$14,595,165</td>
<td>$2,435,059</td>
<td>16.68%</td>
</tr>
<tr>
<td>Central and South</td>
<td></td>
<td>$12,495,474</td>
<td>$2,065,833</td>
<td>16.53%</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$3,531,277</td>
<td>$549,392</td>
<td>15.34%</td>
</tr>
</tbody>
</table>

The steps below show you how to modify the evaluation order of derived elements.
To modify the evaluation order of derived elements

1. In Desktop, log in to a project that includes a report with derived elements.

2. Browse to and right-click a report, and select Run. The report is executed in the Report Editor.

3. From the Data menu, select Report Data Options. The Report Data Options dialog box opens.

4. Expand the Calculations category, and then select Evaluation Order.

5. Clear the Show consolidations only check box. This displays all objects on the report that can have their evaluation order modified.

6. In the Evaluation Order column, select the evaluation orders for each object in the report.

   Objects are evaluated from lowest evaluation order number to highest. However, if you define an object to use Default as its evaluation order, it may be evaluated before an object with an evaluation order of 1.

   For example, if you define a derived element with an evaluation order of 1 and a derived metric with an evaluation order of Default, the derived metric is evaluated first. This is because derived metrics are evaluated before derived elements by default. To evaluate a derived element before a derived metric, define the derived metric to have a higher evaluation order than the derived element.

7. Click OK to save your changes and close the Report Data Options dialog box.

Any changes caused by the new evaluation orders are displayed on the report. To view the evaluation order of the various objects on the report, display the report in SQL View.
Introduction

Dynamic aggregation is an OLAP Services feature that allows you to change the level of report aggregation on the fly, while you are reviewing the report results. This feature allows metric values to be aggregated at different levels based on the attributes included on the report without having to re-execute the report against the data warehouse. Dynamic aggregation occurs when you move attributes and metrics between the Report Objects pane and the report grid.
To better understand dynamic aggregation, consider a report that includes two attributes, Region and Category, and two metrics, Revenue and Units Sold. A subset of the report results are shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Units Sold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Books</td>
<td>$251,279</td>
<td>15,715</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$5,307,467</td>
<td>15,278</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$241,405</td>
<td>15,500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$1,112,762</td>
<td>75,809</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$210,320</td>
<td>13,023</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$22,565,830</td>
<td>65,424</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$208,237</td>
<td>13,299</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$194,436</td>
<td>13,063</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>Books</td>
<td>$2,048,826</td>
<td>127,017</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$9,563,073</td>
<td>24,974</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$395,246</td>
<td>25,355</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$368,269</td>
<td>24,817</td>
<td></td>
</tr>
</tbody>
</table>

To remove an attribute from the report without using dynamic aggregation, right-clicking it (the Category attribute is used for this example) and select **Remove from Report**. The data for Revenue and Units Sold is aggregated to the Region level, resulting in the report below.

Notice that the Category attribute is not in the Report Objects pane, and the report remains a **Standard** report as indicated in the bottom right corner.

When you remove an attribute or metric from the report, a message is displayed that indicates the manipulation causes the report to be re-executed. This is because standard metric aggregation needs to be re-executed against the data warehouse, and therefore requires new report SQL to be generated.
Now consider the same report, but this time you use dynamic aggregation to return a different view of the data. Instead of removing Category from the report, Category is moved to the Report Objects pane, as shown in the image below.

Although the report data is exactly the same as the previous report with standard aggregation, dynamic aggregation has the following differences:

- **No re-execution and no new SQL required**: Dynamic aggregation does not require re-execution against the data warehouse. Aggregating data within the report can improve the performance of your system by reducing the load on the data warehouse. Dynamic aggregation also improves performance by returning the new report results as soon as the attribute or metric is moved to Report Objects, instead of having to wait for the results to be returned from the data warehouse.

- **Attributes remain in Report Objects**: Category is now in the Report Objects pane and is no longer displayed with bold formatting. Since Category is part of the report definition, it can be used to define the view of the report. For example, you can build a view filter to return Revenue only for the electronics category. A standard report would have to use the report filter to filter on an attribute not on the report, which would also require re-execution against the data warehouse.

With attributes available in the Report Objects pane, you can easily add them back onto the report grid.

- **OLAP report**: The report is marked as OLAP (as indicated in the bottom right corner of the Report Editor) because of the dynamic aggregation performed.
This chapter discusses the following aspects of dynamic aggregation:

- **Using dynamic aggregation, page 148**
- **Functions used in dynamic aggregation, page 149**
- **View filter effect on dynamic aggregation, page 163**

**Using dynamic aggregation**

To change the aggregation level of a report, you must either remove an attribute from a report entirely, or use dynamic aggregation to remove the attribute from the report grid only. With dynamic aggregation, the attribute is still a part of the report definition and is included in the Report Objects.

To use dynamic aggregation, right-click an attribute on a report grid and select **Remove from Grid**, as shown below.

![Attribute Grid Example](image)

Be careful not to select **Remove from Report**. If you select this option, the attribute is completely removed from the report definition, and the report is re-executed against the data warehouse.

You need the Use Report Objects Window (Desktop) and/or the Web Use Report Objects Window (Web) privileges to use dynamic aggregation. These privileges are part of OLAP Services.

Another benefit of using dynamic aggregation is that the attributes removed from the report grid can be easily included back onto the report grid or
page-by area. To do this, right-click an attribute in the Report Objects pane and select Add to Column, Add to Row, or Add to Page-by to add an attribute from the Report Objects pane onto the report. The options to move an attribute to the report are shown in the image below.

You can also move attribute forms between the Report Objects pane and the report grid. If you only move an attribute form and not the attribute itself, dynamic aggregation is not triggered. For example, if the attribute forms Last Name and First Name for the attribute Customer are displayed on a report, you can move First Name to the Report Objects pane without triggering dynamic aggregation. The same First Name attribute form can be moved back to the report grid without triggering dynamic aggregation.

An attribute must have at least one attribute form displayed to be on the report grid.

Functions used in dynamic aggregation

The dynamic aggregation feature of a metric dictates what function to use when the Analytical Engine must dynamically aggregate the metric. You can modify the dynamic aggregation function from the Subtotals / Aggregation tab of the Metric Editor. A metric has a default aggregation function if the metric is defined with certain functions, as described below:

- A metric defined with one or more of the functions Sum, Count, Minimum, Maximum, and Product has a default aggregation function.

For more information on metrics that have a default dynamic aggregation
function, see *Metrics with default dynamic aggregation functions, page 150*.

- A metric defined with any other functions than the ones listed above does not have a default aggregation function. No aggregation function is used as the default and the dynamic aggregation of one of these metrics returns a null value. The null value is replaced with dashes (---) to indicate that the metric cannot be calculated at the higher level. For more information on functions that do not have a default dynamic aggregation function, see *Metrics that are not dynamically aggregated by default, page 152*.

### Metrics with default dynamic aggregation functions

A metric defined with functions such as Sum or Product can be dynamically aggregated because these functions can be calculated at a higher level than the initial calculation. Any metric defined with one of the following functions has a default dynamic aggregation function that allows the data for the metric to be aggregated at different levels on-the-fly:

- **Sum or Count**: Sum is used as the aggregation function. If the Count function is set to count distinct entries, it cannot use Sum as its dynamic aggregation function and returns null values instead.

- **Minimum**: Minimum (Min) is used as the aggregation function.

- **Maximum**: Maximum (Max) is used as the aggregation function.

- **Product**: Product is used as the aggregation function.

For example, consider the **Dynamic Aggregation - Region - Employee** report from the MicroStrategy Tutorial project. To focus only on those metrics that have default dynamic aggregation functions, move the metrics Standard Deviation of Revenue and Count Distinct (Items Sold) to the
Report Objects pane (for steps, see *Using dynamic aggregation, page 148*). The report results for the Central and Mid-Atlantic regions are shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Employee</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Max Revenue</th>
<th>Min Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Ellerkamp</td>
<td>Nancy</td>
<td>$1,169,245</td>
<td>$53,460.00</td>
<td>$102.00</td>
</tr>
<tr>
<td></td>
<td>Gale</td>
<td>Loren</td>
<td>$2,262,146</td>
<td>$76,740.00</td>
<td>$199.20</td>
</tr>
<tr>
<td></td>
<td>Torrison</td>
<td>Mary</td>
<td>$2,364,993</td>
<td>$76,800.00</td>
<td>$195.30</td>
</tr>
<tr>
<td></td>
<td>Zemlicka</td>
<td>George</td>
<td>$1,116,549</td>
<td>$45,240.00</td>
<td>$82.20</td>
</tr>
<tr>
<td></td>
<td>Subtotals</td>
<td></td>
<td>$6,912,934</td>
<td>$76,800.00</td>
<td>$82.20</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Bernstein</td>
<td>Lawrence</td>
<td>$5,295,910</td>
<td>$232,500.00</td>
<td>$109.95</td>
</tr>
<tr>
<td></td>
<td>Brown</td>
<td>Vernon</td>
<td>$1,803,732</td>
<td>$93,300.00</td>
<td>$20.25</td>
</tr>
<tr>
<td></td>
<td>Concoran</td>
<td>Peter</td>
<td>$1,709,388</td>
<td>$62,400.00</td>
<td>$17.40</td>
</tr>
<tr>
<td></td>
<td>Folks</td>
<td>Adrienne</td>
<td>$5,708,091</td>
<td>$278,520.00</td>
<td>$117.45</td>
</tr>
<tr>
<td></td>
<td>Hollywood</td>
<td>Robert</td>
<td>$5,272,618</td>
<td>$232,920.00</td>
<td>$106.35</td>
</tr>
<tr>
<td></td>
<td>Ingle</td>
<td>Walter</td>
<td>$1,013,388</td>
<td>$46,800.00</td>
<td>$13.00</td>
</tr>
<tr>
<td></td>
<td>Smith</td>
<td>Thomas</td>
<td>$1,183,056</td>
<td>$59,800.00</td>
<td>$11.00</td>
</tr>
<tr>
<td></td>
<td>Young</td>
<td>Sarah</td>
<td>$1,192,641</td>
<td>$47,520.00</td>
<td>$16.00</td>
</tr>
<tr>
<td></td>
<td>Subtotals</td>
<td></td>
<td>$23,178,823</td>
<td>$278,520.00</td>
<td>$11.00</td>
</tr>
</tbody>
</table>

A custom subtotal has been included on this report to display subtotals for the different revenue calculations. For more information on custom subtotals, see the *Reports* chapter of the *MicroStrategy Advanced Reporting Guide* or search for “Steps to define a custom subtotal” in the online help.

The Revenue, Max Revenue, and Min Revenue metrics use total, maximum, and minimum subtotals respectively. These are the same default dynamic aggregation functions that are used for these three metrics because they are built with the Sum, Max, and Min functions respectively. You can verify that this is true by moving Employee to the Report Objects pane, which triggers dynamic aggregation, causing the metrics to be aggregated at the regional level.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Max Revenue</th>
<th>Min Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td></td>
<td>$6,912,934</td>
<td>$76,800.00</td>
<td>$82.20</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td>$23,178,823</td>
<td>$279,520.00</td>
<td>$11.00</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td>$11,375,413</td>
<td>$123,900.00</td>
<td>$42.00</td>
</tr>
<tr>
<td>Northwest</td>
<td></td>
<td>$9,415,861</td>
<td>$205,260.00</td>
<td>$39.25</td>
</tr>
<tr>
<td>South</td>
<td></td>
<td>$5,582,540</td>
<td>$107,760.00</td>
<td>$183.20</td>
</tr>
<tr>
<td>Southeast</td>
<td></td>
<td>$3,140,315</td>
<td>$33,600.00</td>
<td>$42.00</td>
</tr>
<tr>
<td>Southwest</td>
<td></td>
<td>$5,179,306</td>
<td>$55,320.00</td>
<td>$40.00</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$3,581,277</td>
<td>$112,500.00</td>
<td>$279.60</td>
</tr>
</tbody>
</table>
Notice that the values for the Northeast and Mid-Atlantic regions are the same values as the regional subtotals in the report prior to triggering dynamic aggregation. When the three metrics are aggregated at the regional level, each metric uses its default dynamic aggregation function to perform the calculation.

**Metrics that are not dynamically aggregated by default**

The ability to aggregate data at a higher level in memory is useful for quick report interaction and analysis. However, due to their aggregation function, some metrics cannot be dynamically aggregated. For certain metric aggregation functions, recalculating the data at the higher level would yield erroneous or null values if using only the data in the report. The metrics would need to be re-executed against the data warehouse to return correct values.

Functions that cannot be calculated correctly using a dynamic aggregation function have the default dynamic aggregation function set to none. Metrics have their default aggregation function set to none if they are defined with the following functions:

- **Average (Avg)**: Sum of input values divided by number of input values.
- **Count (Distinct=true)**: Number of distinct input values.
- **Geometric mean (Geomean)**: Square root of the product of input values.
- **Median (median)**: Middle value when input values are sorted.
- **Mode (Mode)**: Most frequently found input value.
- **Standard Deviation (Stdev)**: Statistical distribution of input values.
- **Variance (Var)**: Square of the standard deviation of input values.
- **Non-group functions or arithmetic operators**: Metrics defined in this way are called compound metrics.

For information on supporting dynamic aggregation for compound metrics, see *Dynamic aggregation for compound metrics, page 158*. 
When you use dynamic aggregation at a level the Analytical Engine considers erroneous for a metric defined with one or more of the functions listed above, null values, represented by dashes (--), are displayed on the report. For information on changing the display of null values, see Changing the display of null values, page 160.

This section provides the following information related to functions and metrics that do not support dynamic aggregation by default:

- **Example of exceptions to dynamic aggregation, page 153**
- **Returning correct metric values by accessing the data warehouse, page 155**
- **Estimating dynamic aggregation values with different aggregation functions, page 156**
- **Dynamic aggregation for compound metrics, page 158**
- **Changing the display of null values, page 160**

### Example of exceptions to dynamic aggregation

This example uses the *Dynamic Aggregation - Region - Employee* report from the MicroStrategy Tutorial project. To look at only those metrics that do not have default dynamic aggregation functions, move the metrics Revenue, Max Revenue, and Min Revenue to the Report Objects pane. To use dynamic aggregation, the Employee attribute is also moved to the Report Objects pane. The report results are shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Standard Deviation of Revenue</th>
<th>Count Distinct (Items Sold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>$3,393.51</td>
<td></td>
<td>1,440</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>$13,931.67</td>
<td></td>
<td>2,879</td>
</tr>
<tr>
<td>Northeast</td>
<td>$6,524.70</td>
<td></td>
<td>2,160</td>
</tr>
<tr>
<td>Northwest</td>
<td>$5,070.02</td>
<td></td>
<td>1,080</td>
</tr>
<tr>
<td>South</td>
<td>$3,895.12</td>
<td></td>
<td>1,080</td>
</tr>
<tr>
<td>Southeast</td>
<td>$248.86</td>
<td></td>
<td>1,440</td>
</tr>
<tr>
<td>Southwest</td>
<td>$1,950.34</td>
<td></td>
<td>1,800</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td></td>
<td>360</td>
</tr>
</tbody>
</table>
By default, the Standard Deviation of Revenue and Count Distinct (Items Sold) metrics would return null values because they use the Standard Deviation and Count Distinct functions in their metric definitions, respectively. The default dynamic aggregation functions have instead been set to the functions used for their metric definitions.

Now compare these values to the values returned by executing against the data warehouse instead of using dynamic aggregation to calculate the values from the data within the report.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Standard Deviation of Revenue</th>
<th>Count Distinct (Items Sold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td></td>
<td>$9,289.44</td>
<td>360</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td>$23,653.84</td>
<td>360</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td>$11,574.77</td>
<td>360</td>
</tr>
<tr>
<td>Northwest</td>
<td></td>
<td>$21,280.25</td>
<td>360</td>
</tr>
<tr>
<td>South</td>
<td></td>
<td>$9,780.08</td>
<td>360</td>
</tr>
<tr>
<td>Southeast</td>
<td></td>
<td>$3,945.78</td>
<td>360</td>
</tr>
<tr>
<td>Southwest</td>
<td></td>
<td>$5,484.26</td>
<td>360</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$17,419.65</td>
<td>360</td>
</tr>
</tbody>
</table>

You can see that the Count Distinct (Items Sold) value is 360 for the Northeast region, which is far different from the 2,160 value returned for the Northeast region in the report that uses dynamic aggregation. The report above is able to query the data warehouse and show the distinct items sold by all employees in the Northeast region. For example, if Employee A, Employee B, and Employee C all sell one or more wrenches, the item is only counted as one distinct item for the Northeast region.

Dynamic aggregation uses the data available in the report. In this example, all the values for each employee in a given region are simply added together. When the calculation is performed, there is no way of relating which distinct items each employee sold. The calculation results in double-counting distinct items sold by two or more different employees. Rolling up data to a higher level for metrics defined with functions such as Standard Deviation and Average also perform erroneous calculations. For this reason, by default, metrics defined with certain functions return null values instead of erroneous results.
**Returning correct metric values by accessing the data warehouse**

If using dynamic aggregation in a report returns null or erroneous data, you can force the report to re-execute against the data warehouse instead of triggering dynamic aggregation.

For example, move the Employee attribute to the Report Objects pane in the **Dynamic Aggregation - Region - Employee** report in the MicroStrategy Tutorial project. If you have not changed how the report displays null values, you get the results shown below.

<table>
<thead>
<tr>
<th>Name</th>
<th>Count Distinct...</th>
<th>Max Revenue</th>
<th>Min Revenue</th>
<th>Revenue</th>
<th>Standard Deviation of Revenue</th>
<th>Metrics</th>
<th>Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Revenue</td>
<td>$6,912,934</td>
</tr>
<tr>
<td>Region</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standard Deviation</td>
<td>$76,800.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Of Revenue</td>
<td>$82.20</td>
</tr>
<tr>
<td>Central</td>
<td></td>
<td></td>
<td></td>
<td>$23,178,823</td>
<td>$123,900.00</td>
<td>Max Revenue</td>
<td>$11.00</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td></td>
<td></td>
<td>$11,375,413</td>
<td>$205,260.00</td>
<td>Min Revenue</td>
<td>$39.25</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td></td>
<td></td>
<td>$9,415,861</td>
<td>$107,760.00</td>
<td>Count Distinct (Items Sold)</td>
<td>360</td>
</tr>
<tr>
<td>Northwest</td>
<td></td>
<td></td>
<td></td>
<td>$5,152,540</td>
<td>$112,500.00</td>
<td></td>
<td>$279.60</td>
</tr>
<tr>
<td>South</td>
<td></td>
<td></td>
<td></td>
<td>$3,140,316</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td></td>
<td></td>
<td></td>
<td>$5,179,305</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td></td>
<td></td>
<td></td>
<td>$3,581,277</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notice that Employee is still a part of the Report Objects pane.

To execute the report against the data warehouse, right-click the Employee attribute and select **Remove from Report**. The report is re-executed against the data warehouse, and returns the data shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Standard Deviation of Revenue</th>
<th>Max Revenue</th>
<th>Min Revenue</th>
<th>Count Distinct (Items Sold)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>$6,912,934</td>
<td>$9,289.44</td>
<td>$76,800.00</td>
<td>$82.20</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>$23,178,823</td>
<td>$23,658.84</td>
<td>$278,520.00</td>
<td>$11.00</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>$11,375,413</td>
<td>$11,574.77</td>
<td>$123,900.00</td>
<td>$42.00</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Northwest</td>
<td>$9,415,861</td>
<td>$12,280.25</td>
<td>$205,260.00</td>
<td>$39.25</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>South</td>
<td>$5,152,540</td>
<td>$9,780.08</td>
<td>$107,760.00</td>
<td>$183.20</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td>$3,140,316</td>
<td>$3,945.78</td>
<td>$33,600.00</td>
<td>$42.00</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td>$5,179,305</td>
<td>$5,484.26</td>
<td>$55,320.00</td>
<td>$40.00</td>
<td>360</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>$3,581,277</td>
<td>$17,419.65</td>
<td>$112,500.00</td>
<td>$279.60</td>
<td>360</td>
<td></td>
</tr>
</tbody>
</table>

The report is re-executed against the warehouse without using dynamic aggregation only if all the attributes in the Report objects pane are on the report.
Note that taking Employee off the report changes the data definition of the report, instead of the view definition. While you get the correct results, you are no longer taking advantage of dynamic aggregation to perform the calculations. The SQL must be regenerated and executed against the data warehouse to retrieve the results.

**Estimating dynamic aggregation values with different aggregation functions**

If you use a function that does not support dynamic aggregation, you can use a different function to estimate the values using dynamic aggregation. This tactic depends greatly on the function used and the amount of error that is allowable for your report results. For steps on how to change the default aggregation function, see *Changing the default dynamic aggregation function, page 161*.

The Dynamic Aggregation - Region - Employee from the MicroStrategy Tutorial project is used as the base report to illustrate this tactic. Move all the attributes and metrics to the Report Objects pane except for the attribute Region and the metric Standard Deviation of Revenue. When you run this report, null values are displayed for the metric because the metric’s function does not support dynamic aggregation.

You can estimate these values by using the Average function as the dynamic aggregation function. This estimation is possible because the Standard Deviation and Average formulas are similar in this situation.
You can change the dynamic aggregation function in the Subtotals / Aggregation tab of the Metric Editor. The report below shows the results for the metric using the Average dynamic aggregation function on the left, compared to the Standard Deviation function on the right, run against the data warehouse.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Standard Deviation of Revenue</th>
<th>Standard Deviation of Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>$8,670.24</td>
<td>Central</td>
<td>$9,289.44</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>$18,995.93</td>
<td>Mid-Atlantic</td>
<td>$23,658.84</td>
</tr>
<tr>
<td>Northeast</td>
<td>$9,333.97</td>
<td>Northeast</td>
<td>$11,574.77</td>
</tr>
<tr>
<td>Northwest</td>
<td>$20,831.90</td>
<td>Northwest</td>
<td>$21,280.25</td>
</tr>
<tr>
<td>South</td>
<td>$9,087.91</td>
<td>South</td>
<td>$9,780.08</td>
</tr>
<tr>
<td>Southeast</td>
<td>$3,941.30</td>
<td>Southeast</td>
<td>$3,945.78</td>
</tr>
<tr>
<td>Southwest</td>
<td>$5,105.98</td>
<td>Southwest</td>
<td>$5,484.26</td>
</tr>
<tr>
<td>Web</td>
<td>$17,419.65</td>
<td>Web</td>
<td>$17,419.65</td>
</tr>
</tbody>
</table>

Using the average function to perform the dynamic aggregation estimates the metric values to within $1,000 dollars for all but two regions.

The accuracy of data returned when using a different dynamic aggregation function than the function used to define the metric depends on the similarity of the functions. For example, you cannot expect an accurate estimation of values if you use Sum as the dynamic aggregation function for a metric defined with the Standard Deviation function.

You can use any function as the dynamic aggregation function of a metric. However, be aware that not all functions are well suited for dynamic aggregation (see *Metrics that are not dynamically aggregated by default, page 152*).

You can also create your own subtotal to use as the dynamic aggregation function. You cannot directly use a function as the dynamic aggregation function of a metric, you must create a subtotal that uses the function in its definition. For information on creating a subtotal, search the online help for “Creating user-defined subtotals”.

© 2011 MicroStrategy, Inc. Functions used in dynamic aggregation 157
Dynamic aggregation for compound metrics

A compound metric is any metric defined by a formula based on arithmetic operators and non-group functions. Arithmetic operators are +, -, *, and /; non-group functions are OLAP and scalar functions such as RunningSum or Rank. The operators and functions can be applied to facts, attributes, or metrics.

This section covers compound metrics as they are used with dynamic aggregation. For complete information on compound metrics, see the Advanced Reporting Guide.

For example, a metric defined with the formula $\text{Sum(Revenue)} - \text{Sum(Cost)}$ is a compound metric because of the minus (-) operator.

Another example of a compound metric is one defined with the formula $\text{RunningSum(Revenue)}$ because of the RunningSum function.

By default, dynamic aggregation causes compound metrics to return null values. One reason for this is that the calculation of the formula is performed before it is dynamically aggregated, which can cause erroneous or null results depending on the formula (see Functions used in dynamic aggregation, page 149). Another possible reason is that when a metric has a complex formula for its definition, there is no way to assume the aggregation function that should be performed on the data. Rather than returning possibly erroneous results by default, you can decide how the metric handles dynamic aggregation.

For example, consider the Dynamic Aggregation report from the MicroStrategy Tutorial project, which contains the metrics Revenue, Cost, and Profit as well as the attributes Region and Employee. The report is shown below, with Employee off the report grid, but still in the Report Objects pane.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Cost</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td></td>
<td>$6,912,934</td>
<td>$5,827,613</td>
<td>$1,085,321</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td>$23,178,823</td>
<td>$19,171,243</td>
<td>$4,007,581</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td>$11,375,413</td>
<td>$9,365,673</td>
<td>$2,009,740</td>
</tr>
<tr>
<td>Northwest</td>
<td></td>
<td>$9,415,961</td>
<td>$7,780,870</td>
<td>$1,634,990</td>
</tr>
<tr>
<td>South</td>
<td></td>
<td>$5,582,540</td>
<td>$4,602,027</td>
<td>$980,513</td>
</tr>
<tr>
<td>Southeast</td>
<td></td>
<td>$3,140,316</td>
<td>$2,650,191</td>
<td>$490,125</td>
</tr>
<tr>
<td>Southwest</td>
<td></td>
<td>$5,179,305</td>
<td>$4,379,227</td>
<td>$800,078</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$3,581,277</td>
<td>$3,031,885</td>
<td>$549,392</td>
</tr>
</tbody>
</table>
The Profit metric in this case is based on a profit fact stored in the data warehouse. Suppose that your data warehouse only has facts for revenue and cost, but you want to create a metric that calculates profit. One way you can achieve this is by creating a profit metric called Compound Profit that combines the two metrics Revenue and Cost. The metric can be defined as \( \text{Sum}(\text{Revenue}) - \text{Sum}(\text{Cost}) \).

You can also see a comparison of profit margins by including a Profit Margin metric on the report. The definition for Profit Margin is \( \frac{\text{Sum}(\text{Profit})}{\text{Sum}(\text{Revenue})} \). When you add these two metrics to the Dynamic Aggregation report, the report returns the results shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Cost</th>
<th>Profit</th>
<th>Compound Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td></td>
<td>$6,912,934</td>
<td>$5,827,613</td>
<td>$1,085,321</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td>$23,178,823</td>
<td>$19,171,243</td>
<td>$4,007,581</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td>$11,375,413</td>
<td>$9,365,673</td>
<td>$2,009,740</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Northwest</td>
<td></td>
<td>$9,415,861</td>
<td>$7,780,870</td>
<td>$1,634,990</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>South</td>
<td></td>
<td>$5,582,540</td>
<td>$4,602,027</td>
<td>$980,513</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Southeast</td>
<td></td>
<td>$3,140,316</td>
<td>$2,650,191</td>
<td>$490,125</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Southwest</td>
<td></td>
<td>$5,179,305</td>
<td>$4,379,227</td>
<td>$800,078</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td>$3,581,277</td>
<td>$3,031,885</td>
<td>$549,392</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

With Employee in the Report Objects pane and not on the report grid, the metrics are dynamically aggregated to the regional level. The Profit metric is a simple sum of the Profit fact. This calculation can be aggregated from the Employee level to the Region level. By default, the metrics Compound Profit and Profit Margin do not use a dynamic aggregation function.

For Compound Profit to be dynamically aggregated correctly, you change the dynamic aggregation function to Sum. In this case the Sum function can be used to aggregate the data after the subtraction because the order of operations does not matter in a formula with only sum and subtract. For more information on changing the default dynamic aggregation function, see *Changing the default dynamic aggregation function, page 161*.

For Profit Margin, you cannot choose Sum as the dynamic aggregation function, because the definition of the metric includes a division. If Sum is chosen, the division is performed first and then these values are added together, which would use the formula \( \text{Sum}(\text{Profit}/\text{Revenue}) \). Recall that the definition of the metric is \( \frac{\text{Sum}(\text{Profit})}{\text{Sum}(\text{Revenue})} \), performing the sum aggregations first and then dividing the sums.

To return valid results in this case, you can calculate the subtraction after dynamic aggregation. You can achieve this functionality by defining the
compound metric as a smart metric. After you define both of the compound metrics as smart metrics, the correct results are returned, as shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Cost</th>
<th>Profit</th>
<th>Compound Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>$6,912,934</td>
<td>$5,827,613</td>
<td>$1,085,321</td>
<td>$1,085,321</td>
<td></td>
<td>15.70%</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>$23,178,823</td>
<td>$19,171,243</td>
<td>$4,007,581</td>
<td>$4,007,581</td>
<td></td>
<td>17.29%</td>
</tr>
<tr>
<td>Northeast</td>
<td>$11,375,413</td>
<td>$9,365,673</td>
<td>$2,009,740</td>
<td>$2,009,740</td>
<td></td>
<td>17.67%</td>
</tr>
<tr>
<td>Northwest</td>
<td>$9,415,861</td>
<td>$7,780,870</td>
<td>$1,634,990</td>
<td>$1,634,990</td>
<td></td>
<td>17.36%</td>
</tr>
<tr>
<td>South</td>
<td>$5,582,540</td>
<td>$4,602,027</td>
<td>$980,513</td>
<td>$980,513</td>
<td></td>
<td>17.56%</td>
</tr>
<tr>
<td>Southeast</td>
<td>$3,140,316</td>
<td>$2,650,191</td>
<td>$490,125</td>
<td>$490,125</td>
<td></td>
<td>15.61%</td>
</tr>
<tr>
<td>Southwest</td>
<td>$5,179,305</td>
<td>$4,379,227</td>
<td>$800,078</td>
<td>$800,078</td>
<td></td>
<td>15.45%</td>
</tr>
<tr>
<td>Web</td>
<td>$3,581,277</td>
<td>$3,031,885</td>
<td>$549,392</td>
<td>$549,392</td>
<td></td>
<td>15.34%</td>
</tr>
</tbody>
</table>

You can define a metric as a smart metric using the procedure below. For an introduction to smart metrics, see *Smart metrics, page 142*.

**To define a metric as a smart metric**

1. In MicroStrategy Desktop, right-click a metric and select **Edit**. The Metric Editor opens.

2. On the **Subtotals / Aggregation** tab, select the **Allow Smart Metric** check box.

3. Click **Save and Close** to save your changes and close the Metric Editor.

**Changing the display of null values**

You can change the default display of null values caused by incorrect aggregation for your reports.

**To change the display of aggregation null values**

1. Open a report.

2. From the **Data** menu, select **Report Data Options**. The Report Data Options dialog box opens.

3. Expand the **Display** category, and then select **Null Values**. The Display - Null Values tab opens.
4 Under Aggregation null values, clear **Use Default**.

5 Replace “--” with the symbol you want to use for null values, for example, `00`, `null`, `blank`, and so on.

### Changing the default dynamic aggregation function

The dynamic aggregation function is used to aggregate the data whenever the metric is dynamically aggregated, regardless of what function is used in the metric definition. You can change the default function used for dynamic aggregation in the Subtotals / Aggregation tab of the Metric Editor, as shown below.

The validity of the data depends on whether the function can correctly calculate the data within the report. For this reason, it is recommended that you evaluate your report requirements and consider the report results before you make the function selection for dynamic aggregation.

The functions that can be used as the dynamic aggregation function are:

- Average
- Count (Distinct=true)
- Geometric Mean
- Maximum
• Median
• Minimum
• Mode
• Product
• Standard Deviation
• Sum
• Variance
• User-defined function

You cannot directly use a function as the dynamic aggregation function of a metric. You must use or create a subtotal that uses the function in its definition. For information on creating a subtotal, search the online help for “Creating user-defined subtotals”.

Metrics defined with certain functions use a default dynamic aggregation function that returns the correct results in most situations. For example, a metric defined with the Sum function uses the Sum function to dynamically aggregate its data. You can change the dynamic aggregation function for this type of metric, but it is recommended that you do not change these default functions as this can cause erroneous or null results for the metrics in a report. For more information on metrics with default dynamic aggregation functions, see Metrics with default dynamic aggregation functions, page 150.

Aggregating data from a report with certain functions can return erroneous or null results, and therefore the default dynamic aggregation function is set to none. You can set the default dynamic aggregation function so that these metrics return data instead of null values. For example, if a metric is defined with Standard Deviation, you can change the function used for dynamic aggregation from Default to Standard Deviation.

You can, however, use a function that does not match the function or functions used to define the metric. To see an example that uses this technique, see Estimating dynamic aggregation values with different aggregation functions, page 156.

To change the default dynamic aggregation function

1 In MicroStrategy Desktop, right-click a metric and select Edit. The Metric Editor opens.
2 On the **Subtotals / Aggregation** tab, select a function from the **Dynamic aggregation function** drop-down list.

3 Click **Save and Close** to save your changes and close the Metric Editor.

### View filter effect on dynamic aggregation

Dynamic aggregation can be affected by the use of view filters used in a report. For information on view filter effects on dynamic aggregation and other reporting features, see *View filter effects on reporting features, page 219.*
DERIVED METRICS
Creating Metrics On-The-Fly

Introduction

A derived metric is a calculation based on metric data included in the report definition. You can use derived metrics to perform column math, that is, calculations on metrics included in the report definition, without regenerating or re-executing SQL against the data warehouse.

Derived metrics are easy to create and can be made on-the-fly when you are viewing a report. A typical case is to create a derived metric to perform calculations between columns of metric data. For example, a derived metrics can subtract the data of one metric by the data of another metric (Metric1 - Metric2) to obtain a new metric calculation.

To see how derived metrics can be used in reports, consider the Derived Metric report from the MicroStrategy Tutorial project. The report includes a
A derived metric that is defined as $\text{Revenue}/1000$, which returns the Revenue values in thousands of dollars, as shown below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Employee</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Cost</th>
<th>Profit</th>
<th>Derived Revenue (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Ellerkamp</td>
<td>Nancy</td>
<td>$1,169,245</td>
<td>$987,540</td>
<td>$181,705</td>
<td>$1,169</td>
</tr>
<tr>
<td></td>
<td>Gale</td>
<td>Loren</td>
<td>$2,262,146</td>
<td>$1,907,140</td>
<td>$355,006</td>
<td>$2,262</td>
</tr>
<tr>
<td></td>
<td>Torrison</td>
<td>Mary</td>
<td>$2,364,993</td>
<td>$1,992,733</td>
<td>$372,260</td>
<td>$2,365</td>
</tr>
<tr>
<td></td>
<td>Zemlicka</td>
<td>George</td>
<td>$1,116,549</td>
<td>$940,199</td>
<td>$176,349</td>
<td>$1,117</td>
</tr>
</tbody>
</table>

Notice that in the Report Objects pane to the left of the report (and shown below), the derived metric is preceded by an $\text{fx}$ symbol, instead of the usual metric symbol, meaning this is a new metric based on the existing metrics in the report.

Since derived metrics are created within a report, they can only be used for the report in which they are created. Derived metrics cannot be saved as individual objects in the project, and therefore cannot be applied to other reports in the project.

This chapter discusses the following topics on derived metrics:

- Creating a derived metric, page 167
- Editing derived metrics, page 180
- Formatting derived metrics, page 181
- Deleting derived metrics, page 181
- View filter effects on derived metrics, page 182
Creating a derived metric

Derived metrics can be created as part of the design or analysis of a report. This allows you to create these metrics while focusing on a given report.

You can create a derived metric in any report, as long as you have an OLAP Services license. This includes reports connected to an Intelligent Cube, reports returning information directly from a data warehouse, Freeform SQL reports returning information from an Excel spreadsheet, and so on.

Best practices

Follow the guidelines below when creating derived metrics:

• You can define derived metrics with objects in the Report Objects pane. The Report Objects are the components included in the report definition, even if they are not displayed on the report grid.

• A derived metric can be defined only with the metrics in the report definition. The Input Metric Formula dialog box where you create derived metrics allows you to choose only from objects included in the report definition, as shown below.

• Attributes included in the report definition are also available to use in the definition of a derived metric. If you use an attribute as part of the metric definition, the metric calculation requires new report SQL to be executed against the data warehouse. This re-execution is not required for derived metrics that only use metrics in their definitions.
• You can use one or more functions or operators in the formula of the derived metric. Click the $f_x$ button to access available functions and operators.

• You can change the level at which a derived metric is calculated. For example, the derived metric $\text{sum}(M1) \{\text{Attribute1}\}$ is calculated at the Attribute1 level. For information on metric levels, see the Advanced Reporting Guide.

• Any user can modify a derived metric after report execution, since its formula is visible to all users. If a derived metric should not be modified by end users, create the metric in the Metric Editor and add it to the report as a normal metric.

• Transformation objects cannot be used with derived metrics because they require SQL to be re-executed against the data warehouse.

• View filters can filter the results of a derived metric. A view filter is an additional filter applied in memory to the report results to restrict the amount of data displayed on the report. For more information on view filters, see Chapter 7, View Filters.

• A derived metric is dependent on any report objects that are included in the derived metric's definition. Because of this dependency, you cannot remove an object from the report that is used in a derived metric definition.

  If you try to remove an object from the report, a message is displayed that indicates you cannot remove the object because it is being used by the derived metric. You can however move an object off the report grid so that it only appears in the Report Objects pane. This allows you to hide the object from the report grid and still support any derived metrics that are dependent on it.

You can create derived metrics with the following methods described in this section:

• Creating a derived metric using the Input Metric Formula dialog box, page 169: You can create any type of derived metric by defining derived metric expressions using the Input Metric Formula dialog box.

• Using rank and percent-to-total metric analysis, page 173: You can quickly create derived metrics that display the percent in relation to a selected total of each item affected by the metric or display a ranking number to the metric values for a given attribute. These can be quickly created using shortcut metrics.
Creating a derived metric using the Input Metric Formula dialog box

The Input Metric Formula dialog box allows you to use functions, operators, and report objects to create derived metrics. You can create any type of derived metric by defining derived metric expressions using the Input Metric Formula dialog box. This includes shortcut metrics that are described in *Using rank and percent-to-total metric analysis, page 173.*

The procedure below describes the high-level steps for creating a derived metric with the Input Metric Formula dialog box. For an example of creating a derived metric, see *Example: Average profit per customer with transactions, page 171.*

**Prerequisites**

- You need the Create Derived Metrics (Desktop) and/or the Web Create Derived Metrics (Web) privileges. These privileges are part of OLAP Services.

**To create a derived metric with the Input Metric Formula dialog box**

For details on each option for any of the steps below, click Help.

1. Open a report in MicroStrategy Desktop or Web.
2. Open the Input Metric Formula dialog box to create a new metric by performing one of the following steps:
   - In Desktop: From the **Insert** menu, select **New Metric**.
   - In Web: From the **Data** menu, select **Insert New Metric**.
3. The pane on the left displays the Report Objects, which shows the components (attributes, attribute forms, metrics, custom groups, consolidations, and so on) included in the report, even if the components are not displayed in the report grid.
   - In Desktop, double-click or drag-and-drop an object to use it to define the derived metric.
   - In Web, select an object and click the arrow to use it to define the derived metric. This moves the object to the window on the right.
4 Add functions and operators by typing their syntax or characters. You can also click the fx button to open the Insert Function Wizard, which guides you through adding functions and operators.

5 Continue to add report components, functions, operators, constant values, and other valid metric formula objects to complete your formula.

6 To add the level at which to calculate the metric, enclose the metric formula in parentheses. Type the attribute name between curly braces {} after the metric formula. If the attribute name contains a space, enclose the name within brackets []. For example, ([Unit Cost] * [Units Sold]) {[Customer Name]}, where Unit Cost and Units Sold are metrics and Customer Name is an attribute. This is a valid expression.

For more information on metric levels, see the Advanced Reporting Guide.

7 Once you have created the desired expression, you can determine whether the expression is valid by performing one of the following steps:

- In Desktop: Click Validate. An error is displayed if the expression is invalid.
- In Web: Click Apply. An error is displayed if the expression is invalid.

8 In the Metric Name (Desktop) or Name (Web) field, enter a name for the new metric.

When naming a MicroStrategy object, you must follow the naming convention rules for your particular database platform. Using a word reserved by your database platform can result in an error when the report is executed. Refer to your database documentation for a list of these database-reserved words.

9 Click OK to create the new derived metric for your report. The Input Metric Formula dialog box closes and the derived metric is added to the report.

To format the values or headers of a derived metric, see Formatting derived metrics, page 181.
**Example: Average profit per customer with transactions**

In this example, a derived metric is created for the Customer Profitability - By Tenure report from the Customer Analysis Module. This report provides monthly trends in customer information based on tenure, as shown below.

<table>
<thead>
<tr>
<th>Month</th>
<th>Tenure Ranges</th>
<th>Metrics</th>
<th>Active Customers</th>
<th>Customers with Transactions</th>
<th>Revenue</th>
<th>Profit</th>
<th>Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 2007</td>
<td>1 Month</td>
<td></td>
<td>38</td>
<td>30</td>
<td>$15,197.75</td>
<td>$988.91</td>
<td>6.51%</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td></td>
<td>40</td>
<td>28</td>
<td>$6,181.77</td>
<td>$341.72</td>
<td>5.53%</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td></td>
<td>37</td>
<td>30</td>
<td>$17,201.78</td>
<td>$1,292.69</td>
<td>7.51%</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td></td>
<td>40</td>
<td>33</td>
<td>$14,241.71</td>
<td>$1,203.63</td>
<td>8.45%</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td></td>
<td>38</td>
<td>27</td>
<td>$9,305.84</td>
<td>$795.54</td>
<td>8.55%</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td></td>
<td>1,106</td>
<td>865</td>
<td>$273,794.53</td>
<td>$17,493.32</td>
<td>6.39%</td>
</tr>
<tr>
<td>Nov 2007</td>
<td>1 Month</td>
<td></td>
<td>40</td>
<td>27</td>
<td>$11,447.74</td>
<td>$900.94</td>
<td>7.87%</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td></td>
<td>37</td>
<td>29</td>
<td>$8,785.75</td>
<td>$357.27</td>
<td>4.01%</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td></td>
<td>39</td>
<td>28</td>
<td>$5,001.27</td>
<td>$291.32</td>
<td>5.82%</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td></td>
<td>36</td>
<td>31</td>
<td>$5,801.76</td>
<td>$347.67</td>
<td>5.99%</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td></td>
<td>39</td>
<td>32</td>
<td>$10,344.24</td>
<td>$505.92</td>
<td>4.89%</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td></td>
<td>1,142</td>
<td>862</td>
<td>$279,610.09</td>
<td>$16,257.96</td>
<td>5.81%</td>
</tr>
<tr>
<td>Dec 2007</td>
<td>1 Month</td>
<td></td>
<td>40</td>
<td>35</td>
<td>$10,554.66</td>
<td>$581.89</td>
<td>5.51%</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td></td>
<td>38</td>
<td>34</td>
<td>$17,786.73</td>
<td>$1,157.86</td>
<td>6.51%</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td></td>
<td>37</td>
<td>23</td>
<td>$10,071.25</td>
<td>$775.26</td>
<td>7.70%</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td></td>
<td>37</td>
<td>33</td>
<td>$20,653.22</td>
<td>$1,169.74</td>
<td>5.66%</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td></td>
<td>36</td>
<td>30</td>
<td>$10,877.18</td>
<td>$540.68</td>
<td>4.97%</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td></td>
<td>1,175</td>
<td>866</td>
<td>$310,550.97</td>
<td>$19,642.77</td>
<td>6.33%</td>
</tr>
</tbody>
</table>

A derived metric is created to display the average profit generated per customer with transactions. The procedure below describes how to create and format this derived metric for the report shown above.

**Prerequisites**

- You need the Create Derived Metrics (Desktop) and/or the Web Create Derived Metrics (Web) privileges. These privileges are part of OLAP Services.

**To create an average profit per customer with transactions derived metric**

1. In MicroStrategy Desktop, log in to the Customer Analysis Module.
2. Browse to the Customer Profitability - By Tenure report, right-click the report, and then select Run Report. A Choose from all elements of ‘Quarter’ prompt opens.
3 Keep the default prompt answer of **2007 Q4**, and click **Finish**. The Customer Profitability - By Tenure report opens in Grid View.

4 From the **Insert** menu in the Report Editor, select **New Metric**. The Input Metric Formula dialog box opens.

5 In the **Metric Name** field, type **Average Profit Per Customer with Transactions**.

   **To create the expression for the derived metric**

   You create this expression using the following steps:

   \[(\text{Profit} / [\text{Customers with Transactions}])\]

6 Drag-and-drop the **Profit** metric from the list of Report Objects in the left pane to the metric definition area on the right.

7 Click the division sign (\(\div\)) on the operators and functions toolbar. The division sign should be inserted to the right of the Profit metric.

8 Drag-and-drop the **Customer with Transactions** metric from the list of Report Objects to the metric definition area on the right. It should be placed on the right side of the division sign (\(\div\)).

9 Click **Validate**. A green check is displayed in the right-hand corner below the metric definition area, indicating that the metric definition is valid.

10 Click **OK** to return to the report. The new metric column for the derived metric is displayed in the report.

**To format the values of the new derived metric**

The metric is included without any value formatting. Follow the steps below to format the metric values to be displayed with a currency format.

11 Right-click the **Average Profit per Customer with Transactions** derived metric, point to **Formatting**, and select **Average Profit per Customer with Transactions Values**. The Format Cells dialog box opens.

12 On the **Number** tab, from the Category pane, select **Currency**.
13 Keep all defaults for the Currency format and click **OK**. You are returned to the report, which is shown below.

<table>
<thead>
<tr>
<th>Month</th>
<th>Tenure Ranges</th>
<th>Active Customers</th>
<th>Customers with Transactions</th>
<th>Revenue</th>
<th>Profit</th>
<th>Margin</th>
<th>Average Profit Per Customer with Transactions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 2007</td>
<td>1 Month</td>
<td>38</td>
<td>30</td>
<td>$15,197.75</td>
<td>$988.91</td>
<td>6.51%</td>
<td>$32.96</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td>40</td>
<td>28</td>
<td>$5,101.77</td>
<td>$341.72</td>
<td>5.53%</td>
<td>$12.20</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td>37</td>
<td>30</td>
<td>$17,201.78</td>
<td>$1,292.69</td>
<td>7.51%</td>
<td>$43.09</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td>40</td>
<td>33</td>
<td>$14,241.71</td>
<td>$1,203.63</td>
<td>8.45%</td>
<td>$36.47</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td>38</td>
<td>27</td>
<td>$9,305.04</td>
<td>$795.54</td>
<td>6.55%</td>
<td>$29.46</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td>1,106</td>
<td>865</td>
<td>$273,794.53</td>
<td>$17,493.32</td>
<td>6.39%</td>
<td>$20.22</td>
</tr>
<tr>
<td>Nov 2007</td>
<td>1 Month</td>
<td>40</td>
<td>27</td>
<td>$11,447.74</td>
<td>$900.04</td>
<td>7.87%</td>
<td>$33.37</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td>37</td>
<td>29</td>
<td>$8,785.75</td>
<td>$357.27</td>
<td>4.07%</td>
<td>$12.32</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td>39</td>
<td>28</td>
<td>$5,001.27</td>
<td>$291.32</td>
<td>5.82%</td>
<td>$10.40</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td>36</td>
<td>31</td>
<td>$5,801.76</td>
<td>$347.67</td>
<td>5.99%</td>
<td>$11.22</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td>39</td>
<td>32</td>
<td>$10,344.24</td>
<td>$505.92</td>
<td>4.89%</td>
<td>$15.81</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td>1,142</td>
<td>862</td>
<td>$279,610.09</td>
<td>$15,257.96</td>
<td>5.91%</td>
<td>$19.86</td>
</tr>
<tr>
<td>Dec 2007</td>
<td>1 Month</td>
<td>40</td>
<td>35</td>
<td>$10,554.66</td>
<td>$581.89</td>
<td>5.51%</td>
<td>$15.63</td>
</tr>
<tr>
<td></td>
<td>2 Months</td>
<td>38</td>
<td>34</td>
<td>$17,786.73</td>
<td>$1,157.86</td>
<td>6.51%</td>
<td>$34.05</td>
</tr>
<tr>
<td></td>
<td>3 Months</td>
<td>37</td>
<td>23</td>
<td>$10,071.25</td>
<td>$775.26</td>
<td>7.70%</td>
<td>$33.71</td>
</tr>
<tr>
<td></td>
<td>4 Months</td>
<td>37</td>
<td>33</td>
<td>$20,653.22</td>
<td>$1,169.74</td>
<td>5.66%</td>
<td>$35.45</td>
</tr>
<tr>
<td></td>
<td>5 Months</td>
<td>36</td>
<td>30</td>
<td>$10,877.18</td>
<td>$540.68</td>
<td>4.97%</td>
<td>$18.02</td>
</tr>
<tr>
<td></td>
<td>&gt; 5 Months</td>
<td>1,175</td>
<td>866</td>
<td>$310,550.97</td>
<td>$19,642.77</td>
<td>6.32%</td>
<td>$22.60</td>
</tr>
</tbody>
</table>

**Using rank and percent-to-total metric analysis**

You can create derived metrics that display the percent in relation to a selected total of each item affected by the metric or display a ranking number to the metric values for a given attribute.

These can be quickly created using shortcut metrics. **Shortcut metrics** are a set of quick metrics you can create on the fly for a given report, based on the metrics already in the report. Shortcut metrics include:

- Percent-to-total shortcut metrics, which display the percent in relation to a selected total of each item affected by the metric. To create this type of shortcut metric, see *Creating a percent-to-total shortcut metric*, page 174.

- Rank shortcut metrics, which apply a ranking number to the metric values for a given attribute. To create this type of shortcut metric, see *Creating a rank shortcut metric*, page 177.

- Transformation shortcut metrics, which apply offset values, such as "four months ago," to an attribute that you select.
All the shortcut metrics are derived metrics, except for transformation shortcut metrics. Transformation shortcut metrics must be calculated in SQL and are therefore only available to users that have the Modify the List of Report Objects (Use Object Browser) privilege. Because they must be calculated, the report re-executes to display the new metric. Transformation metrics are not covered in this guide, see the Advanced Reporting Guide. Reports are not re-executed when other shortcut metrics are created.

Creating a percent-to-total shortcut metric

A percent-to-total shortcut metric displays the percent in relation to a selected total of each item affected by the metric. Use a percent-to-total shortcut metric to show cell-level values as percents of an accumulated row or column total. The metric can also show a total by page, show a total for each value of the attribute, or show the grand total.

For example, a report contains Sales Region, Sales Representative, and the Revenue by Sales Representative metric as shown below.

<table>
<thead>
<tr>
<th>Sales Region</th>
<th>Sales Representative</th>
<th>Metrics</th>
<th>Revenue by Sales Representative</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>Alac Berg</td>
<td></td>
<td>$4,355,000</td>
</tr>
<tr>
<td></td>
<td>Bonnie Zastrow</td>
<td></td>
<td>$3,130,000</td>
</tr>
<tr>
<td></td>
<td>Carole Alaniz</td>
<td></td>
<td>$3,160,000</td>
</tr>
<tr>
<td></td>
<td>Danny Leavitt</td>
<td></td>
<td>$2,915,000</td>
</tr>
<tr>
<td></td>
<td>Erin Dull</td>
<td></td>
<td>$2,325,000</td>
</tr>
<tr>
<td></td>
<td>Gerald Blubaugh</td>
<td></td>
<td>$3,310,000</td>
</tr>
<tr>
<td>Canada</td>
<td>Jori Oakes</td>
<td></td>
<td>$3,755,000</td>
</tr>
<tr>
<td></td>
<td>Mko Kemna</td>
<td></td>
<td>$3,785,000</td>
</tr>
<tr>
<td></td>
<td>oral Beason</td>
<td></td>
<td>$4,035,000</td>
</tr>
<tr>
<td></td>
<td>Palmer Buzzario</td>
<td></td>
<td>$2,885,000</td>
</tr>
<tr>
<td></td>
<td>Anany Harlan</td>
<td></td>
<td>$2,920,000</td>
</tr>
<tr>
<td></td>
<td>Benoit Ward</td>
<td></td>
<td>$2,485,000</td>
</tr>
<tr>
<td>Europe</td>
<td>Constance Imes</td>
<td></td>
<td>$3,200,000</td>
</tr>
<tr>
<td></td>
<td>Dome Bachmeier</td>
<td></td>
<td>$1,665,000</td>
</tr>
<tr>
<td></td>
<td>Enzo Kellaway</td>
<td></td>
<td>$3,375,000</td>
</tr>
<tr>
<td></td>
<td>Gordon Cutting</td>
<td></td>
<td>$2,100,000</td>
</tr>
<tr>
<td></td>
<td>Josh Hadley</td>
<td></td>
<td>$2,115,000</td>
</tr>
<tr>
<td></td>
<td>Anthony Vanderwerff</td>
<td></td>
<td>$2,240,000</td>
</tr>
</tbody>
</table>

Along with revenue totals for each sales representative, you can also highlight a sales representative’s percent contribution to revenue for their
sales region, as well as company-wide. You insert two percent-to-total shortcut metrics to display this information, as shown in the report below:

<table>
<thead>
<tr>
<th>Sales Region</th>
<th>Sales Representative</th>
<th>Metrics</th>
<th>Revenue by Sales Representative</th>
<th>Percent to Total (Revenue by Sales Representative)</th>
<th>Percent to Grand Total (Revenue by Sales Representative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>Alec Berg</td>
<td></td>
<td>$4,355,000</td>
<td>22.69%</td>
<td>8.10%</td>
</tr>
<tr>
<td></td>
<td>Bonnie Zastrow</td>
<td></td>
<td>$3,130,000</td>
<td>16.31%</td>
<td>5.82%</td>
</tr>
<tr>
<td></td>
<td>Carole Alaniz</td>
<td></td>
<td>$3,160,000</td>
<td>16.46%</td>
<td>5.88%</td>
</tr>
<tr>
<td></td>
<td>Danny Leavitt</td>
<td></td>
<td>$2,915,000</td>
<td>15.19%</td>
<td>5.42%</td>
</tr>
<tr>
<td></td>
<td>Erin Dull</td>
<td></td>
<td>$2,325,000</td>
<td>12.11%</td>
<td>4.33%</td>
</tr>
<tr>
<td></td>
<td>Gerald Blubaugh</td>
<td></td>
<td>$3,310,000</td>
<td>17.24%</td>
<td>6.16%</td>
</tr>
<tr>
<td>Canada</td>
<td>Jori Oakes</td>
<td></td>
<td>$3,755,000</td>
<td>18.90%</td>
<td>6.99%</td>
</tr>
<tr>
<td></td>
<td>Miko Kemna</td>
<td></td>
<td>$3,785,000</td>
<td>19.05%</td>
<td>7.04%</td>
</tr>
<tr>
<td></td>
<td>Oral Beason</td>
<td></td>
<td>$4,035,000</td>
<td>20.31%</td>
<td>7.51%</td>
</tr>
<tr>
<td></td>
<td>Palmer Buzzério</td>
<td></td>
<td>$2,885,000</td>
<td>14.52%</td>
<td>5.37%</td>
</tr>
<tr>
<td></td>
<td>Anany Harlan</td>
<td></td>
<td>$2,920,000</td>
<td>14.70%</td>
<td>5.43%</td>
</tr>
<tr>
<td></td>
<td>Benoît Ward</td>
<td></td>
<td>$2,485,000</td>
<td>12.51%</td>
<td>4.62%</td>
</tr>
<tr>
<td>Europe</td>
<td>Constance Ims</td>
<td></td>
<td>$3,200,000</td>
<td>21.78%</td>
<td>5.95%</td>
</tr>
<tr>
<td></td>
<td>Dome Bachmeier</td>
<td></td>
<td>$1,665,000</td>
<td>11.33%</td>
<td>3.10%</td>
</tr>
<tr>
<td></td>
<td>Enzo Kellaway</td>
<td></td>
<td>$3,375,000</td>
<td>22.97%</td>
<td>6.28%</td>
</tr>
<tr>
<td></td>
<td>Gordon Cutting</td>
<td></td>
<td>$2,100,000</td>
<td>14.29%</td>
<td>3.91%</td>
</tr>
<tr>
<td></td>
<td>Josh Hadley</td>
<td></td>
<td>$2,115,000</td>
<td>14.39%</td>
<td>3.93%</td>
</tr>
<tr>
<td></td>
<td>Anthony Vanderwerff</td>
<td></td>
<td>$2,240,000</td>
<td>15.24%</td>
<td>4.17%</td>
</tr>
</tbody>
</table>

This report provides analysis into sales representatives’ performances at both the regional and company-wide level.

The following rules apply to percent-to-total shortcut metrics:

- Row and column percent-to-totals refer to the extreme-left and topmost positions, respectively.
- Page percent-to-totals affect all attributes on a page.
- Percent to All -> A1, where A1 is an attribute, indicates that the calculation is performed across all elements of that attribute. An example is percent to all stores.
- If a report does not contain attributes at a given percent-to-total level, the level is unavailable for that report.
- In some cases, two or more percent-to-total calculations at different logical levels yield the same result. For example, Percent-to-Page Total data can be the same as Percent-to-Grand Total data in a single-page report.
The level of a percent-to-total shortcut metric remains constant once the metric has been calculated; subsequent manipulation of objects on the report does not affect it.

The steps below show you how to create a percent-to-total shortcut metric.

**Prerequisites**

- You need the Create Derived Metrics (Desktop) and/or the Web Create Derived Metrics (Web) privileges. These privileges are part of OLAP Services.

**To create a percent-to-total shortcut metric**

1. Open a report in MicroStrategy Desktop or Web.
2. Right-click the metric to see data displayed as percents of a total, point to **Insert** (Desktop) or **Insert Metric** (Web), point to **Percent to Total**, and then select the portions of the report for which percent-to-total data is to be calculated. Options are:
   - **Over Rows**: Displays values in each row of the report as percents of an attribute row total. Row percent-to-totals refer to the attribute in the extreme-left position. This option is best suited for reports that display attributes in the rows and metrics in the columns.
   - **Over Columns**: Displays values in each column as percents of an attribute column total. Column percent-to-totals refer to the attribute in the topmost position. This option is best suited for reports that display attributes in the columns and metrics in the rows.
   - **Page Total**: Displays all values on a page as percents of that page’s total. This option is available only on reports that include an object in the page-by area.
   - **Grand Total**: Displays all values in a report as percents of the grand total for that report.
   - **Total for each**: Displays all values pertaining to a given report component (an attribute, for example) as percents of the total accumulated for that component.

The report is updated showing your new shortcut metric that displays the percent-to-totals for the components selected. To edit a shortcut metric, see *Editing derived metrics, page 180*. To format the values or headers of a derived metric, see *Formatting derived metrics, page 181*. 
Creating a rank shortcut metric

A rank shortcut metric applies a ranking number to the metric values for a given attribute. Use the rank function to show the relative position of a given cell content in relation to other values for a report component. The rank shortcut metric provides break-by options for each attribute on the report.

For example, a report contains Sales Region, Sales Representative, and the Revenue by Sales Representative metric as shown below.

This report was created in the Sales Force Analysis Module.

<table>
<thead>
<tr>
<th>Sales Region</th>
<th>Sales Representative</th>
<th>Metrics</th>
<th>Revenue by Sales Representative</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>Alec Berg</td>
<td>$4,355,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bonnie Zastrow</td>
<td>$3,130,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Carole Alaniz</td>
<td>$3,160,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Danny Leavitt</td>
<td>$2,915,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Erin Dull</td>
<td>$2,325,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gerald Blubaugh</td>
<td>$3,310,000</td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td>Jori Oakes</td>
<td>$3,755,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Miko Kemna</td>
<td>$3,785,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oral Beason</td>
<td>$4,035,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Palmer Buzzero</td>
<td>$2,885,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Anany Harlan</td>
<td>$2,920,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Benoit Ward</td>
<td>$2,485,000</td>
<td></td>
</tr>
<tr>
<td>Europe</td>
<td>Constance Imes</td>
<td>$3,200,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dome Bachmeier</td>
<td>$1,665,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enzo Kellaway</td>
<td>$3,375,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gordon Cutting</td>
<td>$2,100,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Josh Hadley</td>
<td>$2,115,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Anthony Vanderwerff</td>
<td>$2,240,000</td>
<td></td>
</tr>
</tbody>
</table>

Along with revenue totals for each sales representative, you can also highlight sales representatives’ relative performances company-wide. You
insert a rank shortcut metric to display this information, as shown in the report below:

<table>
<thead>
<tr>
<th>Sales Region</th>
<th>Sales Representative</th>
<th>Metrics</th>
<th>Revenue by Sales Representative</th>
<th>Rank (Revenue by Sales Representative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>Alan Berg</td>
<td>$4,355,000</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gerald Blubaugh</td>
<td>$3,310,000</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Carole Alaniz</td>
<td>$3,160,000</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bonnie Zastrow</td>
<td>$3,130,000</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Danny Leavitt</td>
<td>$2,915,000</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Erin Dull</td>
<td>$2,325,000</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td>Oral Beason</td>
<td>$4,035,000</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Miko Kemna</td>
<td>$3,785,000</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jori Oakes</td>
<td>$3,755,000</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Anany Harlan</td>
<td>$2,920,000</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Palmer Buzzero</td>
<td>$2,885,000</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Benoit Ward</td>
<td>$2,485,000</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>Europe</td>
<td>Enzo Kellaway</td>
<td>$3,375,000</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constance Imes</td>
<td>$3,200,000</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Anthony Vanderwerff</td>
<td>$2,240,000</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Josh Hadley</td>
<td>$2,115,000</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gordon Cutting</td>
<td>$2,100,000</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dome Bachmeier</td>
<td>$1,665,000</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

This report provides analysis into sales representatives’ relative performances company-wide. If you use Sales Region as a break-by, employees would be ranked within their sales regions.

By default, a rank shortcut metric ranks values from low to high in ascending order. Therefore, the lowest value has a rank of 1 and the highest value has a rank equal to the total number of items being ranked within a given break-by grouping. You can change the metric to rank in descending order by editing the metric.

Refer to the procedure below for steps to create a rank shortcut metric.

**Prerequisites**

- You need the Create Derived Metrics (Desktop) and/or the Web Create Derived Metrics (Web) privileges. These privileges are part of OLAP Services.
To create a rank shortcut metric

1. Open a report in MicroStrategy Desktop or Web.

2. Right-click the metric to rank, point to Insert (Desktop) or Insert Metric (Web), point to Rank, and then select the break by value from the list. You have the following options:
   - **Break by None**: No break by is used and the rank is calculated accounting for each separate item on the report.
   - **Break by Object**: A break by on Object is used to calculate the rank. For example, a report contains Region, Employee, and the Revenue metric. You select to Break by Region. Each employee is ranked within their region; that is, each region has a separate ranking.

3. If you are creating a rank shortcut metric in MicroStrategy Web, you must choose from the following sort orders:
   - **Ascending**: Values are ranked from least to greatest in ascending order. Therefore, the smallest value has a rank of 1 and the largest value has a rank equal to the total number of items being ranked within a given break-by grouping.
   - **Descending**: Values are ranked from greatest to least in descending order. Therefore, the largest value has a rank of 1 and the smallest value has a rank equal to the total number of items being ranked within a given break-by grouping.

   In Desktop, by default, a rank shortcut metric ranks values from low to high in ascending order. You can change the metric to rank in descending order by editing the metric, as described in To change the ranking order of a rank shortcut metric, page 179.

The report is updated showing your new shortcut metric that displays the ranking for the components selected. To edit a shortcut metric, see Editing derived metrics, page 180. To format the values or headers of a derived metric, see Formatting derived metrics, page 181.

To change the ranking order of a rank shortcut metric

When creating rank shortcut metrics in Desktop, a rank shortcut metric ranks values from low to high in ascending order. You can change the metric to rank in descending order by editing the metric, as described in the steps below.
Prerequisites

- You need the Create Derived Metrics (Desktop) and/or the Web Create Derived Metrics (Web) privileges. These privileges are part of OLAP Services.

To change the ranking order of a rank shortcut metric

1. In MicroStrategy Desktop, open a report that contains a rank shortcut metric.

2. Right-click the rank shortcut metric and select Edit. The Input Metric Formula dialog box opens and displays the metric formula.

3. Modify the ranking order by modifying the $ASC=ranking\_order$ parameter, as described below:
   - $ASC=True$: (Default) Values are ranked from least to greatest in ascending order. Therefore, the smallest value has a rank of 1 and the largest value has a rank equal to the total number of items being ranked within a given break-by grouping.
   - $ASC=False$: Values are ranked from greatest to least in descending order. Therefore, the largest value has a rank of 1 and the smallest value has a rank equal to the total number of items being ranked within a given break-by grouping.

4. Click Validate to confirm that the metric definition is valid for use by the MicroStrategy engine.

5. Click OK to return to the Report Editor or Report Viewer. Any rank ordering changes are displayed in the report.

Editing derived metrics

To change the definition of a derived metric after the metric is applied to a report, right-click the derived metric and select Edit. The Input Metric Formula dialog box is displayed. From this dialog box, you can modify the metric’s definition as needed. For details on any option, click Help. Click OK after you finish, and the modified metric definition is calculated and displayed on the report instantly without having to re-execute the report against the data warehouse.
For an example using the Input Metric Formula dialog box, see *Creating a derived metric using the Input Metric Formula dialog box, page 169.*

**Formatting derived metrics**

The same formatting options for regular metrics are available for derived metrics.

You can format the metric headers and values for a derived metric included on the report grid. To format a derived metric, right-click the derived metric on the grid, point to **Formatting**, and select the derived metrics headers or values. The Format Cells dialog box opens to modify the derived metric formatting. For details on each formatting option, click **Help**. For an example of creating and formatting a derived metric, see *Example: Average profit per customer with transactions, page 171.*

You can also use thresholds with derived metrics in the same ways you can use thresholds with standard metrics created with the Metric Editor. Thresholds are used to create conditional formatting for metric values. To open the Threshold dialog box for a derived metric, right-click the derived metric on the report grid and select **Thresholds**. For more information on thresholds, see the *MicroStrategy Basic Reporting Guide*.

**Deleting derived metrics**

Once a derived metric is created, it becomes part of the report and is saved with the other objects in the report. If you do not want the derived metric to be on the grid but want to keep it in the report definition, you can right-click it in the grid and select **Remove from Grid**. This action is shown in the
image below. The metric is taken off the grid and is no longer bold in the Report Objects pane.

If you do not want the derived metric to be saved as part of the report, you can delete it from the report completely by right-clicking the derived metric and selecting Remove from Report.

If the derived metric is included in the definition of another derived metric on the report, an error message is displayed. To delete the derived metric from the report, you must remove or modify any derived metrics that are dependent on the derived metric you are attempting to delete.

View filter effects on derived metrics

Derived metrics calculate their values using the results in a report. Since view filters restrict the report results, view filters can have an effect on derived metrics. For information on view filter effects on derived metrics and other reporting features, see View filter effects on reporting features, page 219.
Derived element effects on derived metrics

Most derived metrics are represented correctly by derived elements. However, a derived metric that uses a division such as \( \text{Profit/Revenue} \) is initially calculated incorrectly when using derived elements. This is because, by default, derived metrics are evaluated before derived elements. You must change the evaluation order to evaluate derived elements before this type of derived metric. For information on derived element effects on derived metrics and how to evaluate them correctly, see *Derived metrics, page 135*. 
Introduction

A view filter restricts the amount of data displayed on the report, providing you with a different view of the data. This new view can provide a new business perspective for analysis, without having to re-execute the report’s SQL against the data warehouse.
For example, you have a report with Region, Category, and Profit displayed, as shown in the report below.

<table>
<thead>
<tr>
<th>Region</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>Books</td>
<td>$55,495</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$935,123</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$16,347</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$78,357</td>
<td></td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>Books</td>
<td>$46,461</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,933,032</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$14,351</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$13,737</td>
<td></td>
</tr>
<tr>
<td>Northeast</td>
<td>Books</td>
<td>$455,964</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,500,514</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$27,967</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$25,895</td>
<td></td>
</tr>
<tr>
<td>Northwest</td>
<td>Books</td>
<td>$19,734</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$1,604,110</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$5,639</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$5,508</td>
<td></td>
</tr>
<tr>
<td>South</td>
<td>Books</td>
<td>$216,663</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$739,064</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$12,905</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,882</td>
<td></td>
</tr>
<tr>
<td>Southeast</td>
<td>Books</td>
<td>$25,609</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$419,847</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$37,704</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$6,965</td>
<td></td>
</tr>
<tr>
<td>Southwest</td>
<td>Books</td>
<td>$41,576</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$686,421</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$60,787</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$11,294</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Books</td>
<td>$29,195</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$469,635</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$42,611</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$7,952</td>
<td></td>
</tr>
</tbody>
</table>

After a view filter is applied, the resulting report below includes the following view filter qualifications:

- Region In list {Northwest, Southwest}: This qualification restricts the report results to display data only for the Northwest and Southwest regions.

- Profit Greater than 15000: This qualification restricts the report results to display data only for product categories in the Northwest or Southwest regions that had greater than $15,000 in profits.
The view filter’s definition is displayed above the report, as shown below.

The following table lists scenarios where you can use view filters to best support your business model and enhance the analysis of your reports.

<table>
<thead>
<tr>
<th>Analysis Capability</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify the data displayed without re-executing SQL against the data warehouse.</td>
<td>Adding, deleting, or modifying view filters are all executed against a report in memory.</td>
</tr>
<tr>
<td>Allow multiple users to create separate views of data on a single report in memory.</td>
<td>Multiple users can define individual view filters to further restrict the data of a report connected to a shared Intelligent Cube.</td>
</tr>
<tr>
<td>Filter on attributes included in the report.</td>
<td>With the attribute Year on a report, you can use a view filter to determine which years of data to display on the report.</td>
</tr>
<tr>
<td>Perform attribute-to-attribute comparisons.</td>
<td>With the attributes Customer City and Store City on a report, you can specify that Customer City be the same as the Store City. This can give a view of how a store is performing with local customers.</td>
</tr>
<tr>
<td>Filter on metrics included in the report. The output level for the filter can be applied at the report level or the level of the attributes displayed on the report.</td>
<td>With the metric Profit on a report, you can filter on Profit greater than or equal to $1,000,000.</td>
</tr>
<tr>
<td>Perform metric-to-metric comparisons.</td>
<td>With Revenue and Operating Cost metrics on a report, you can specify that Revenue be greater than or equal to Operating Cost.</td>
</tr>
<tr>
<td>Filter on attributes or metrics that are not displayed on the report.</td>
<td>You can drag-and-drop the Profit metric from the report grid to the Report Objects pane. This removes the Profit metric from the display, but any view filters based on that object are still calculated.</td>
</tr>
</tbody>
</table>
This chapter discusses the following topics related to view filters:

- Comparing view filters to report filters and report limits, page 188
- Creating a view filter, page 194
- Deleting a view filter, page 219
- View filter effects on reporting features, page 219

Comparing view filters to report filters and report limits

Filtering reports restricts the data displayed to highlight the data that is of greatest interest or is most applicable to the area of analysis. In addition to view filters, you can also use report filters and report limits to filter data on reports. To decide when to use a view filter, report filter, or report limit, you must understand the advantages and implications of each feature. For a detailed comparison between view filters and other standard MicroStrategy features, see the following sections:

- View filters versus report filters, page 188
- View filters versus report limits, page 192

View filters versus report filters

While they both share some filtering capabilities, view filters and report filters also offer their own unique filtering features that fit different filtering requirements.

The main difference between report filters and view filters is how they are evaluated by the system.

Report filters are a standard MicroStrategy reporting feature that enable you to filter the data retrieved from the data source. Since report filters are evaluated by querying the data source, report filters can perform various types of advanced qualifications, use prompts in qualifications, filter on objects not included in the report, and so on. For more information on report filters in general, refer to the Basic Reporting Guide and Advanced Reporting Guide.
View filters are an OLAP Services feature that enable you to filter the data available on a report after its data has been retrieved from a data source. View filters are evaluated without having to query the data source. While this enables view filters to be evaluated without the overhead of querying the data source, it also means that view filters only have access to the data available on the report. Due to this limited access to data, view filters cannot perform all of the advanced qualifications possible with report filters.

The table below compares the available features and feature requirements of view filters and report filters:

<table>
<thead>
<tr>
<th>Features and Feature Requirements</th>
<th>Available in View Filters</th>
<th>Available in Report Filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute element and attribute form qualifications</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Simple metric qualifications</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Relationship qualifications</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Joint element list qualifications</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Custom expression qualifications</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Can define filtering at report run time by including a prompt in a qualification</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Can filter on objects not included in the Report Objects pane</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>(See Example: Report filter on an attribute not in the report, page 191)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Can filter on objects included in the Report Objects pane, but not included in the report (See Example: Report filter on an attribute not in the report, page 191)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Evaluated without re-executing SQL and querying the data source</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Requires OLAP Services license</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Can be saved as a stand alone object and used in multiple reports</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Can quickly switch the level at which the qualification is evaluated from report level to the level of attributes displayed on the report</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Can be modified while viewing the report data</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
Design considerations

The decision to use a view filter or a report filter to answer your business questions relies on two key factors, functionality and system management.

View filters and report filters both provide a rich set of filtering functionalities, which can be used to answer your business questions. However, since report filters are executed against the data warehouse, more advanced filtering is supported. You may need to use a report filter to implement some of your more advanced business questions. For example, you can define a report filter at report run time by including a prompt in the filter definition. For a list of features supported by report filters that are not supported for view filters, see View filters versus report filters, page 188.

From a system management perspective, report filters and view filters provide two alternatives that affect memory and data warehouse usage.

Report filters help to reduce the memory size of reports by returning less data from the data warehouse. These results can be stored in a cache that decreases the time it takes to access and run the report. The drawback to this approach is that any modifications to the report filter cause the system to access the data source again to create a new report definition, which must be stored in the cache in place of the old definition. The cache still provides quick access to the new report, but this process causes an extra load on the system.

View filters can help reduce memory used by reports, by utilizing Intelligent Cubes. Intelligent Cubes are sets of data that can be shared as a single in-memory copy, among many different reports created by multiple users. Filtering on reports connected to Intelligent Cubes is only achievable with view filters. View filters provide much of the same filtering functionality as report filters, while allowing multiple users to perform analysis on a single Intelligent Cube.
Example: Report filter on an attribute not in the report

For this example, run the Employee Headcount by Region report in the MicroStrategy Tutorial project; the report is shown below. This report does not have a report filter.

Add a report filter (Country = U.S.) that qualifies on the attribute Country, which is not part of the report's definition. The result is the report shown below, which contains data for the regions in the U.S. only, as defined in the report filter.

Although the report filter is based on an attribute not in the report itself, the report data is still affected because of the relationships among the objects in the report, which are all part of the same attribute hierarchy.

In contrast, a view filter can be created only on objects that are part of the report's definition. (The objects in a report's definition are displayed in the Report Objects pane.) This is because view filters only have access to data in the Report Objects pane of the report, rather than the entire data warehouse.

To provide the same type of analysis used in the first report that uses a report filter, include Country in the Report Objects pane of the report, but remove it
from the grid. With Country in the Report Objects pane, you can create a view filter to restrict data to Regions in the US, as shown below.

### View filters versus report limits

View filters and report limits are similar filtering features, as they can both be used to restrict the final result set by setting criteria on the report metrics.

You can define report limits on the value, rank, or percent of the metric value returned. For example, if you set a report limit of `Employee Headcount` greater than (>) 5 in the `Employee Headcount by Region` report in the MicroStrategy Tutorial project, the report returns the following results.

<table>
<thead>
<tr>
<th>Region</th>
<th>Metrics</th>
<th>Employee Headcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mid-Atlantic</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Northeast</td>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

An important aspect of report limits is that they are processed by the SQL Engine after metrics are aggregated. In the report SQL, the report limit definition is included in the `Having` clause, instead of the `Where` clause as
for the report filter. You can observe this in the report’s SQL statement, shown in the image below.

![SQL Statement]

In addition to this functionality, the following features supported for report limits are not supported for view filters:

- Prompts within report limit
- Break By

The table below compares the available features and feature requirements of view filters and report limits:

<table>
<thead>
<tr>
<th>Features and Feature Requirements</th>
<th>Available in View Filters</th>
<th>Available in Report Filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute qualifications</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Simple metric qualifications</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Can define filtering at report run time by including a prompt in a qualification</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Can define Break By filtering parameters</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Evaluated without re-executing SQL and querying the data source</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Requires OLAP Services license</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Can quickly switch the level at which the qualification is evaluated from report level to the level of attributes displayed on the report</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

For general information on report limits, see the Basic Reporting Guide.
**Design considerations**

A view filter is similar to a report limit in that it can also be applied at the report level. However, the report limit and the view filter are not interchangeable. A report limit restricts the size of the report data set that is returned from the data warehouse. In contrast, the view filter is applied to the report dataset without altering its size, allowing you to view a subset of that information. A view filter retrieves information quickly because Intelligence Server dynamically accesses the data already in the report results.

Report designers must consider how to balance the memory usage and the processing power between the data warehouse and Intelligence Server. A report limit is more efficient in terms of report data size because it does not return unnecessary information from the data warehouse. Therefore, the report limit can be used to save space on the Intelligence Server memory. However, if a report limit is too restrictive, you may need to frequently redefine the data definition to yield the information users want to see.

On the other hand, a view filter is more flexible, allowing you to refine your analysis after the report is executed. A view filter gives you more control over the subset of data retrieved from the database you want to see. The view filter may be more useful for analysts because it allows analysts to conduct further investigation and refinement of the report results after the report is executed against the data warehouse.

**Creating a view filter**

You can create a view filter by defining a condition that is based on any attribute or metric on your report. While attribute and metric qualifications both restrict the data displayed on the report, they are created and defined in slightly different ways.

You can create a view filter in any report, as long as you have an OLAP Services license. This includes reports connected to an Intelligent Cube, reports returning information directly from a data warehouse, Freeform SQL reports returning information from an Excel spreadsheet, and so on.

The following sections provide examples of using view filters on reports returning information directly from a data warehouse.
Refer to the following sections to create a view filter that contains attribute and/or metric qualifications:

- **Filtering data based on business attributes, page 195**: Using view filters based on business attributes, you can view a subset of report data that focuses on the business data you are interested in.

  When creating an attribute qualification for a view filter, you can either qualify on a list of attribute elements, or you can qualify on attribute forms:

  - Qualifying on a list of attribute elements is achievable by using the In list or Not in list operators and selecting from a list of attribute elements. For basic steps to filter on a list of attribute elements, see the procedure *To create a view filter with an attribute qualification, page 197*.

  - Qualifying on attribute forms is achievable by using the Where operator and selecting from available attribute forms. For information on qualifying on attribute forms, see *Filtering based on attribute form qualifications, page 198*.

- **Filtering data based on metrics, page 200**: Using view filters based on metrics, you can view a subset of report data that focuses on the data values and ranges you are interested in.

- **Combining view filter qualifications with operators, page 218**: When a view filter has multiple qualifications at the same output level, they are always joined by operators. When qualifications are joined, operators govern the interaction between the different filtering conditions.

## Filtering data based on business attributes

Using a view filter based on business attributes, you can view a subset of report data that focuses on the business data you are interested in. Narrowing the focus of a report to the business data that is of interest of you enables another level of report analysis that can highlight business trends and figures.

For example, the **TOP 5 Materials by Net Sales Amount and Quarter** report from the Sales and Distribution Analysis Module is used for this
example. The year 2007 has been selected to answer the prompt on year. The resulting report is shown below.

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Material</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>Net Sales Order Amount</strong></td>
</tr>
<tr>
<td>2007 Q1</td>
<td>UC PCEconomy 200</td>
<td>$4,250,000</td>
</tr>
<tr>
<td></td>
<td>UC PCValue 200</td>
<td>$4,150,000</td>
</tr>
<tr>
<td></td>
<td>UC PCEconomy 100</td>
<td>$2,150,000</td>
</tr>
<tr>
<td></td>
<td>UC PCTop 200</td>
<td>$1,870,000</td>
</tr>
<tr>
<td></td>
<td>UC PCValue 100</td>
<td>$1,725,000</td>
</tr>
<tr>
<td>2007 Q2</td>
<td>UC PCEconomy 100</td>
<td>$4,100,000</td>
</tr>
<tr>
<td></td>
<td>UC PCValue 200</td>
<td>$2,740,000</td>
</tr>
<tr>
<td></td>
<td>UC PCValue 100</td>
<td>$2,413,500</td>
</tr>
<tr>
<td></td>
<td>UC PCTop 200</td>
<td>$2,200,000</td>
</tr>
<tr>
<td></td>
<td>UC NBProfessional 100</td>
<td>$1,703,000</td>
</tr>
<tr>
<td>2007 Q3</td>
<td>UC PCValue 100</td>
<td>$5,697,500</td>
</tr>
<tr>
<td></td>
<td>UC PCValue 200</td>
<td>$4,165,000</td>
</tr>
<tr>
<td></td>
<td>UC LaserPrinter B/W - E10</td>
<td>$2,743,500</td>
</tr>
<tr>
<td></td>
<td>UC PCEconomy 100</td>
<td>$2,510,000</td>
</tr>
<tr>
<td></td>
<td>UC PCTop 200</td>
<td>$1,570,000</td>
</tr>
<tr>
<td>2007 Q4</td>
<td>UC PCValue 100</td>
<td>$2,175,000</td>
</tr>
<tr>
<td></td>
<td>UC PCTop 200</td>
<td>$1,632,000</td>
</tr>
<tr>
<td></td>
<td>UC PCEconomy 200</td>
<td>$1,600,000</td>
</tr>
<tr>
<td></td>
<td>UC NBProfessional 100</td>
<td>$1,393,000</td>
</tr>
<tr>
<td></td>
<td>UC PCTop 100</td>
<td>$1,216,000</td>
</tr>
</tbody>
</table>

The report identifies, for each quarter, the top five materials by net sales order amount.

Once you review the top 5 materials for each quarter in 2007, you decide to focus on the UC PCEconomy 100 and UC PCEconomy 200 by creating a view
Creating a view filter

A view filter that includes these two materials. The view filter along with the resulting report is shown below.

```
Material In list: UC PCEconomy 100, UC PCEconomy 200

Click here to start a new qualification
```

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Material</th>
<th>Metrics</th>
<th>Net Sales Order Amount</th>
<th>Sales Orders Items</th>
<th>Sales Orders Quantity (Base Units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007 Q1</td>
<td>UC PCEconomy 200</td>
<td></td>
<td>$4,250,000</td>
<td>4</td>
<td>5,350</td>
</tr>
<tr>
<td></td>
<td>UC PCEconomy 100</td>
<td></td>
<td>$2,150,000</td>
<td>3</td>
<td>3,000</td>
</tr>
<tr>
<td>2007 Q2</td>
<td>UC PCEconomy 100</td>
<td></td>
<td>$4,100,000</td>
<td>3</td>
<td>6,750</td>
</tr>
<tr>
<td>2007 Q3</td>
<td>UC PCEconomy 100</td>
<td></td>
<td>$2,510,000</td>
<td>3</td>
<td>5,000</td>
</tr>
<tr>
<td>2007 Q4</td>
<td>UC PCEconomy 200</td>
<td></td>
<td>$1,600,000</td>
<td>1</td>
<td>2,000</td>
</tr>
</tbody>
</table>

The report shown above focuses on the UC PCEconomy 100 and UC PCEconomy 200 materials to show when these two materials had net sales order amounts in the top 5 for a given quarter.

The steps below show you how to create a view filter with an attribute qualification, as well as how to create the example scenario above.

**Prerequisites**

- You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.

**To create a view filter with an attribute qualification**

1. Log in to a project in MicroStrategy Desktop. For steps to create a view filter in MicroStrategy Web, see the MicroStrategy Web Help.

   For the example scenario, log in to the Sales and Distribution Analysis Module project.

2. Run the report in Grid View.

   For the example scenario, browse to and run the **TOP 5 Materials by Net Sales Amount and Quarter** report. Select the 2007 attribute element to answer the prompt.
3 If the View Filter area is not displayed, from the View menu, select View Filter.

4 In the View Filter area, click Click here to start a new qualification.

5 Click Field, and then select an attribute.

For the example scenario, select Material.

6 Click Operator, and then select one of the following operators to create a filter qualification (for the example scenario, select In list):

- **In list**: Restricts attribute data to the list of attribute elements you select. Click Value, and then select the attribute elements to restrict attribute data to.

  For the example scenario, select UC PCEconomy 100 and UC PCEconomy 200.

- **Not in List**: Restricts attribute data to the list of attribute elements that are not in the list of attribute elements you select. Click Value, and select the attribute elements to exclude data for.

- **Where**: Restricts attribute data based on a filter qualification of an attribute form. For information on how to define this type of filter qualification for a view filter, see Filtering based on attribute form qualifications, page 198.

7 Click OK to complete the view filter qualification.

8 If the Auto-Apply Changes check box is cleared, click Apply to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter. If you define multiple view filter qualifications at the same output level, you can modify the logical operator used to join the qualifications, as described in Combining view filter qualifications with operators, page 218. For information on the output level of view filter qualifications, see Evaluating qualifications at the report or grid level, page 213.

**Filtering based on attribute form qualifications**

You can qualify on attribute forms using the Where operator. Once you select an attribute form, you can use various logical and mathematical operators to create a view filter qualification based on an attribute form, to restrict data.
An attribute form qualification using a Contains operator that restricts data to materials that contain Economy in their DESC form is shown below.

If you use this view filter with the TOP 5 Materials by Net Sales Amount and Quarter report, the resulting report is the same as the example scenario in *Filtering data based on business attributes*, page 195, which uses the attribute element qualification Material In list {UC PCEconomy 100, UC PCEconomy 200}.

You can use any of the following operators in attribute form qualifications. These operators are described in detail in *Appendix A, Logical and Mathematical Operators for Filtering* in the *MicroStrategy Advanced Reporting Guide*:

<table>
<thead>
<tr>
<th>Operator Type</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison Operators</td>
<td></td>
</tr>
<tr>
<td>Comparison operators</td>
<td></td>
</tr>
<tr>
<td>values. The values can</td>
<td>Exactly</td>
</tr>
<tr>
<td>be numbers, text</td>
<td>Different from</td>
</tr>
<tr>
<td>strings, or expressions.</td>
<td>Greater than</td>
</tr>
<tr>
<td></td>
<td>Less than</td>
</tr>
<tr>
<td></td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td></td>
<td>Less than or equal to</td>
</tr>
<tr>
<td></td>
<td>Between</td>
</tr>
<tr>
<td></td>
<td>Is Null</td>
</tr>
<tr>
<td></td>
<td>Is Not Null</td>
</tr>
</tbody>
</table>

Click here to start a new qualification
Filtering data based on metrics

Using view filters based on metrics, you can view a subset of report data that focuses on the data values and ranges you are interested in. Narrowing the focus of a report to the data that is of interest of you enables another level of report analysis that can highlight business trends and figures.

For example, review the **16 View Filter - Metric Qualification** report from the Tutorial project, shown below.

![View Filter - Metric Qualification](image)

This report returns revenue, cost, and profit data for employees, while also displaying the region the employee is in. The view filter restricts the report results to only return data for those employees who generated less than $500,000 in revenue. This reduces the large number of employee results to a smaller set of employees that are generating a relatively low amount of revenue.

When creating metric qualifications in a view filter, you can use various logical and mathematical operators. You can use any of the following operators in metric qualifications. These are described in detail in Appendix.

<table>
<thead>
<tr>
<th>Operator Type</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern Operators</td>
<td>Like</td>
</tr>
<tr>
<td></td>
<td>Not Like</td>
</tr>
<tr>
<td></td>
<td>Contains</td>
</tr>
<tr>
<td></td>
<td>Does not contain</td>
</tr>
<tr>
<td></td>
<td>Begins with</td>
</tr>
<tr>
<td></td>
<td>Does not begin with</td>
</tr>
<tr>
<td></td>
<td>Ends with</td>
</tr>
<tr>
<td></td>
<td>Does not end with</td>
</tr>
</tbody>
</table>
A, Logical and Mathematical Operators for Filtering in the MicroStrategy Advanced Reporting Guide:

- Exactly
- Different from
- Greater than
- Less than
- Greater than or equal to
- Less than or equal to
- Between
- Not Between
- Is Null
- Is Not Null

Once you select an operator, you can either type in a value or select a metric to return the value to qualify on.

You can also qualify on the rank or percentage of a metric value for a given report. For example, you can restrict the report shown above to display all data for employees in the bottom 20% of revenue. For information on using view filters to restrict report results based on ranks or percentages of metric data, see Filtering metrics on rank and percentage ranges, page 205.

The steps below show you how to create a view filter with a metric qualification, as well as how the example scenario above was created.

**Prerequisites**

- You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.

---

**To create a view filter with a metric qualification**

1. Log in to a project in MicroStrategy Desktop. For steps to create a view filter in MicroStrategy Web, see the MicroStrategy Web Help.

   For the example scenario, log in to the MicroStrategy Tutorial project.
2 Run the report in Grid View.
   For the example scenario, browse to and run the 16 View Filter - Metric Qualification report.

3 If the View Filter area is not displayed, from the View menu, select View Filter.

4 In the View Filter area, click Click here to start a new qualification.
   You can also create a new view filter qualification on a metric by right-clicking a metric, pointing to Filter On, and selecting Add Condition.

5 Click Field, and then select a metric.
   For the example scenario, select Revenue.

6 Click Operator, and then select an operator.
   For the example scenario, select Less than.

7 Click Value, and then select Type a value, or select a metric to return the value to qualify on. Selecting a metric allows you to create a metric-to-metric qualification, which are described in Filtering based on metric-to-metric comparisons, page 202.
   For the example scenario, select Type a value, and then type 500000.

8 If the Auto-Apply Changes check box is cleared, click Apply to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter. If you define multiple view filter qualifications at the same output level, you can modify the logical operator used to join the qualifications, as described in Combining view filter qualifications with operators, page 218. For information on the output level of view filter qualifications, see Evaluating qualifications at the report or grid level, page 213.

Filtering based on metric-to-metric comparisons

You can compare values of two metrics to filter the results of a report by creating metric-to-metric qualifications. This type of metric qualification can provide analysis such as comparing metric values over time.
For example, you can create a report that restricts the data to profit margins greater than last year's profit margins. The report shown below includes Year, Region, Profit Margin, and Profit Margin (Last Year), and was created in the MicroStrategy Tutorial project.

<table>
<thead>
<tr>
<th>Year</th>
<th>Region</th>
<th>Metrics</th>
<th>Profit Margin</th>
<th>Profit Margin (Last Year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006</td>
<td>Central</td>
<td></td>
<td>15.81%</td>
<td>15.63%</td>
</tr>
<tr>
<td></td>
<td>Mid-Atlantic</td>
<td></td>
<td>17.29%</td>
<td>17.32%</td>
</tr>
<tr>
<td></td>
<td>Northeast</td>
<td></td>
<td>17.85%</td>
<td>17.40%</td>
</tr>
<tr>
<td></td>
<td>Northwest</td>
<td></td>
<td>17.35%</td>
<td>17.31%</td>
</tr>
<tr>
<td></td>
<td>South</td>
<td></td>
<td>18.07%</td>
<td>17.62%</td>
</tr>
<tr>
<td></td>
<td>Southeast</td>
<td></td>
<td>15.65%</td>
<td>15.44%</td>
</tr>
<tr>
<td></td>
<td>Southwest</td>
<td></td>
<td>15.36%</td>
<td>15.45%</td>
</tr>
<tr>
<td></td>
<td>Web</td>
<td></td>
<td>15.41%</td>
<td>15.08%</td>
</tr>
<tr>
<td>2007</td>
<td>Central</td>
<td></td>
<td>15.66%</td>
<td>15.81%</td>
</tr>
<tr>
<td></td>
<td>Mid-Atlantic</td>
<td></td>
<td>17.27%</td>
<td>17.29%</td>
</tr>
<tr>
<td></td>
<td>Northeast</td>
<td></td>
<td>17.69%</td>
<td>17.85%</td>
</tr>
<tr>
<td></td>
<td>Northwest</td>
<td></td>
<td>17.42%</td>
<td>17.35%</td>
</tr>
<tr>
<td></td>
<td>South</td>
<td></td>
<td>16.91%</td>
<td>18.07%</td>
</tr>
<tr>
<td></td>
<td>Southeast</td>
<td></td>
<td>15.69%</td>
<td>15.65%</td>
</tr>
<tr>
<td></td>
<td>Southwest</td>
<td></td>
<td>15.51%</td>
<td>15.36%</td>
</tr>
<tr>
<td></td>
<td>Web</td>
<td></td>
<td>15.40%</td>
<td>15.41%</td>
</tr>
</tbody>
</table>

You can create a view filter to then restrict the data on the report to profit margins greater than last year's profit margins. The metric-to-metric qualification and resulting report are shown below.

You can now review when regions had increases in profit margins. For example, from the report above you can determine that the Northwest and
Southeast regions have shown increases in profit margins from 2006 to 2007.

You can take advantage of view filters’ ability to update the report results without having to re-execute SQL against the data warehouse to perform further quick analysis. For example, you can switch the operator from Greater Than to Less Than to quickly switch to a view of data for profit margins that are less than the previous year’s profit margins. The metric-to-metric qualification in the view filter and resulting report are shown below.

The steps below show you how to create a view filter with a metric-to-metric qualification, as well as how the example scenario above was created.

**Prerequisites**

- You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.

---

**To create a view filter with a metric-to-metric qualification**

1. Log in to a project in MicroStrategy Desktop. For steps to create a view filter in MicroStrategy Web, see the MicroStrategy Web Help.
   
   For the example scenario, log in to the MicroStrategy Tutorial project.

2. Run the report in Grid View.
   
   For the example scenario, create a report with Year, Region, and Profit Margin on the report. Create a metric that returns a transformation of
last year’s profit margin values, and name the metric Profit Margin (Last Year). For details to create a transformation metric, see the Advanced Reporting Guide.

3 If the View Filter area is not displayed, from the View menu, select View Filter.

4 In the View Filter area, click Click here to start a new qualification.
   You can also create a new view filter qualification on a metric by right-clicking a metric, pointing to Filter On, and selecting Add Condition.

5 Click Field, and then select a metric.
   For the example scenario, select Profit Margin.

6 Click Operator, and then select an operator.
   For the example scenario, select Greater than.

7 Click Value, and then select a metric to return the value to qualify on.
   For the example scenario, select Profit Margin (Last Year).

8 If the Auto-Apply Changes check box is cleared, click Apply to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter. If you define multiple view filter qualifications at the same output level, you can modify the logical operator used to join the qualifications, as described in Combining view filter qualifications with operators, page 218. For information on the output level of view filter qualifications, see Evaluating qualifications at the report or grid level, page 213.

Filtering metrics on rank and percentage ranges

In addition to restricting report results based on metric qualifications, you can also restrict report results based on the rank or percentage range of a metric value for a given report. This enables you to display reports with data such as products in the top 40% of profit. You could also create a report to return the top 10 employees in terms of average performance score.

It is important to understand the operators that are available for these types of view filters. When creating view filter qualifications on the rank or percentage ranges of metric values, you can use the following mathematical
and logical operators, which are described as they relate to rank and percent metric qualifications in the table below:

In Desktop, all of the operators listed below can be used to create rank and percent metric qualifications as part of a view filter. Rank and percent metric qualifications using any of these operators can be viewed in MicroStrategy Web. However, only the Is Highest (referred to as Highest or Highest% in Web) and Is Lowest (referred to as Lowest or Lowest% in Web) can be used to create or modify rank and percent metric qualifications in Web.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Functionality and considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exactly</td>
<td>Identifies a specific value, as described below:</td>
</tr>
<tr>
<td></td>
<td>• For rank ranges, this restricts a report to data for only one specific rank. For example, a rank range defined as Exactly 1 returns data for only the highest value of a given metric. This scenario produces the same results as using the Is Highest operator described below in this table.</td>
</tr>
<tr>
<td></td>
<td>• For percent ranges, this operator has limited value. This is because it restricts a report to data that is exactly a specific percentage. Percentages are not always integer values and therefore it is difficult to predict what percentage would return meaningful results.</td>
</tr>
<tr>
<td>Different from</td>
<td>Identifies values that are other than the specific value indicated:</td>
</tr>
<tr>
<td></td>
<td>• For rank ranges, this restricts a report to data for only ranks that are different from the given value. For example, a rank range defined as different from 1 returns data for all ranks except the highest value of a given metric.</td>
</tr>
<tr>
<td></td>
<td>• For percent ranges, this operator has limited value. This is because it restricts a report to data that is different from a specific percentage. Percentages are not always integer values and therefore it is difficult to predict what percentage would return meaningful results. In most cases, using this operator with percent metric qualifications does not restrict any data.</td>
</tr>
<tr>
<td>Between</td>
<td>Identifies values in a range that has both a lower and an upper limit:</td>
</tr>
<tr>
<td></td>
<td>• For rank ranges, you can provide a lower and upper rank limit. For example, you can display only ranks 20 through 40.</td>
</tr>
<tr>
<td></td>
<td>• For percent ranges, you can provide a lower and upper percentage limit. For example, you can display only data with metric values between 20% and 40% of all values.</td>
</tr>
<tr>
<td>Not Between</td>
<td>Identifies values in a range that has both a lower and an upper limit. Data is restricted for any values outside of this range:</td>
</tr>
<tr>
<td></td>
<td>• For rank ranges, you can provide a lower and upper rank limit. For example, you can display only ranks not between 20 through 40. This displays ranks 1 through 19 and 41 and down.</td>
</tr>
<tr>
<td></td>
<td>• For percent ranges, you can provide a lower and upper percentage limit. For example, you can display only data with metric values not between 20% and 40% of all values. This displays data with metric values between 1% and 19% and between 41% and 100%.</td>
</tr>
</tbody>
</table>
The rank and percentage ranges view filters are discussed in the sections listed below:

- *Creating a view filter on a rank range of metric values, page 208*: You can create a view filter that restricts report results based on a rank range.
of metric values for a given report. This can allow you to view analysis such as the bottom 20 products in terms of profit margin.

- **Creating a view filter on a percentage range of metric values, page 211:** You can create a view filter that restricts report results based on a percent range of metric values for a given report. This can allow you to view analysis such as employees between 30% and 60% of tenure length with the company.

### Creating a view filter on a rank range of metric values

You can create a view filter that restricts report results based on a rank range of metric values for a given report. This can allow you to view analysis such as the bottom 20 products in terms of profit margin. Further analysis can be explored on this rank range of products to determine how to increase profit margins for these products, or decide which products should be discontinued.

This type of report and analysis can be created in the MicroStrategy Tutorial project. You first create a report with Item, Revenue, Profit, and Profit Margin on the report, as shown below.

<table>
<thead>
<tr>
<th>Item</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 Places to Go While Still Young at Heart</td>
<td>$135,930</td>
<td>$35,859</td>
<td>26.38%</td>
<td></td>
</tr>
<tr>
<td>Art As Experience</td>
<td>$31,739</td>
<td>$7,312</td>
<td>23.04%</td>
<td></td>
</tr>
<tr>
<td>The Painted Word</td>
<td>$17,495</td>
<td>$2,080</td>
<td>11.89%</td>
<td></td>
</tr>
<tr>
<td>Hirschfeld on Line</td>
<td>$103,318</td>
<td>$25,087</td>
<td>24.28%</td>
<td></td>
</tr>
<tr>
<td>Adirondack Style</td>
<td>$77,428</td>
<td>$19,826</td>
<td>25.61%</td>
<td></td>
</tr>
<tr>
<td>Architecture : Form, Space, &amp; Order</td>
<td>$82,099</td>
<td>$20,027</td>
<td>24.39%</td>
<td></td>
</tr>
<tr>
<td>50 Favorite Rooms</td>
<td>$51,105</td>
<td>$12,026</td>
<td>23.53%</td>
<td></td>
</tr>
<tr>
<td>500 Best Vacation Home Plans</td>
<td>$23,239</td>
<td>$4,771</td>
<td>20.53%</td>
<td></td>
</tr>
<tr>
<td>Blue &amp; White Living</td>
<td>$32,014</td>
<td>$7,424</td>
<td>23.19%</td>
<td></td>
</tr>
<tr>
<td>Ways of Seeing</td>
<td>$30,961</td>
<td>$7,134</td>
<td>23.04%</td>
<td></td>
</tr>
<tr>
<td>Gorzo, the Art</td>
<td>$81,743</td>
<td>$16,697</td>
<td>20.43%</td>
<td></td>
</tr>
<tr>
<td>Cabin Fever : Rustic Style Comes Home</td>
<td>$27,230</td>
<td>$5,633</td>
<td>20.68%</td>
<td></td>
</tr>
<tr>
<td>American Bungalow Style</td>
<td>$95,617</td>
<td>$23,296</td>
<td>24.35%</td>
<td></td>
</tr>
</tbody>
</table>

Only a subset of the report results are shown above, but notice that data for 360 items have been returned. To narrow the analysis of the report, you
create a view filter to restrict the report results to the bottom 20 products in terms of profit margin. The view filter and resulting report are shown below.

With this updated report, you can now perform further analysis on each item to determine a strategy to improve your profit margins.

Notice that the view filter above uses the Bottom operator. For a description of each operator available when creating view filters with rank ranges, see Filtering metrics on rank and percentage ranges, page 205.

The steps below show you how to create a view filter with a rank metric qualification, as well as how the example scenario above was created.

**Prerequisites**

- You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.
To create a view filter with a rank metric qualification

1. Log in to a project in MicroStrategy Desktop. For steps to create a view filter in MicroStrategy Web, see the MicroStrategy Web Help.

   For the example scenario, log in to the MicroStrategy Tutorial project.

2. Run a report in Grid View.

   For the example scenario, create a report with Item, Revenue, Profit, and Profit Margin on the report, as shown in Creating a view filter on a rank range of metric values, page 208.

3. If the View Filter area is not displayed, from the View menu, select View Filter.

4. In the View Filter area, click Click here to start a new qualification.

5. To create a rank metric qualification, click Field, point to Rank, and then select a metric.

   For the example scenario, select Profit Margin.

6. Click Operator, and then select an operator.

   For the example scenario, select Bottom.

7. Click Value, and then select Type a value. Type the value for the rank number you want to restrict data to.

   For the example scenario, type 20.

8. If the Auto-Apply Changes check box is cleared, click Apply to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter. If you define multiple view filter qualifications at the same output level, you can modify the logical operator used to join the qualifications, as described in Combining view filter qualifications with operators, page 218. For information on the output level of view filter qualifications, see Evaluating qualifications at the report or grid level, page 213.
**Creating a view filter on a percentage range of metric values**

You can create view filters that restrict report results based on a percent range of metric values for a given report. This can allow you to view analysis such as employees between 30% and 60% of tenure length with the company. You can then explore performance metrics within this percentage range of employees and determine where promotions and other actions are necessary.

For example, you create a report with Year, Region, Category, Revenue, and Profit on the report, with Category not displayed on the report grid, as shown below.
You decide to analyze this report to show only the data that is within the top 10% of profit. The view filter and resulting report are shown below.

With this analysis, you can now perform further analysis to determine why profit was at its highest during these years and within these regions.

Notice that the view filter above uses the Top operator. For a description of each operator available when creating view filters with percent ranges, see Filtering metrics on rank and percentage ranges, page 205.

The steps below show you how to create a view filter with a percent metric qualification, as well as how the example scenario above was created.

Prerequisites

• You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.

To create a view filter with a percent metric qualification

1 Log in to a project in MicroStrategy Desktop. For steps to create a view filter in MicroStrategy Web, see the MicroStrategy Web Help.

For the example scenario, log in to the MicroStrategy Tutorial project.

2 Run the report in Grid View.

For the example scenario, create a report with Year, Region, Category, Revenue, and Profit on the report with Category not displayed on the report grid, as shown in Creating a view filter on a percentage range of metric values, page 211.

3 If the View Filter area is not displayed, from the View menu, select View Filter.

4 In the View Filter area, click Click here to start a new qualification.
5 To create a percent metric qualification, click Field, point to Percent, and then select a metric.

For the example scenario, select Profit.

6 Click Operator, and then select an operator.

For the example scenario, select Top.

7 Click Value, and then select Type a value. Type the value for the rank number you want to restrict data to.

For the example scenario, type 20.

Valid numbers for percent metric qualifications are 1 to 100.

8 If the Auto-Apply Changes check box is cleared, click Apply to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter. Further analysis on this report is performed to demonstrate how you can change the level of evaluation for view filter metric qualifications. This analysis is provided in Evaluating qualifications at the report or grid level, page 213.

If you define multiple view filter qualifications at the same output level, you can modify the logical operator used to join the qualifications, as described in Combining view filter qualifications with operators, page 218. For information on the output level of view filter qualifications, see Evaluating qualifications at the report or grid level, page 213.

**Evaluating qualifications at the report or grid level**

When a metric qualification in a view filter is evaluated by the system, the evaluation can be performed for all data that is returned for the report, or only the view of data that is currently available on the report grid. These two options can produce different report results when using the OLAP Services feature called dynamic aggregation.

Dynamic aggregation enables you to remove attributes from the report grid, but keep them as part of the report definition. The action of moving attributes on or off of the report grid aggregates the metric values at the new level of the report. For information on dynamic aggregation, see Chapter 5, Dynamic Aggregation.
By default, metric qualifications in a view filter are evaluated at the level of data that is available on the report grid. This means that any attributes that are included in the Report Objects pane but not on the report grid are not used to determine the level of the metric qualification.

For example, you create a report with Year, Region, Category, Revenue, and Profit on the report, with Category not displayed on the report grid, as shown below.

This report can be created in the MicroStrategy Tutorial project.

You can use dynamic aggregation to drag and drop the Category attribute from the report grid to the Report Objects pane. This allows Category to affect the report level without being displayed on the grid.
You decide to analyze this report to show only the data that is within the top 10% of profit. The view filter and resulting report are shown below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006</td>
<td>Mid-Atlantic</td>
<td>$7,662,786</td>
<td>$1,324,696</td>
<td></td>
</tr>
<tr>
<td>2007</td>
<td>Mid-Atlantic</td>
<td>$9,384,211</td>
<td>$1,620,821</td>
<td></td>
</tr>
</tbody>
</table>

Steps to create the report shown above is described in *Creating a view filter on a percentage range of metric values, page 211*.

Notice that only two rows of data are returned. The metric qualification has been evaluated at the level of the report grid, which is Year and Region. This gives you a view of data within the top 10% of profit for the data displayed on the report grid.

However, this report also includes the Category attribute in the Report Objects pane. Since this attribute is available on the report, you can also view data within the top 10% of profit at the Category, Region, and Year level. Evaluating the metric qualification at this level returns the report results shown below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Region</th>
<th>Metrics</th>
<th>Revenue</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Mid-Atlantic</td>
<td>$5,971,153</td>
<td>$1,042,487</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northwest</td>
<td>$2,502,737</td>
<td>$435,648</td>
<td></td>
</tr>
<tr>
<td>2006</td>
<td>Mid-Atlantic</td>
<td>$7,460,047</td>
<td>$1,299,891</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northeast</td>
<td>$2,755,465</td>
<td>$487,923</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northwest</td>
<td>$3,106,552</td>
<td>$543,087</td>
<td></td>
</tr>
<tr>
<td>2007</td>
<td>Central</td>
<td>$2,247,755</td>
<td>$393,976</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mid-Atlantic</td>
<td>$9,134,630</td>
<td>$1,590,653</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northeast</td>
<td>$3,576,010</td>
<td>$627,891</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northwest</td>
<td>$3,556,006</td>
<td>$624,374</td>
<td></td>
</tr>
</tbody>
</table>

Notice that there are many more rows of data that are within the top 10% of profit. This is because Category is now included in the calculation of the metric qualification. While this evaluation option for metric qualifications returns a different type of analysis, the same analysis can be achieved by simply adding all attributes from the Report Objects pane onto the report grid, so that all attributes are then present on the report grid.
The following information should be taken into consideration when choosing an evaluation level for a metric qualification. This information assumes you are familiar with report levels as explained in the *Advanced Reporting Guide*:

- **Evaluation at the report grid level**: Evaluating metric qualifications at the level present on the report grid allows the view filter to dynamically display analysis that reflects the data available on the report grid. If all attributes are on the report grid, then this level is used to calculate the metric qualification. Additionally, anytime an attribute is moved between the Report Objects pane and the report grid, the view filter dynamically recalculates the metric qualification to reflect the new level of data on the report grid.

  You can join metric qualifications evaluated at the grid level to any other metric qualifications evaluated at the grid with logical operators, as described in *Combining view filter qualifications with operators*, page 218.

  If derived metrics are also on this report, evaluating metric qualifications at the grid level also causes the metric qualifications to be evaluated after derived metrics by default. This means that these qualifications filter the results of any derived metric calculations. For more information, see *View filter effects on derived metrics*, page 220.

- **Evaluation at the report level**: Evaluating metric qualifications at the report level, regardless of what attributes are on the report grid or the Report Objects pane, provides a consistent level of analysis during dynamic aggregation.

  You can also join metric qualifications evaluated at the report level to attribute qualifications or other metric qualifications evaluated at the report level with logical operators, as described in *Combining view filter qualifications with operators*, page 218.

  If derived metrics are also on this report, evaluating metric qualifications at the report level also causes the metric qualifications to be evaluated before derived metrics by default. This means that these qualifications filter data before any derived metric calculations are applied. For more information, see *View filter effects on derived metrics*, page 220.

- **Metric-to-metric qualifications**: The evaluation level of metric-to-metric qualifications cannot be modified. All metric-to-metric qualifications are evaluated at the report level. For information on metric-to-metric qualifications, see *Filtering based on metric-to-metric comparisons*, page 202.
The steps below show you how to modify the evaluation of metric qualifications in a view filter.

**Prerequisites**

- You need the Use View Filter Editor (Desktop) and/or the Web Use View Filter Editor (Web) privileges. These privileges are part of OLAP Services.
- A report with a metric qualification in the view filter that is not a metric-to-metric qualification.
- To observe how this modification can affect report results, the report should also have some attributes in the Report Objects pane, but not on the report grid.

You cannot modify the evaluation level of metric-to-metric qualifications. All metric-to-metric qualifications are evaluated at the report level.

---

**To modify the evaluation level of metric qualifications in a view filter**

1. Log in to a project in MicroStrategy Desktop. For steps to modify the evaluation level of metric qualifications in a view filter in MicroStrategy Web, see the MicroStrategy Web Help.

2. Run the report in Grid View.

3. If the View Filter area is not displayed, from the View menu, select View Filter.

4. In the View Filter area, right-click a metric qualification and select one of the following options, which you can switch between:

   - **Apply Condition at the Grid Level** (default): Evaluates the metric qualification only for the attributes included on the report grid. Attributes in the Report Objects pane but not on the report grid are not included in the metric qualification evaluation.

   - **Apply Condition at the \{attributes in Report Objects\} Level**: Evaluates the metric qualification for all attributes included in the Report Objects pane, regardless of whether they are displayed on the report grid.

You can choose different evaluation options for separate metric qualifications in the same view filter.
5 If the Auto-Apply Changes check box is cleared, click **Apply** to apply the view filter to the report.

The report is updated. The report data is restricted as defined by the view filter.

**Combining view filter qualifications with operators**

When a view filter has multiple qualifications at the same output level, they are always combined by operators. When qualifications are combined, operators govern the interaction between different filtering conditions. Whenever you have more than one qualification in a view filter at the same output level, you can define the operator as any of the following:

- **AND**
- **OR**
- **OR NOT**
- **AND NOT**

For information and examples on how each of these operators govern interaction between filter qualifications, see the *Basic Reporting Guide*.

You cannot change the logical operator between two metric qualifications if all of the following are true:

- The metric qualifications use two different metrics (for example, Revenue in Qualification 1 and Profit in Qualification 2).
- The metric qualifications are not metric-to-metric qualifications, but instead compare the metrics to numeric values.
- The metric qualifications are evaluated at the grid level.

Because the output level of view filter qualifications determines which qualifications can be joined with logical operators, by default all attribute qualifications are at the same output level as other attribute qualifications, and all metric qualifications are at the same output level as other metric qualifications. This mean that an attribute qualification can be joined with other attribute qualifications, and metric qualifications can be joined with other metric qualifications, but attribute and metric qualifications cannot be joined.
However, you can modify the output level of metric qualifications. If you modify a metric qualification to be evaluated at the report level, the metric qualification can be joined with attribute qualifications and with any other metric qualifications that have been defined to be evaluated at the report level. Metric qualifications evaluated at the report level cannot be joined with metric qualifications evaluated at the grid level. For information on the benefits of evaluating metric qualifications at the report level versus the grid level, see *Evaluating qualifications at the report or grid level, page 213*.

## Deleting a view filter

When deleting a view filter, you have the option of deleting all of the qualifications of a view filter at once, or deleting the qualifications individually. The basic steps to perform both options are given below.

### To delete all view filter qualifications

1. In an opened report with a view filter, in the **View Filter** pane, click **Clear**.
2. If you do not have Auto-Apply changes selected, click **Apply**. If you only perform Clear without Apply, the view filter appears in the report the next time you open it.

### To delete a view filter qualification individually

1. In an opened report with a view filter, right-click the qualification and select **Remove qualification**.
2. If you do not have Auto-Apply changes selected, click **Apply**. If you only perform Clear without Apply, the view filter appears in the report the next time you open it.

## View filter effects on reporting features

Applying a view filter to a report restricts the data that is displayed on the report grid. However, unlike report filters, data restricted from display by a
view filter is still available for a report. This difference has an effect on how view filters interact with various reporting features. This section discusses:

- **View filter effects on derived metrics, page 220**
- **View filter effects on metrics with relative functions, page 226**
- **View filter effects on dynamic aggregation, page 228**
- **View filter effects on derived elements, page 232**

### View filter effects on derived metrics

For most derived metrics, view filters have the same effects as they do on standard metrics created with the Metric Editor. When data is restricted from view, the derived metric values along with any related data is hidden from the view of the report. For example, if a view filter restricts data to the year 2007, then any derived metrics and standard metrics included on the report only display data for the year 2007.

However, derived metrics created with relative functions (OLAP functions and Rank and Ntile functions) are affected by the evaluation of view filters in a unique way.

A derived metric uses the data available on a report to calculate its values. Since view filter qualifications restrict the data displayed on the report, this can also restrict the data available for derived metrics, depending on the evaluation order of the two objects. This restriction of data, combined with derived metrics with relative functions that depend on other values on the report, can effect the display of report results.

The view filter effects on derived metrics with relative functions depends on how view filters are used in such reports, as described in the sections listed below:

- **Using view filters that affect derived metrics with relative functions, page 221**: Derived metrics with relative functions such as RunningSum or Rank return values that are relative to other values on a report. When data is restricted by view filter qualifications on other objects of a report, you can allow the values of derived metrics with relative functions to calculate their relative values based on the new view of report data.

- **Using derived metrics with relative functions in view filters, page 223**: Derived metrics with relative functions such as RunningSum or Rank return values that are relative to the data on a report. When derived
metrics with relative functions are used to define view filters, you can allow the relative values to reflect the new view of data or retain the relative values for all report data.

**Using view filters that affect derived metrics with relative functions**

Derived metrics with relative functions such as RunningSum or Rank return values that are relative to other values on a report. When data is restricted by view filter qualifications on other objects of a report, you can allow the values of derived metrics with relative functions to calculate their relative values based on the new view of report data.

For example, consider a report created in the MicroStrategy Tutorial project with Year, Category, Profit, Profit Margin, and a rank shortcut derived metric named Rank (Profit Margin) as shown below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Profit Margin</th>
<th>Rank (Profit Margin)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Books</td>
<td>$251,345</td>
<td>22.04%</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,768,984</td>
<td>17.43%</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$55,041</td>
<td>6.69%</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$43,243</td>
<td>6.90%</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2006</td>
<td>Books</td>
<td>$294,789</td>
<td>22.32%</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,389,107</td>
<td>17.56%</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$71,686</td>
<td>6.88%</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$53,666</td>
<td>7.17%</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2007</td>
<td>Books</td>
<td>$344,562</td>
<td>22.04%</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$4,129,652</td>
<td>17.46%</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$91,602</td>
<td>6.87%</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$64,682</td>
<td>6.88%</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Notice that the 12 rows are ranked in ascending order from 1 to 12 by their profit margin values. You then create a view filter qualification to display...
data only when profit is less than $1,000,000. The view filter and resulting report are shown below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Profit Margin</th>
<th>Rank (Profit Margin)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Books</td>
<td>$251,345</td>
<td>22.04%</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$55,041</td>
<td>6.66%</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$43,243</td>
<td>6.90%</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2006</td>
<td>Books</td>
<td>$294,789</td>
<td>22.32%</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$71,068</td>
<td>6.88%</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$53,666</td>
<td>7.17%</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2007</td>
<td>Books</td>
<td>$344,562</td>
<td>22.04%</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$91,602</td>
<td>6.87%</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$64,682</td>
<td>6.86%</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Notice that while some data no longer appears on the report, the values of the Rank (Profit Margin) derived metric remain the same. This allows you to view the rank of profit margins as applied over all the data for the report, including the data that has been filtered from view.

Since the Rank (Profit Margin) metric is a derived metric, you can modify the report so that the metric’s values are relative to the new view of data supplied by the view filter, as shown in the report below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Profit Margin</th>
<th>Rank (Profit Margin)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Books</td>
<td>$251,345</td>
<td>22.04%</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$55,041</td>
<td>6.66%</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$43,243</td>
<td>6.90%</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2006</td>
<td>Books</td>
<td>$294,789</td>
<td>22.32%</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$71,068</td>
<td>6.88%</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$53,666</td>
<td>7.17%</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2007</td>
<td>Books</td>
<td>$344,562</td>
<td>22.04%</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$91,602</td>
<td>6.87%</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$64,682</td>
<td>6.86%</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>
You can apply this type of analysis in one of two ways:

- Evaluate the view filter qualification at the report level. This causes the view filter to be evaluated before any derived metrics. The derived metrics then calculate their data using the view of data created by the view filter. For information on evaluating view filter qualifications at the report level, see *Evaluating qualifications at the report or grid level, page 213*.

- Evaluate the view filter qualification at the grid level and change the evaluation order of derived metrics to be calculated after the view filter. For information on changing the evaluation order of individual report objects such as view filter qualifications and derived metrics, see the *Advanced Reporting Guide*.

**Using derived metrics with relative functions in view filters**

Derived metrics with relative functions such as RunningSum or Rank return values that are relative to the data on a report. When data is restricted by view filter qualifications on derived metrics with relative functions, you can allow the relative values to reflect the new view of data, or retain the relative values for all report data.

When derived metrics with relative functions are used to define view filters, you can allow the relative values to reflect the new view of data or retain the relative values for all report data.

For example, consider a report created in the MicroStrategy Tutorial project with Year, Category, Profit, Profit Margin, and a percent-to-total shortcut
derived metric named Percent to Total Over Rows (Profit Margin), as shown below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Books</td>
<td>$251,345</td>
<td>22.04%</td>
<td>2.17%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$2,768,984</td>
<td>17.43%</td>
<td>23.96%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$55,041</td>
<td>6.69%</td>
<td>0.48%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$43,243</td>
<td>6.90%</td>
<td>0.37%</td>
</tr>
<tr>
<td>2006</td>
<td>Books</td>
<td>$294,789</td>
<td>22.32%</td>
<td>2.55%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$3,399,107</td>
<td>17.56%</td>
<td>29.32%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$71,068</td>
<td>6.88%</td>
<td>0.61%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$53,666</td>
<td>7.17%</td>
<td>0.46%</td>
</tr>
<tr>
<td>2007</td>
<td>Books</td>
<td>$344,562</td>
<td>22.04%</td>
<td>2.98%</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>$4,129,652</td>
<td>17.46%</td>
<td>35.73%</td>
</tr>
<tr>
<td></td>
<td>Movies</td>
<td>$91,602</td>
<td>6.87%</td>
<td>0.79%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$64,682</td>
<td>6.88%</td>
<td>0.56%</td>
</tr>
</tbody>
</table>

Notice that the percent-to-total values display the distribution of profits over all 12 rows of data. You then create a view filter qualification based on the percent-to-total derived metric to display data only when the percent-to-total profit is less than two percent (.02). The view filter and resulting report are shown below.

While it appears that data is being shown for percent-to-totals greater than two percent, this is because the Percent to Total Over Rows (Profit) derived metric values have dynamically changed to reflect the new view of data.
displayed on the report. A grand total is displayed to show that the derived metric values add up to 100%, even though these rows of data combined account for less than four percent of the profits of the original report. This analysis is applied because the view filter qualification is evaluated at the report level by default.

The report above provides a view of percent-to-total profit data as displayed on the report. However, view filter qualifications on derived metrics can also be evaluated at the grid level so that the derived metrics retain their values that reflect all data available for the report, as shown in the report below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Category</th>
<th>Profit</th>
<th>Profit Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Movies</td>
<td>$55,041</td>
<td>6.69%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$43,243</td>
<td>6.90%</td>
</tr>
<tr>
<td>2006</td>
<td>Movies</td>
<td>$71,068</td>
<td>6.88%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$53,566</td>
<td>7.17%</td>
</tr>
<tr>
<td>2007</td>
<td>Movies</td>
<td>$91,502</td>
<td>6.87%</td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>$64,582</td>
<td>6.88%</td>
</tr>
<tr>
<td>Grand Total</td>
<td></td>
<td></td>
<td>3.28%</td>
</tr>
</tbody>
</table>

Notice that the percent-to-total profit values now appear to be less than two percent and reflect the values of the original report that included all report data. This is also reflected in the grand total of 3.28%.

As these scenarios illustrate, you have two options to evaluate view filter qualifications based on derived metrics with relative functions, summarized below:

- Evaluate the view filter qualification at the report level (default). This causes the derived metric values to dynamically reflect the new view of data on the report after the view filter qualification is applied.

- Evaluate the view filter qualification at the grid level. This causes the derived metric to retain its values that reflect all data available for the report.

For information on evaluating view filter qualifications at the report level versus the grid level, see *Evaluating qualifications at the report or grid level, page 213.*
View filter effects on metrics with relative functions

Metrics with relative functions (OLAP functions and Rank and Ntile functions) such as RunningSum or Rank return values that are relative to other values on a report. When data is restricted by view filter qualifications, some of the report data is hidden from view while the data available is not changed. This can cause the values returned by metrics with relative functions to appear to have incorrect values for the data displayed on the report.

For example, consider a report with Customer Group, Year, Average Net Sales Order Amount per Customer, and RunningAvg (Average Net Sales Order Amount per Customer), as shown below.

The report below and the metric using the RunningAvg relative function were created using the Sales and Distribution Analysis Module.

<table>
<thead>
<tr>
<th>Customer Group</th>
<th>Year</th>
<th>Metrics</th>
<th>Average Net Sales Order Amount per Customer</th>
<th>RunningAvg (Avg Net Sales Order Amount per Customer)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retailers</td>
<td>2006</td>
<td>$5,019,000</td>
<td>$6,019,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2007</td>
<td>$5,024,400</td>
<td>$5,521,700</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$460,000</td>
<td>$3,834,467</td>
<td></td>
</tr>
<tr>
<td>Wholesalers</td>
<td>2006</td>
<td>$575,000</td>
<td>$575,000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2007</td>
<td>$1,695,100</td>
<td>$1,135,050</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$750,000</td>
<td>$1,006,700</td>
<td></td>
</tr>
<tr>
<td>Large Company</td>
<td>2006</td>
<td>$225,750</td>
<td>$225,750</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2007</td>
<td>$3,324,333</td>
<td>$1,775,042</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$3,934,375</td>
<td>$2,494,819</td>
<td></td>
</tr>
<tr>
<td>Medium Company</td>
<td>2006</td>
<td>$116,500</td>
<td>$116,500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2007</td>
<td>$409,913</td>
<td>$263,206</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$114,000</td>
<td>$213,471</td>
<td></td>
</tr>
</tbody>
</table>
You then create a view filter to restrict data to only years 2007 and 2008. The view filter qualification and resulting report are shown below.

<table>
<thead>
<tr>
<th>New</th>
<th>Clear</th>
<th>✔️ Auto-Apply changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year In list</td>
<td>(2008, 2007)</td>
<td></td>
</tr>
</tbody>
</table>

Click here to start a new qualification

<table>
<thead>
<tr>
<th>Customer Group</th>
<th>Year</th>
<th>Average Net Sales Order Amount per Customer</th>
<th>RunningAvg (Avg Net Sales Order Amount per Customer)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retailers</td>
<td>2007</td>
<td>$5,024,400</td>
<td>$5,521,700</td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$460,000</td>
<td>$3,834,467</td>
</tr>
<tr>
<td>Wholesalers</td>
<td>2007</td>
<td>$1,695,100</td>
<td>$1,135,050</td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$750,000</td>
<td>$1,006,700</td>
</tr>
<tr>
<td>Large Company</td>
<td>2007</td>
<td>$3,324,333</td>
<td>$1,775,042</td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$3,934,375</td>
<td>$2,494,819</td>
</tr>
<tr>
<td>Medium Company</td>
<td>2007</td>
<td>$409,913</td>
<td>$263,206</td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>$114,000</td>
<td>$213,471</td>
</tr>
</tbody>
</table>

Notice that the values for the RunningAvg (Average Net Sales Order Amount per Customer) metric do not change, only the 2006 values are hidden. The values displayed do not accurately reflect the view of data and instead reflect the data available for the entire report, including the 2006 data hidden from view.

This is because the view filter is calculated without re-executing SQL against the data warehouse, and is evaluated after calculating the metrics on the report. Therefore, the metric is not recalculated to reflect the view of data shown on the report as restricted by the view filter.

If you plan to use metrics with relative functions and require them to reflect the data displayed on the report, you can use one of the options summarized below:

- Use report filters rather than view filters. A report filter causes a report to re-execute its SQL against the data warehouse, which can cause more processing time than a view filter. However, this allows a metric with a relative function to recalculate its values based on the filtering criteria.
- Use derived metrics with relative functions rather than standard metrics. Derived metrics can be evaluated after view filters and thus reflect the view of data on a report without having to re-execute SQL. However, derived metrics cannot be saved for use in multiple reports and can
require modification to work as intended with view filters. For information on the interaction between view filters and derived metrics, see View filter effects on derived metrics, page 220.

View filter effects on dynamic aggregation

Dynamic aggregation occurs when an attribute is moved between the report layout and the Report Objects pane. This type of aggregation dynamically affects the data returned for the report. For more information on dynamic aggregation, refer to Chapter 5, Dynamic Aggregation.

A view filter also dynamically affects the data returned for a report, by restricting the report data using various types of qualifications. You can use both view filters and dynamic aggregation to return different analyses of report data, as described in the sections listed below:

- **Using attribute qualifications on hidden attributes, page 228**: When report data is restricted using a view filter attribute qualification, this can affect the data displayed on the report even if the attribute it qualifies on is not displayed on the report grid.

- **Using metric qualifications at the report or grid level, page 230**: When report data is restricted using a view filter metric qualification, this can affect the data displayed on the report in various ways depending on when the view filter metric qualification is evaluated.

Using attribute qualifications on hidden attributes

When report data is restricted using a view filter attribute qualification, this can affect the data displayed on the report even if the attribute it qualifies on is not displayed on the report grid.
For example, consider the View Filter - Dynamic Aggregation report in the MicroStrategy Tutorial project, as shown below.

Notice that dynamic aggregation has been used to remove the Employee attribute from the report grid. However, there is also a view filter attribute qualification on the Employee attribute. Even though Employee is not shown on the report, the data displayed is restricted by the view filter to only display data for the employees Caitlin Bell, Beatrice Conner, Andrew Johnson, Laura Kelly, and Jack Kieferson. This can be verified by using dynamic aggregation to drag and drop Employee on the report grid, as shown below.

Data is dynamically aggregated to be displayed at the region and employee level. A subtotal has been added to the report above to show that the original report was only displaying data for the employees listed in the view filter attribute qualification. For example, notice that the revenue, cost, and profit data for the Northeast region of the original report matches the values for the Northeast subtotal values of the report above.
Using metric qualifications at the report or grid level

When report data is restricted using a view filter metric qualification, this can affect the data displayed on the report in various ways depending on when the view filter metric qualification is evaluated.

For example, consider the View Filter - Dynamic Aggregation report in the MicroStrategy Tutorial project, as shown below.

Notice that dynamic aggregation has been used to remove the Employee attribute from display on the report grid. You then create a view filter metric qualification to restrict the report data for profits greater than $300,000. The view filter qualification and resulting report are shown below.

The view filter metric qualification is evaluated at the grid level by default. This means that the dynamic aggregation of removing Employee from the report layout is evaluated first, and then the view filter metric qualification restricts data based on the remaining data displayed on the report.

With this evaluation order, the view filter metric qualification returns regions with profits greater than $300,000.
However, you can also evaluate view filter metric qualifications at the report level. You can right-click the Profit Greater than $300,000 qualification, and select **Apply Condition at the {Employee, Region} level**. Evaluating the metric qualification at the report level returns the report results shown below.

Notice that only data for the Northeast region is returned, and all metric values are lower. This is because a view filter metric qualification evaluated at the report level includes all attributes, in the calculation to restrict data from the report.

Data is first restricted to employees with profits greater than $300,000, and then this data is aggregated and displayed at the region level. In the first report the Southwest region included two employees who combined to have more than $300,000 in profit, but neither employee had more than $300,000 alone. These employees are restricted by the view filter and are not included when aggregating the data at the region level.

These two options provide two different types of analysis on report data, summarized below:

- **Evaluating view filter metric qualifications at the grid level**: When view filter metric qualifications are evaluated at the grid level and dynamic aggregation is used, only the attributes displayed on the report grid are used to determine the data restricted from the report.

  In the example above, any regions with profits greater than $300,000 for the included employees are displayed on the report.

- **Evaluating view filter metric qualifications at the report level**: When view filter metric qualifications are evaluated at the report level and dynamic aggregation is used, all attributes in the Report Objects pane are used to determine the data restricted from the report. This includes attributes that are not on the report grid.
In the example above, any employees with profits greater than $300,000 for the included employees are displayed on the report. The data for any remaining employees is then aggregated and displayed at the region level.

For information on evaluating view filter qualifications at the report level versus the grid level, see Evaluating qualifications at the report or grid level, page 213.

View filter effects on derived elements

View filters restrict data on reports by enabling users to create qualifications on attributes and metrics. These qualifications then determine what data is shown on the report, as well as what data is available to analyze and format with derived elements.

Derived elements cannot be selected in a view filter, and therefore cannot be a part of a view filter qualification that restricts the report results. View filters can only include qualifications on attributes and metrics.

However, derived elements are associated with attribute elements. Due to this association, any view filter qualifications that restrict the data for a derived element’s associated attribute elements also restricts data returned for the derived element.

For example, a report includes a Group derived element defined to combine profit values for the Music and Movies elements of the Category attribute. This report is shown below.

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Music and Movies</td>
<td>$379,300</td>
<td></td>
</tr>
<tr>
<td>Books</td>
<td>$890,696</td>
<td></td>
</tr>
<tr>
<td>Electronics</td>
<td>$10,287,744</td>
<td></td>
</tr>
</tbody>
</table>

The Music and Movies derived element combines the profit values for the Music attribute element and Movies attribute element. You then create a view filter qualification that restricts the report data to the individual Movies
attribute element and Electronics attribute element. The report results are shown below.

<table>
<thead>
<tr>
<th>View filter</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>✅ Auto-Apply changes</td>
</tr>
<tr>
<td>Category In list</td>
<td>(Electronics, Movies)</td>
</tr>
<tr>
<td>Click here to start a new qualification</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
<th>Metrics</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Music and Movies</td>
<td></td>
<td>$217,710</td>
</tr>
<tr>
<td>Electronics</td>
<td></td>
<td>$10,287,744</td>
</tr>
</tbody>
</table>

Notice that the Music and Movies derived element is still displayed, but the profit value has decreased. This is because the view filter has restricted the data to only Movies and Electronics. The Music and Movies derived element can only return profit values for the available Movies attribute element data. When using view filters and derived elements on the same report, be aware that any view filter qualifications restrict the data available to analyze and format with derived elements.
**Introduction**

The general goal of reporting on data in Intelligent Cubes is to create reports that are linked to a published Intelligent Cube. Dynamic sourcing makes Intelligent Cubes more accessible by allowing regular reports to automatically access published Intelligent Cubes that can satisfy the data requirements of the report.

This can improve the performance of reports, which get executed against an Intelligent Cube, without having to manually find and select the Intelligent Cube that fits the analysis of the report. If the data requirements for the report change, or cannot be satisfied by an available Intelligent Cube, the reports automatically access the data warehouse to return the required information.

To support dynamic sourcing, you will need to configure your projects, Intelligent Cubes, reports, attributes, metrics, and any aggregate tables you use. This appendix describes the benefits of dynamic sourcing, the necessary steps to support dynamic sourcing, and how to track how successful dynamic sourcing is for your project. This chapter assumes you are familiar with the information in *Chapter 2, Sharing Sets of Data Among Reports: Intelligent Cubes*.
The sections listed below describe the following information:

- Scenarios that benefit from dynamic sourcing, page 236
- Best practices for supporting dynamic sourcing, page 238
- Configuring dynamic sourcing, page 245
- Using Cube Advisor to support dynamic sourcing, page 260
- Tracking the use of dynamic sourcing, page 272

**Scenarios that benefit from dynamic sourcing**

Dynamic sourcing can greatly improve the performance of your projects, because it allows reports to automatically detect an Intelligent Cube that can satisfy its data requirements, and execute against the Intelligent Cube rather than the data warehouse.

This section also discusses how you can tune your project to better support dynamic sourcing for these features, as described in Tuning your project for dynamic sourcing, page 237.

Dynamic sourcing compliments the ability to create reports that are connected to a specific Intelligent Cube. This feature gives you the following performance benefits:

- Report designers do not need to know whether an Intelligent Cube includes the information they need, or which Intelligent Cube they should use. With dynamic sourcing, an Intelligent Cube that can satisfy the report’s data requirements is detected automatically, without the report designer having to consider which Intelligent Cubes are available.

- The performance of pre-existing reports can be improved without having to modify the report to access a specific Intelligent Cube. Dynamic sourcing can allow these reports to automatically detect an Intelligent Cube that satisfies the report’s data requirements.

- The performance of prompted reports can be greatly improved. Prompted reports can cause performance issues, because it is difficult to use report caches with them. When different prompt answers are chosen, a report cache cannot return information for the report and the report request must be submitted through the data warehouse again.
With dynamic sourcing, Intelligent Cubes can provide a set of data that can satisfy the data requirements of reports executed with different prompt answers.

- Reports can drill from one Intelligent Cube to another Intelligent Cube.

### Tuning your project for dynamic sourcing

You can tune your project to provide a reporting environment that is more likely to be able to take advantage of dynamic sourcing, as described in the sections listed below:

- *Tuning Intelligent Cubes for dynamic sourcing, page 237*
- *Tracking the success of dynamic sourcing, page 238*

For information on the various features that are supported for dynamic sourcing, see *Best practices for supporting dynamic sourcing, page 238*.

#### Tuning Intelligent Cubes for dynamic sourcing

One of the most important aspects in providing dynamic sourcing for as many reports as possible in your project is the creation and availability of Intelligent Cubes. Once Intelligence Server determines which Intelligent Cubes can satisfy a report’s needs, it automatically selects the Intelligent Cube that offers the best performance for the report.

You can improve the chances that a report will use dynamic sourcing by making more Intelligent Cubes available to it. When creating additional Intelligent Cubes, however, you must consider that Intelligent Cubes can consume a significant amount of Intelligent Server’s memory. If you can determine the objects and type of analysis most commonly used in reports, you can use this information to tailor the Intelligent Cubes to these requirements.

Be aware that enabling dynamic sourcing requires some system overhead to determine whether a report can access an Intelligent Cube rather than the data warehouse. This overhead increases as the number of Intelligent Cubes enabled for dynamic sourcing increases. However, the overhead only affects Intelligence Server’s performance if there are over a thousand Intelligent Cubes, which is an unlikely business scenario.
Tracking the success of dynamic sourcing

If you enable dynamic sourcing for a project, you can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports. By determining why dynamic sourcing failed for a given report, you can modify your reports, Intelligent Cubes, and other objects to allow reports in your project to use dynamic sourcing. Tracking dynamic sourcing information is described in Tracking the use of dynamic sourcing, page 272.

Best practices for supporting dynamic sourcing

For a report to successfully retrieve its information from an Intelligent Cube, various conditions must be met. The considerations and steps required to fully support dynamic sourcing are listed below:

• While many MicroStrategy features can be used with dynamic sourcing, some features prevent the use of dynamic sourcing between reports and Intelligent Cubes. Refer to the following sections for a list of features that can and cannot be used with dynamic sourcing:
  - Features that can be used with dynamic sourcing, page 240
  - Features that prevent the use of dynamic sourcing, page 241
• You can use Cube Advisor to create Intelligent Cubes that will allow as many reports as possible to use dynamic sourcing. For steps to use Cube Advisor to support dynamic sourcing, see Using Cube Advisor to support dynamic sourcing, page 260.
• For an Intelligent Cube to satisfy the data requirements of a report, the Intelligent Cube and report must have matching settings for VLDB properties that can affect the data that is returned.
• If you use dynamic sourcing, you must keep your Intelligent Cubes up to date with any changes to the data in your data warehouse, otherwise reports can return outdated data. You can use schedules and subscriptions to update the data of Intelligent Cubes. For information on these techniques, see Updating Intelligent Cubes using schedules, page 32 and Publishing Intelligent Cubes using a schedule, page 41.
• In general, you should not create restrictive filter qualifications on Intelligent Cubes. The more restrictive the filter, the less likely it is that the Intelligent Cube can satisfy the data requirements of reports.
Restrictive filters can be applied to the reports that access Intelligent Cubes.

- Reports cannot combine the data from multiple Intelligent Cubes to use dynamic sourcing. All the data requirements of a report must be satisfied by a single Intelligent Cube.

- Dynamic sourcing must be enabled for projects, reports, and Intelligent Cubes; this is described in the sections listed below:
  - Enabling or disabling dynamic sourcing for projects, page 245
  - Enabling or disabling dynamic sourcing for reports, page 248
  - Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250

- To ensure that correct data is available in an Intelligent Cube for a report, you must verify that the following objects can support dynamic sourcing:
  - Attributes are available for dynamic sourcing by default. You should disable dynamic sourcing for attributes if:
    - Attribute data in fact and lookup tables contains NULL values.
    - The attribute elements in fact and lookup tables are not identical.
    For steps to disable dynamic sourcing for attributes, see Disabling dynamic sourcing for attributes, page 252.

  - Metrics are available for dynamic sourcing by default. You should disable dynamic sourcing for metrics if metric data in fact tables contains NULL values. For steps to disable dynamic sourcing for metrics, see Disabling dynamic sourcing for metrics, page 256.

  - Aggregate tables are available for dynamic sourcing by default. You should disable dynamic sourcing for aggregate tables if:
    - Aggregation functions other than Sum are used.
    - The aggregate table includes different data than is available in lookup and fact tables. For example, an aggregate table with years 2006, 2007, and 2008 should not be used for dynamic sourcing if your lookup and fact tables only include the years 2007 and 2008.
    For steps to disable dynamic sourcing for aggregate tables, see Disabling dynamic sourcing for aggregate tables, page 258.

- If the report is based on an MDX data source, such as an SAP BW Cube, it can also use dynamic sourcing, and retrieve data from an Intelligent Cube that is also based on an MDX data source.
In such a scenario, the following additional conditions apply:

- The report and Intelligent Cube must be based on the same source MDX cube.
- Filters on the report must meet the following criteria:
  - Attributes used in the filter should also be on the Intelligent Cube’s definition.
  - The report filter must include at least the same restrictions as the Intelligent Cube’s filter, if present. For example, if the Intelligent Cube restricts data to only the year 2010, the report must include the same restriction.

For additional information on creating reports that access MDX sources, refer to the MicroStrategy MDX Cube Reporting Guide.

**Features that can be used with dynamic sourcing**

Features that can be supported by dynamic sourcing include, but are not limited to, the list below:

- Security filters
- Compound attributes and joint element lists
- Conditional metrics
- Prompts in a report
- Transformation metrics, under the following conditions:
  - A metric with the same formula and transformation exists in the Intelligent Cube.
  - For the hierarchy used for the transformation, the report must be calculated at the same level as the Intelligent Cube. For example, if the transformation is along the Time hierarchy and the Intelligent Cube is calculated at the Month level, the report must also be calculated at the Month level.
- Metrics that include Count, Sum, Min, or Max in their formulas
- Metrics that include Count Distinct in their formula for reports and Intelligent Cubes that share the same level of attribute data
• Nested metrics for which the inner metric and all attributes used for aggregation are available in the Intelligent Cube

• Non-aggregatable metrics for which the non-aggregation attribute is available in the Intelligent Cube

• Filter qualifications including the functions Exactly, Different from, In list, Not in list, Greater than, Less than, Greater than or equal to, Less than or equal to, Between, Not Between, Is Null, and Is Not Null for data types Numeric, BigDecimal, DateTime, and String

• OLAP Services features in the report, including:
  ▪ View filters
  ▪ Derived metrics
  ▪ Dynamic aggregation

• OLAP features used in a report, such as:
  ▪ Pivoting
  ▪ Banding
  ▪ Thresholds
  ▪ Page-by
  ▪ Outline mode
  ▪ Sorting
  ▪ Subtotals

Features that prevent the use of dynamic sourcing

Some features prevent the use of dynamic sourcing between Intelligent Cubes and report, as described in the sections listed below:

• Features that prevent reports from using dynamic sourcing, page 242

• Features that prevent Intelligent Cubes from being available for dynamic sourcing, page 243

In addition to these features, if you use filter qualifications on attribute forms in reports and Intelligent Cubes, a mismatch in the attribute forms used in the qualifications can prevent dynamic sourcing between the report and
Intelligent Cube. For example, a report includes a filter qualification on the ID form of the Customer attribute. An Intelligent Cube includes a filter qualification on the Description form of the Customer attribute. Since different attribute forms are used to restrict the data, a direct relationship between the data cannot be verified and dynamic sourcing cannot be used to connect the report to the Intelligent Cube.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in *Tracking the use of dynamic sourcing, page 272*.

**Features that prevent reports from using dynamic sourcing**

If any of the features listed below are present in reports, the reports are not able to use dynamic sourcing:

- Consolidations
- Custom groups
- Query Builder or Freeform SQL reports cannot use dynamic sourcing.
- MDX cube reports connected to SAP BI, Microsoft Analysis Services, or Hyperion Essbase cannot use dynamic sourcing.
- Set qualifications in the report filter or metric conditions:
  - Metric qualifications
  - Relationship qualifications that do not include a filter in their definition, or that use a Relate By option other than Use System Default.
- OR logical operator used in the report filter to combine qualifications which include different attributes or attribute forms.
- AND NOT or OR NOT logical operators
- Metrics that use passthrough functions such as ApplySimple. Alternatively, facts can be defined using passthrough functions such as ApplySimple. These facts can then be used to create metrics, which can be used on reports that can support dynamic sourcing.
- Freeform MDX metrics. On reports based on MDX data sources, these are analogous to metrics with passthrough functions.
- Metrics that use facts with extensions or degradations
• Data marts
• Report as filter used in the report filter
• Using any of the options listed below for the following VLDB properties:

<table>
<thead>
<tr>
<th>VLDB property</th>
<th>Options that prevent report from using dynamic sourcing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Downward outer join</td>
<td>Preserve all rows for metrics higher than template level without report filter</td>
</tr>
<tr>
<td></td>
<td>Do not do downward outer join for database that support full outer join</td>
</tr>
<tr>
<td>OLAP function support</td>
<td>Preserve backward compatibility with 8.1.x and earlier</td>
</tr>
</tbody>
</table>

**Features that prevent Intelligent Cubes from being available for dynamic sourcing**

If the features listed below are present in an Intelligent Cube, the Intelligent Cube cannot be used for dynamic sourcing:

• Set qualifications in the Intelligent Cube filter:
  ▪ Metric qualifications
  ▪ Relationship qualifications
• OR logical operator used in the Intelligent Cube filter to combine qualifications which include different attributes or attribute forms
• AND NOT or OR NOT logical operators used in the Intelligent Cube filter
• Passthrough functions such as ApplySimple
• Report as filter used in the Intelligent Cube filter
• Report limits
• For Intelligent Cubes based on MDX data sources, dynamic sourcing is not supported if the Intelligent Cube uses incremental refresh settings. For information on incremental refresh, see *Updating Intelligent Cubes without re-processing: Incremental Refresh, page 45.*

If the features listed below are present in an Intelligent Cube the features may not be available for dynamic sourcing, but the rest of the Intelligent Cube may be able to support dynamic sourcing:

• Metrics that use facts with fact extensions or degradations are not available for dynamic sourcing.
• Conditional metrics are not available for dynamic sourcing. Conditional metrics are described in the *Advanced Reporting Guide*.

• Compound metrics that are not defined as smart metrics in an Intelligent Cube are not available for dynamic sourcing.

**Dynamic sourcing for incrementally refreshed Intelligent Cubes**

Apart from the best practices listed above, additional considerations apply when you are using Intelligent Cubes that are incrementally refreshed. For information on incrementally refreshing Intelligent Cubes, see *Updating Intelligent Cubes without re-processing: Incremental Refresh, page 45*.

**Intelligent Cube refresh options supported for dynamic sourcing**

The following Intelligent Cube refresh options are supported for dynamic sourcing:

• Full refresh
• Dynamic refresh
• Insert and Update, only if data is being updated for one attribute form

For information on Intelligent Cube refresh settings, see *Defining Intelligent Cube refresh settings, page 46*.

**Incremental refresh types supported for dynamic sourcing**

You can create an incremental refresh filter or report to refresh the data in an Intelligent Cube. If the Intelligent Cube is refreshed using a filter, the following increment types are supported:

• Update Only
• Insert, Update and Delete, only if the data is being updated for one attribute form

If the Intelligent Cube is refreshed using an incremental refresh report, it cannot be used for dynamic sourcing. For information on incremental refresh filters and reports, see *Defining an incremental refresh filter or report, page 47*. 
Configuring dynamic sourcing

To enable reports to dynamically connect to an Intelligent Cube for their data requirements, you must enable various aspects of the MicroStrategy system for dynamic sourcing.

In general, you must enable projects, reports, and Intelligent Cubes for dynamic sourcing. Depending on the objects available in or accessed by the reports and Intelligent Cubes, you may also need to define other objects as available for dynamic sourcing. The sections listed below describe the requirements and considerations for configuring dynamic sourcing for various aspects of the MicroStrategy system:

- **Enabling or disabling dynamic sourcing for projects, page 245**
- **Enabling or disabling dynamic sourcing for reports, page 248**
- **Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250**
- **Disabling dynamic sourcing for attributes, page 252**
- **Disabling dynamic sourcing for metrics, page 256**
- **Disabling dynamic sourcing for aggregate tables, page 258**
- **Accessing the dynamic sourcing VLDB properties for a project, page 247**

Enabling or disabling dynamic sourcing for projects

By default, dynamic sourcing is enabled for projects. However, you can disable it for a project to prevent reports from connecting to Intelligent Cubes through the use of dynamic sourcing. This allows project administrators to have project-wide control over the use of dynamic sourcing.

The steps below show you how to enable or disable dynamic sourcing for projects as well as define default dynamic sourcing behavior for Intelligent Cubes.

For information on defining default dynamic sourcing behavior for reports, Intelligent Cubes, attributes, metrics, and aggregate tables, see *Accessing the dynamic sourcing VLDB properties for a project, page 247*. 
Prerequisites

- A project has been created in MicroStrategy.

Enabling or disabling dynamic sourcing for a project

1. In MicroStrategy Desktop, log in to a project source with a user account that has administrative privileges.

2. Right-click a project and select Project Configuration. The Project Configuration Editor opens.

3. In the Categories list, expand Intelligent Cubes, and then select General.

4. Perform one of the following to either enable or disable dynamic sourcing:
   - To enable dynamic sourcing for the project, select the Enable dynamic sourcing check box.
   - To disable dynamic sourcing for the project, clear the Enable dynamic sourcing check box.

5. You can define a default dynamic sourcing behavior for all Intelligent Cubes in a project using the check boxes described below:
   - **Make Intelligent Cubes available for dynamic sourcing by default:** Select this check box to enable dynamic sourcing for all Intelligent Cubes in a project. You can clear this check box to disable dynamic sourcing as the default behavior for all Intelligent Cubes in a project.

   You can enable and disable dynamic sourcing for individual Intelligent Cubes, as described in Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250.

   - **Allow dynamic sourcing even if outer join properties are not set:** Select this check box to allow reports to connect to Intelligent Cubes using dynamic sourcing even when some outer join properties are not defined. However, this can cause reports to return incorrect data in scenarios when outer joins would be helpful.

   For example, a report includes the attribute Day and the metric Revenue. It connects to an Intelligent Cube that includes the attribute Day, and the metrics Revenue and Cost. For some days there is data
for Revenue, but no data for Cost. If the Intelligent Cube does not support any outer joins, then the data for Revenue for which there is no data for Cost is not included in the final result. In this scenario, the report cannot return complete information from the Intelligent Cube without outer joins.

It is recommended that you define your Intelligent Cube to support and use outer joins when necessary, as described in *Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250*. This ensures that all data is returned. Be aware that outer joins can cause additional load on your database and may require larger Intelligent Cubes.

6 Click **OK** to save your changes to the project and close the Project Configuration Editor.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in *Tracking the use of dynamic sourcing, page 272*.

**Accessing the dynamic sourcing VLDB properties for a project**

A number of VLDB properties control whether dynamic sourcing is enabled for reports, attributes, metrics, and aggregate tables. Defining these VLDB properties at the project level allows you to define a default dynamic sourcing behavior for these objects. While this defines the default behavior, any modifications to the VLDB properties for a specific report, attribute, or metric take precedence over the project-wide default.

The steps below show you how to access the dynamic sourcing VLDB properties for a project to define project-wide defaults, and includes links to information on how to set VLDB properties for each object type.

---

**To define dynamic sourcing VLDB properties for a project**

1 In MicroStrategy Desktop, log in to a project source with a user account with administrative privileges.

2 Right-click a project and select **Project Configuration**. The Project Configuration Editor opens.
3 In the **Categories** list, expand **Project definition**, and then select **Advanced**.

4 In the **Analytical Engine VLDB Properties** area, click **Configure**. The VLDB Properties Editor opens.

5 From the **Tools** menu, select the **Show Advanced Settings** option, if it is not already selected.

6 In the **VLDB Settings** list, expand **Dynamic Sourcing**. The dynamic sourcing VLDB properties are displayed. For information on how each VLDB property can be used to enable or disable dynamic sourcing, refer to the sections listed below:

- **Enable Dynamic Sourcing for Report**: *Enabling or disabling dynamic sourcing for reports, page 248*
- **Attribute Validation**: *Disabling dynamic sourcing for attributes, page 252*
- **Metric Validation**: *Disabling dynamic sourcing for metrics, page 256*
- **Aggregate Table Validation**: *Disabling dynamic sourcing for aggregate tables, page 258*
- **String Comparison Behavior**: *Supporting filtering on attributes for dynamic sourcing, page 254*

7 Click **Save and Close** to save your changes to the VLDB properties and close the VLDB Properties Editor.

8 Click **OK** to save your changes to the project and close the Project Configuration Editor.

**Enabling or disabling dynamic sourcing for reports**

By default, dynamic sourcing is enabled for reports. You can enable dynamic sourcing for a report so that active Intelligent Cubes (that are also enabled for dynamic sourcing) are checked to see if the report can retrieve its data from an Intelligent Cube. If an Intelligent Cube fits the data requirements of a report, the report can be run without executing against the data warehouse.
You can enable dynamic sourcing for reports by modifying the **Enable Dynamic Sourcing for Report** VLDB property. This VLDB property has the following options:

- **Disable dynamic sourcing for report**: Dynamic sourcing is disabled for reports.
- **Enable dynamic sourcing for report**: Dynamic sourcing is enabled for reports.

You can enable dynamic sourcing for reports individually or you can enable dynamic sourcing for all reports within a project. While the definition of the VLDB property at the project level defines a default for all reports in the project, any modifications at the report level take precedence over the project level definition. For information on accessing the VLDB Properties Editor for a project to define a default dynamic sourcing option for all reports, see *Accessing the dynamic sourcing VLDB properties for a project, page 247*.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in *Tracking the use of dynamic sourcing, page 272*.

The procedure below describes how to enable or disable dynamic sourcing for an individual report.

**Prerequisites**

- A report has been created in MicroStrategy.

---

**To enable or disable dynamic sourcing for a report**

1. In MicroStrategy Desktop, browse to a report, and then right-click the report and select **Edit**. The report opens in the Report Editor.

2. From the **Data** menu, select **VLDB Properties**. The VLDB Properties Editor opens.

3. From the **Tools** menu, select the **Show Advanced Settings** option if it is not already selected.

4. In the **VLDB Settings** list, expand **Dynamic Sourcing**, and then select **Enable Dynamic Sourcing for Report**.

5. Clear the **Use default inherited value** check box.
6 Select one of the following options, depending on whether you want to disable or enable dynamic sourcing for the report:

- To disable dynamic sourcing for the report, select **Disable dynamic sourcing for report**.
- To enable dynamic sourcing for the report, select **Enable dynamic sourcing for report**.

7 Click **Save and Close** to save your changes to VLDB properties and close the VLDB Properties Editor.

8 Click **Save and Close** to save the report and close the Report Editor.

**Enabling or disabling dynamic sourcing for Intelligent Cubes**

You can enable dynamic sourcing for all Intelligent Cubes in a project, as described in *Enabling or disabling dynamic sourcing for projects, page 245*, or you can enable dynamic sourcing for each Intelligent Cube individually.

By default, dynamic sourcing is disabled for Intelligent Cubes, and they are therefore unavailable for reports checking whether Intelligent Cubes can satisfy their data requirements.

While enabling dynamic sourcing for Intelligent Cubes at the project level defines a default for all Intelligent Cubes in the project, any modifications for individual Intelligent Cubes take precedence over the project level definition.

The steps below show you how to enable or disable dynamic sourcing for an individual Intelligent Cube.

**Prerequisites**

- An Intelligent Cube has been created in a project.

**To enable or disable dynamic sourcing for an Intelligent Cube**

1 In MicroStrategy Desktop, browse to an Intelligent Cube, and then right-click the Intelligent Cube and select **Edit**. The Intelligent Cube opens in the Report Editor.
2 From the Data menu, select Configure Intelligent Cube. The Intelligent Cube Options dialog box opens.

3 In the Dynamic Sourcing area, use the Make Intelligent Cubes available for dynamic sourcing by default options to enable or disable dynamic sourcing for the Intelligent Cube:

- To enable dynamic sourcing for the Intelligent Cube, select Enabled.
- To disable dynamic sourcing for the Intelligent Cube, select Disabled.
- To specify that the Intelligent Cube inherits its dynamic sourcing behavior from the project settings, select Use Default Project-Level Behavior.

4 In the Dynamic Sourcing area, use the Allow dynamic sourcing even if outer join properties are not set options to determine if reports can connect to Intelligent Cubes using dynamic sourcing even when some outer join properties are not defined:

- **Enabled**: Select this option to allow reports to connect to Intelligent Cubes using dynamic sourcing even when some outer join properties are not defined. However, this can cause incorrect data to be returned in scenarios when outer joins would be helpful.

  For example, a report includes the attribute Day and the metric Revenue. It connects to an Intelligent Cube that includes the attribute Day and the metric Revenue, and it also includes the metric Cost. For some days there is data for Revenue, but there is no data for Cost. If the Intelligent Cube does not support any outer joins, then the values for Revenue which do not have corresponding values for Cost aren’t returned. In this scenario, the report cannot return complete information from the Intelligent Cube without outer joins.

- **Disabled**: Select this option to prohibit reports from connecting to Intelligent Cubes using dynamic sourcing when some outer join properties are not defined. Selecting this option avoids the possibility of displaying incorrect data in reports.

  You can define your Intelligent Cube to support and use outer joins when necessary. This ensures all data is returned. However, outer joins can cause additional load on your database and require larger Intelligent Cubes. You can enable this support by defining the Metric Join Type, described below.

  - **Metric Join Type**: Any metrics included in the Intelligent Cube that are to be available for dynamic sourcing must be defined to use outer joins in the Intelligent Cube. With the Intelligent Cube
open, from the Data menu, select Report Data Options. In the Report Data Options dialog box, expand Calculations, and select Metric Join Type. For each metric to make available for dynamic sourcing, change the Join Type to Outer. Click OK to save your changes.

For details and examples of VLDB properties, see the Supplemental Reference for System Administration.

• Use Default Project-Level Behavior: Select this option to define the Intelligent Cube to inherit its dynamic sourcing behavior from the project settings discussed in Enabling or disabling dynamic sourcing for projects, page 245.

5 Click OK. The Intelligent Cube Options dialog box closes and you are returned to the Intelligent Cube.

6 Click Save and Close to save your changes and close the Report Editor.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in Tracking the use of dynamic sourcing, page 272.

Disabling dynamic sourcing for attributes

Attributes are available for dynamic sourcing by default, but there are some data modeling conventions that should be considered when using dynamic sourcing.

In general, if attributes use outer joins, accurate data can be returned to reports from Intelligent Cubes through dynamic sourcing. However, if attributes use inner joins, which is more common, the resulting data set may be incomplete. In such cases, you should verify that the attribute data can be correctly represented through dynamic sourcing.

Two scenarios can cause attributes that use inner joins to return incorrect or incomplete data when dynamic sourcing is used:

• Attribute information in lookup and fact tables includes NULL values.
• All attribute elements in fact tables are not also present in lookup tables.

These scenarios are uncommon.
If some attributes fit these scenarios, you can disable dynamic sourcing when these attributes are used in reports and Intelligent Cubes. You can enable and disable dynamic sourcing for attributes by modifying the **Attribute Validation VLDB** property. This VLDB property has the following options:

- **Attribute columns in fact tables and lookup tables do not contain NULLs and all attribute elements in fact tables are present in lookup tables**: This is the default option for attributes, which enables attributes for dynamic sourcing.

- **Attribute columns in fact tables and lookup tables may contain NULLs and/or some attribute elements in fact tables are not present in lookup tables**: This option disables dynamic sourcing for attributes. This setting should be used if your attribute data is not modeled to support dynamic sourcing. The inclusion of NULLs in your attribute data, or a mismatch between available attribute data in your fact and lookup tables, can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing for attributes individually or you can disable dynamic sourcing for all attributes within a project. While the definition of the VLDB property at the project level defines a default for all attributes in the project, any modifications at the attribute level take precedence over the project level definition. For information on accessing the VLDB Properties Editor for a project to define a default dynamic sourcing option for all attributes, see *Accessing the dynamic sourcing VLDB properties for a project, page 247.*

The steps below show you how to disable or enable dynamic sourcing for an individual attribute. If your database is case-sensitive, you should also review *Supporting filtering on attributes for dynamic sourcing, page 254* to ensure that dynamic sourcing can correctly return data for your attributes.

**Prerequisites**

- An attribute has been created in a project.

---

**To enable or disable dynamic sourcing for an attribute**

1. In MicroStrategy Desktop, browse to an attribute, right-click on it, and select **Edit**. The attribute opens in the Attribute Editor.

2. From the **Tools** menu, select **VLDB Properties**. The VLDB Properties Editor opens.
From the Tools menu, select the Show Advanced Settings option if it is not already selected.

In the VLDB Settings list, expand Dynamic Sourcing, and then select Attribute Validation.

Clear the Use default inherited value check box.

Select one of the options depending on whether you want to disable or enable dynamic sourcing for an attribute:

- To enable attributes to use dynamic sourcing (the default option), select Attribute columns in fact tables and lookup tables do not contain NULLs and all attribute elements in fact tables are present in lookup tables.

- To disable dynamic sourcing for attributes unless outer joins are used for the attribute, select Attribute columns in fact tables and lookup tables may contain NULLs and/or some attribute elements in fact tables are not present in lookup tables. This setting should be used if your attribute data is not modeled to support dynamic sourcing.

Click Save and Close to save your changes to VLDB properties and close the VLDB Properties Editor.

Click Save and Close to save the attribute and close the Attribute Editor.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in Tracking the use of dynamic sourcing, page 272.

Supporting filtering on attributes for dynamic sourcing

To ensure that dynamic sourcing can return the correct results for attributes, you must also verify that filtering on attributes achieves the same results when executed against your database, versus an Intelligent Cube.

A filter on attributes can potentially return different results when executing against the database, compared to using dynamic sourcing to execute against an Intelligent Cube. This can occur if your database is case-sensitive and you create filter qualifications that qualify on the text data of attribute forms.

If your database is case-sensitive, this is enforced for the filter qualification. However, filtering for an Intelligent Cube is handled by the Analytical Engine which does not enforce case sensitivity.
Consider a filter qualification that filters on customers that have a last name beginning with the letter h. If your database is case-sensitive and uses uppercase letters for the first letter in a name, a filter qualification using a lowercase h is likely to return no data. However, this same filter qualification on the same data stored in an Intelligent Cube returns all customers that have a last name beginning with the letter h, uppercase or lowercase.

You can define attributes to either allow filter qualifications to be completed without enforcing case sensitivity, or to disable dynamic sourcing if these types of filters are used on attributes. You can configure this dynamic sourcing behavior for attributes by modifying the String Comparison Behavior VLDB property. This VLDB property has the following options:

- **Use case insensitive string comparison with dynamic sourcing**: This is the default option for attributes. When attempting to use dynamic sourcing, filter qualifications can qualify on the text data of attribute forms without enforcing case sensitivity.

  This is a good option if your database does not enforce case sensitivity. In this scenario, dynamic sourcing returns the same results that would be returned by the filter qualification if the report was executed against the database.

- **Do not allow any string comparison with dynamic sourcing**: This option disables dynamic sourcing for attributes when a filter qualification is used to qualify on the text data of attribute forms.

  This is a good option if your database is case sensitive. In this scenario, dynamic sourcing could return different results than what would be returned by the filter qualification if the report was executed against the database.

You can modify this VLDB property for attributes individually or you can modify it for all attributes within a project. While the definition of the VLDB property at the project level defines a default for all attributes in the project, any modifications at the attribute level take precedence over the project level definition. For information on accessing the VLDB Properties Editor for a project to define a default dynamic sourcing option for all attributes, see *Accessing the dynamic sourcing VLDB properties for a project, page 247*.

The procedure below describes how to modify the **String Comparison Behavior** VLDB property for an individual attribute.

**Prerequisites**

- An attribute has been created in a project.
To modify the String Comparison Behavior VLDB property for an attribute

1 In MicroStrategy Desktop, browse to an attribute, then right-click the attribute and select Edit. The attribute opens in the Attribute Editor.

2 From the Tools menu, select VLDB Properties. The VLDB Properties Editor opens.

3 From the Tools menu, select the Show Advanced Settings option if it is not already selected.

4 In the VLDB Settings list, expand Dynamic Sourcing, and then select String Comparison Behavior.

5 Clear the Use default inherited value check box.

6 Select one of the options depending on whether you want to disable or enable dynamic sourcing for an attribute:

   • **Use case insensitive string comparison with dynamic sourcing:** This is the default option for attributes. When dynamic sourcing is used, it allows filter qualifications to qualify on the text data of attribute forms without enforcing case sensitivity.

   • **Do not allow any string comparison with dynamic sourcing:** This option disables dynamic sourcing for attributes when a filter qualification is used to qualify on the text data of attribute forms.

7 Click Save and Close to save your changes to VLDB properties and close the VLDB Properties Editor.

8 Click Save and Close to save the attribute and close the Attribute Editor.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in Tracking the use of dynamic sourcing, page 272.

Disabling dynamic sourcing for metrics

Metrics are available for dynamic sourcing by default, but some data modeling conventions should be considered when using dynamic sourcing.
In general, if metrics use outer joins, accurate data can be returned to reports from Intelligent Cubes through dynamic sourcing. However, if metrics use inner joins, which is more common, you should verify that the metric data can be correctly represented through dynamic sourcing.

If the fact table that stores data for metrics includes NULL values for metric data, this can cause metrics that use inner joins to return incomplete data when dynamic sourcing is used. This scenario is uncommon.

If some metrics do fit this scenario, you can disable dynamic sourcing when these metrics are used in reports and Intelligent Cubes. You can enable and disable dynamic sourcing for metrics by modifying the Metric Validation VLDB property. This VLDB property has the following options:

- **Enable dynamic sourcing for metric**: This is the default option for metrics, which enables metrics for dynamic sourcing.
- **Disable dynamic sourcing for metric**: This option disables dynamic sourcing for metrics. This setting should be used if your metric data is not modeled to support dynamic sourcing. For example, the inclusion of NULLs in fact tables that contain your metric data can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing for metrics individually or you can disable dynamic sourcing for all metrics within a project. While the definition of the VLDB property at the project level defines a default for all metrics in the project, any modifications at the metric level take precedence over the project level definition. For information on accessing the VLDB Properties Editor for a project to define a default dynamic sourcing option for all metrics, see *Accessing the dynamic sourcing VLDB properties for a project, page 247.*

The procedure below describes how to disable or enable dynamic sourcing for an individual metric.

**Prerequisites**

- A metric has been created in a project.

**To enable or disable dynamic sourcing for a metric**

1. In MicroStrategy Desktop, browse to a metric, then right-click the metric and select **Edit**. The metric opens in the Metric Editor.
2 From the **Tools** menu, point to **Advanced Settings**, and then select **VLDB Properties**. The VLDB Properties Editor opens.

3 From the **Tools** menu, select the **Show Advanced Settings** option if it is not already selected.

4 In the **VLDB Settings** list, expand **Dynamic Sourcing**, and then select **Metric Validation**.

5 Clear the **Use default inherited value** check box.

6 Select one of the options depending on whether you want to disable or enable dynamic sourcing for a metric:

   • **Enable dynamic sourcing for metric**: This is the default option, which enables dynamic sourcing for the metric.

   • **Disable dynamic sourcing for metric**: This option disables dynamic sourcing for the metric. This setting should be used if your metric data is not modeled to support dynamic sourcing.

7 Click **Save and Close** to save your changes to VLDB properties and close the VLDB Properties Editor.

8 Click **Save and Close** to save the attribute and close the Attribute Editor.

You can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeded or failed for reports, as described in *Tracking the use of dynamic sourcing, page 272.*

### Disabling dynamic sourcing for aggregate tables

Reports that use aggregate tables are available for dynamic sourcing by default, but there are some data modeling conventions that should be considered when using dynamic sourcing.

In general, aggregate tables allow accurate data to be returned to reports from Intelligent Cubes through dynamic sourcing. However, if the aggregate tables use an aggregation other than Sum, or there is different data between aggregate tables and other tables in the data warehouse, this can cause aggregate tables to return incorrect data when dynamic sourcing is used. An example of an aggregate table not containing the same data is if an aggregate table includes data for years 2006, 2007, and 2008 but the lookup table for Year only includes data for 2007 and 2008.
These scenarios are uncommon.

If some aggregate tables do fit these scenarios, you can disable dynamic sourcing when these aggregate tables are used in reports and Intelligent Cubes. You can enable and disable dynamic sourcing for aggregate tables by modifying the Aggregate Table Validation VLDB property. This VLDB property has the following options:

- **Aggregate tables contain the same data as corresponding detail tables and the aggregation function is SUM**: This is the default option for aggregate tables, which enables aggregate tables for dynamic sourcing.

- **Aggregate tables contain either less data or more data than their corresponding detail tables and/or the aggregation function is not SUM**: This option disables dynamic sourcing for aggregate tables. This setting should be used if your aggregate tables are not modeled to support dynamic sourcing. The use of an aggregation function other than Sum or the mismatch of data in your aggregate tables with the rest of your data warehouse can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing individually for reports that use aggregate tables or you can disable dynamic sourcing for all reports that use aggregate tables within a project. While the definition of the VLDB property at the project level defines a default for all reports in the project, any modifications at the report level take precedence over the project level definition. For information on accessing the VLDB Properties Editor for a project to define a default dynamic sourcing option for all metrics, see [Accessing the dynamic sourcing VLDB properties for a project, page 247](#).

The procedure below describes how to disable or enable dynamic sourcing for an individual report that uses an aggregate table.

**Prerequisites**

- A report has been created in a project.

1. In MicroStrategy Desktop, browse to a report, then right-click the report and select **Edit**. The report opens in the Report Editor.
2 From the **Data** menu, select **VLDB Properties**. The VLDB Properties Editor opens.

3 From the **Tools** menu, select the **Show Advanced Settings** option if it is not already selected.

4 In the **VLDB Settings** list, expand **Dynamic Sourcing**, and then select **Aggregate Table Validation**.

5 Clear the **Use default inherited value** check box.

6 Select one of the options depending on whether you want to disable or enable dynamic sourcing for a report that uses aggregate tables:

   - **Aggregate tables contain the same data as corresponding detail tables and the aggregation function is SUM**: This is the default option for aggregate tables, which enables aggregate tables for dynamic sourcing.

   - **Aggregate tables contain either less data or more data than their corresponding detail tables and/or the aggregation function is not SUM**: This option disables dynamic sourcing for aggregate tables. This setting should be used if your aggregate tables are not modeled to support dynamic sourcing.

7 Click **Save and Close** to save your changes to VLDB properties and close the VLDB Properties Editor.

8 Click **Save and Close** to save the report and close the Report Editor.

### Using Cube Advisor to support dynamic sourcing

MicroStrategy Cube Advisor allows you to create and support a dynamic sourcing strategy that can best support the reports in your projects. Cube Advisor provides the following features for creating and supporting a dynamic sourcing strategy:

- Analyzes reports to determine if their report definitions can support the use of dynamic sourcing.
• Recommends and creates Intelligent Cubes that could provide data to reports using dynamic sourcing. This includes the following:

  □ Recommends Intelligent Cube definitions for Intelligent Cubes that could provide data to reports via dynamic sourcing. This includes providing the number of reports that could connect to the Intelligent Cube using dynamic sourcing, as well as the names and locations of these reports.

  You can also supply a MicroStrategy Enterprise Manager report to include more information on the performance benefits of each Intelligent Cube as part of your dynamic sourcing strategy. For information on locating and using this Enterprise Manager report, see *Enterprise Manager report for Cube Advisor, page 263.*

  □ Creates the Intelligent Cubes you select to create to support your dynamic sourcing strategy. These Intelligent Cubes are created in a folder along with shortcuts to all the reports that can connect to the Intelligent Cube using dynamic sourcing. For information on creating Intelligent Cubes using Cube Advisor, see *Reviewing and creating recommended Intelligent Cubes, page 266.*

**Prerequisites for using Cube Advisor**

You must complete a few prerequisites before using Cube Advisor to recommend and create Intelligent Cubes to support a dynamic sourcing strategy. These prerequisites include:

• Enable dynamic sourcing for the project and other aspects of the MicroStrategy system. For steps to configure and enable dynamic sourcing, see *Configuring dynamic sourcing, page 245.*

• Create the metric levels log file used to track the use of dynamic sourcing. For steps to create this log file, see *Creating the metric levels log, page 262.*

• Locate and execute the MicroStrategy Enterprise Manager report to provide additional performance benefit information on each Intelligent Cube recommended by Cube Advisor. This is an optional prerequisite. For information on this report, see *Enterprise Manager report for Cube Advisor, page 263.*
Creating the metric levels log

The metric levels log file is one of the files that can be used to track the use of dynamic sourcing (see *Tracking the use of dynamic sourcing, page 272*).

Cube Advisor uses this log file to help recommend and create Intelligent Cubes to support dynamic sourcing. The steps below show you how to create this log file using the MicroStrategy Diagnostics and Performance Logging tool.

If you install and use Cube Advisor on the same machine that hosts Intelligence Server, the metric levels log is created automatically. In this scenario, you do not have to perform the steps provided below.

To create the metrics level log file

1. From the Windows Start menu, point to Programs, then point to MicroStrategy, then point to Tools, and then select Diagnostics Configuration. The MicroStrategy Diagnostics and Performance Logging tool opens.

2. In the Select Configuration drop-down list, select from one of two configurations to see the current default setup, as follows:
   - **Machine Default**: The components and counters that are displayed reflect the client machine.
   - **CastorServer instance**: The components and counters that are displayed reflect server-specific features.

   When you select the CastorServer instance, you can select whether to use the default configuration. On the Diagnostics tab, this check box is named *Use Default Diagnostics Configuration*; on the Performance tab, this check box is called *Use Default Performance Configuration*. This check box refers to the Machine Default settings. No matter what you have changed and saved on either tab when CastorServer instance is selected, if you check the *Use Default Configuration* box, the system logs whatever information is configured for Machine Default at runtime.

3. On the Diagnostics tab, locate the Dynamic Sourcing component.

4. For the Metric Levels Log dispatcher, select the check box in the Console Log column.
5 For the **Metric Levels Log** dispatcher, in the **File Log** column, click the drop-down list and select *<New>*. The Log Destination Editor opens.

6 In the **File name** field, type a name for the metric levels log. In later examples, the log file is named `MetricLevelsLog`.

7 You can leave the default settings for the other options, and then click **Save**. Click **Close** to close the Log Destination Editor.

8 For the **Metric Levels Log** dispatcher, in the **File Log** column, click the drop-down list and select the metrics level log file that you created in the previous steps.

9 From the **File** menu, select **Save**. The metric levels log is created in the Log directory within the MicroStrategy common files. The default directory is `C:\Program Files\Common Files\MicroStrategy\Log`.

**Enterprise Manager report for Cube Advisor**

You can also use a MicroStrategy Enterprise Manager report to include more information on the performance benefits of each Intelligent Cube as part of your dynamic sourcing strategy. When using this Enterprise Manager report, the information that can be returned for each potential Intelligent Cube includes:

- **Jobs**: The number of jobs that would not have to be executed against the data warehouse if the Intelligent Cube is created to support your dynamic sourcing strategy.

- **Result Rows**: The number of rows of data for all reports that could be returned from the Intelligent Cube rather than the data warehouse if the Intelligent Cube is created to support your dynamic sourcing strategy.

- **Users**: The number of users that execute the reports that could connect to the Intelligent Cube using dynamic sourcing if the Intelligent Cube is created to support your dynamic sourcing strategy.

- **Database Time**: The amount of time to execute SQL statements and retrieve the results from the data warehouse that is saved if the Intelligent Cube is created to support your dynamic sourcing strategy.

This report is available with MicroStrategy Enterprise Manager. Search the Enterprise Manager project for the report named **Report Request Statistics per Project**.
To use this report with Cube Advisor, you must execute the report and export it as a Microsoft Excel file or a text file. The steps below show you how to execute and export this report in MicroStrategy Desktop or Web:

- *Executing and exporting the Enterprise Manager report in Desktop, page 264*
- *Executing and exporting the Enterprise Manager report in MicroStrategy Web, page 265*

### Executing and exporting the Enterprise Manager report in Desktop

The steps below show you how to execute and export this report in MicroStrategy Desktop.

---

**To execute and export the Enterprise Manager report in Desktop**

1. In Desktop, log in to the Enterprise Manager project.
2. Locate the Enterprise Manager report for Cube Advisor, by searching the Enterprise Manager project for the report named *Report Request Statistics per Project*.
3. Right-click the *Report Request Statistics per Project* report and select *Run*. The Report Editor opens and the report is executed.
4. From the *Data* menu, select *Export Options*. The Excel Options dialog box opens.
5. In the *Application* drop-down list, select *MS Excel Worksheet*.
6. On the *Appearance* tab, select the following export options:
   - In the *Exported fields* area, select the *Report Data* check box. All other check boxes must be cleared.
   - In the *Formatting* area, select the *Apply formatting* and *Use 6.x layout* check boxes.
7. Click *OK* to save your changes and return to the Report Editor.
8. From the *Data* menu, point to *Export To*, and then select *MS Excel*. 
9 Once the export is complete, the exported report is displayed in Microsoft Excel.

10 Save the Excel file as a tab delimited text file (.txt).

11 If you use an internationalized MicroStrategy environment that uses various character sets for your data, you must open the tab delimited text file in a text editor and save the file with a UTF-8 encoding.

**Executing and exporting the Enterprise Manager report in MicroStrategy Web**

The steps below show you how to execute and export this report in MicroStrategy Web.

---

**To execute and export the Enterprise Manager report in MicroStrategy Web**

1 In MicroStrategy Web, log in to the Enterprise Manager project.

2 Locate the Enterprise Manager report for Cube Advisor. Search the Enterprise Manager project for the report named **Report Request Statistics per Project**.

3 Click the **Report Request Statistics per Project** report to execute it.

4 From the **Home** menu, point to **Export**, and then select **Plain text**. The Export Options page opens.

5 In the **Delimiter** drop-down list, select **Tab**.

6 Clear all available check boxes.

7 In the **Remove extra column** drop-down list, select **Yes**.

8 Click **Export**. The File Download dialog box opens.

9 Click **Save**. The Save As dialog box opens.

10 Type a name and select a location for the text file, and click **Save** to save the text file.

11 If you use an internationalized MicroStrategy environment that uses various character sets for your data, you must open the tab delimited text file in a text editor and save the file with a UTF-8 encoding.
Reviewing and creating recommended Intelligent Cubes

After you have completed the prerequisites for using Cube Advisor, you can begin using Cube Advisor to create Intelligent Cubes.

The steps below show you how to use Cube Advisor to create Intelligent Cubes to support dynamic sourcing for the reports in your project.

**Prerequisites**

- You must complete the prerequisites described in *Prerequisites for using Cube Advisor, page 261* before using Cube Advisor to recommend and create Intelligent Cubes to support a dynamic sourcing strategy.

- In order to use Cube Advisor, you must log in to a project using a MicroStrategy user account with the following privileges:
  - Privileges required to execute reports and documents.
  - Privileges required to create folders in a MicroStrategy project.
  - Privileges required to create Intelligent Cubes, which is part of an OLAP Services license.

You can use Integrity Manager to perform a before and after test to identify which reports actually connect to a suggested Intelligent Cube and also to test data integrity.

---

**To review and create recommended Intelligent Cubes using Cube Advisor**

1. From the Windows Start menu, point to Programs, then point to MicroStrategy, then point to Tools, and then select Cube Advisor. Cube Advisor opens.

2. You can analyze a new set of reports using Cube Advisor or use the results of a previous Cube Advisor analysis:
   - To use Cube Advisor to analyze a new set of reports, from the Tools menu, select Options. The Options dialog box opens. Continue to step 3 in this procedure and complete all of the steps provided.
   - If you have previously used Cube Advisor to create Intelligent Cubes to support dynamic sourcing for the reports in your project, you can use the results file of this analysis. From the File menu, select Open. In the Open dialog box, select the Cube Advisor results file to use and click Open.
The results of a Cube Advisor analysis are stored in the MicroStrategy common files folder (the default is \Program Files\Common Files\MicroStrategy). The name of the file is in the format ProjectName.details.txt. For example, analyzing the MicroStrategy Tutorial project creates a Cube Advisor results file named MicroStrategy Tutorial.details.txt.

To complete the Cube Advisor analysis to review and create Intelligent Cubes, complete the steps beginning with *To review Cube Advisor analysis, page 269.*

3 Click ... (browse) to supply the following files to support Cube Advisor and define the following options:

- **Metric Level File:** The metric levels log file is one of the files that can be used to track the use of dynamic sourcing. Cube Advisor uses this log file to help recommend and create Intelligent Cubes to support dynamic sourcing.

  If you install and use Cube Advisor on the same machine that hosts Intelligence Server, the metric levels log file is automatically created by Cube Advisor. In this scenario you do not need to create the metric levels log file or provide the location for the file. Otherwise, you must create this file manually and provide the directory it is stored in.

  When manually creating the metric levels log file, the file is created in the Log directory within the MicroStrategy common files. The default directory is \Program Files\Common Files\MicroStrategy\Log. For steps on how to create this file, see *Creating the metric levels log, page 262.*

- **Enterprise Manager File:** You can also supply a MicroStrategy Enterprise Manager report to include more information on the performance benefits of each Intelligent Cube as part of your dynamic sourcing strategy. For information on locating, executing, and exporting this report as well as the information provided by supplying the report, see *Enterprise Manager report for Cube Advisor, page 263.*

- **Ignore Reports That Are Covered By Existing Cubes:** Select this option to exclude reports that already connect to an Intelligent Cube from the Cube Advisor analysis. This prevents Cube Advisor from analyzing and recommending Intelligent Cubes for reports that already connect to Intelligent Cubes.

- **Ignore Reports Not In Enterprise Manager File:** Select this option to exclude reports that are not included in the Enterprise Manager report from Cube Advisor analysis. This allows Cube Advisor to focus
only on the reports that were included in the Enterprise Manager report for further performance analysis.

4 Click OK to close the Options dialog box and return to Cube Advisor.

5 Provide the following connection information:

- **Computer Name**: Type the name of the machine on which Intelligence Server is hosted.
- **Port**: Type the port number used for Intelligence Server. The default is 34952.
- **User**: Type the MicroStrategy user’s user name to connect to a MicroStrategy project. The MicroStrategy user must have the privileges listed in the prerequisites for these steps.
- **Password**: Type the password for the MicroStrategy user.

6 Click Connect. MicroStrategy projects are displayed.

7 Select a project, and then click **Connect To Project**. The Report Selection page opens.

   If you supplied the Enterprise Manager report, you must select the project that was analyzed using the Enterprise Manager report.

8 Expand the folders of the project to locate the reports to analyze with Cube Advisor.

   Selecting a check box for a report includes the report in the analysis. Selecting a check box for a folder includes all reports in that folder and all reports in folders within the folder in the analysis.

9 Once you have selected all the reports to analyze with Cube Advisor, click **Get Cube Recommendations**. The Analyzing Reports page opens and report analysis begins.

   To analyze reports, only the report SQL is analyzed, which allows a large number of reports to be analyzed without having to execute the report SQL against the database.

   If the analysis completes with a message that no reports can use dynamic sourcing, this can be caused by various scenarios:

   - The features used in the reports prevent the reports from being able to use dynamic sourcing. To review a list of features that prevent the use of dynamic sourcing, see *Features that prevent the use of dynamic sourcing, page 241*. 
- The metric levels log file was not created properly. Review the steps to create a metrics level log file (see Creating the metric levels log, page 262) and attempt the Cube Advisor analysis again.

- Dynamic sourcing is not enabled for the project, reports, or other objects and features in the project. For information on configuring dynamic sourcing, see Configuring dynamic sourcing, page 245.

To review Cube Advisor analysis

10 Once analysis is complete, the Cube Recommendations page opens. This page allows you to review the recommended Intelligent Cubes, and select which Intelligent Cubes to create. Depending on whether you supplied an Enterprise Manager report, you can review Intelligent Cubes as described below:

- If you did not supply an Enterprise Manager report, the information on the recommended Intelligent Cubes that is displayed includes the number of reports that could connect to the Intelligent Cube using dynamic sourcing.

You can select a recommended Intelligent Cube to display the attributes and metrics that would be included in the Intelligent Cube, as well as a list of reports that could connect to the Intelligent Cube using dynamic sourcing.
• If you supplied an Enterprise Manager report, a Flash visualization provides detailed information on the recommended Intelligent Cubes, an example of which is shown below:

Some of the benefits of the Flash visualization include:

– You can click Jobs, Result Rows, Users, and DB Time to sort the Intelligent Cubes by these various usage and performance statistics. These usage and performance statistics are displayed as microcharts, which are aligned with the report coverage microcharts provided for each Intelligent Cube. For information on these usage and performance statistics, see Enterprise Manager report for Cube Advisor, page 263.

– You can select an Intelligent Cube to see how creating it affects various performance statistics.

– The area at the top of the Flash visualization provides a summary of the estimated performance improvements that would come from building the selected set of Intelligent Cubes.
You can expand an Intelligent Cube to review more information on the attributes and metrics that would be included in the Intelligent Cube, as well as a list of reports that could connect to the Intelligent Cube using dynamic sourcing.

This also includes information on how many reports that could connect to the Intelligent Cube require certain attributes and metrics. This analysis allows you to determine the benefit of including each attribute and metric in an Intelligent Cube.

Each Intelligent Cube includes a microchart, which represents the subset of reports that could connect to the Intelligent Cube using dynamic sourcing. A bar in the microchart represents that the report is covered by the Intelligent Cube and no bar means that report is not covered by the Intelligent Cube. By moving the cursor over the microchart, a tooltip is displayed that lists the various reports that could connect to the Intelligent Cube using dynamic sourcing.

Once you select a specific Intelligent Cube, the bars for all reports covered by that Intelligent Cube become blue. This color coding also helps to visually analyze the overlap of reports across Intelligent Cubes. If you see two Intelligent Cubes have a very similar report coverage distribution, then only one of the two Intelligent Cubes should be created.

Once you have reviewed the information, select the check boxes for the Intelligent Cubes to create using Cube Advisor, and then click Create Cube Design. The Browse for Folder dialog box opens.

Select a folder to create the Intelligent Cubes in. Each Intelligent Cube is created in a separate folder that is created within the folder you select. Click OK. The Cube Design Options dialog box opens.

Task Progress dialog box opens.

Select one of the following options:

- **Design Intelligent Cubes with outer join properties:** Select this option to enable outer joins on all Intelligent Cubes to be created. This ensures that all warehouse data is captured; however, this can potentially increase the size of the Intelligent Cubes.

- **Allow Dynamic Sourcing without enabling outer join properties:** Select this option to allow reports to connect to Intelligent Cubes using dynamic sourcing even when some outer join properties are not
defined. However, this can cause incorrect data to be returned in certain scenarios.

For additional information on enabling dynamic sourcing for Intelligent Cubes, see Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250.

Click OK to begin creating the Intelligent Cubes. The Creating Cubes dialog box opens.

15 Once the Intelligent Cubes are created, click OK.

Each Intelligent Cube is created in a separate folder. Each folder also contains shortcuts to all the reports that can connect to the Intelligent Cube using dynamic sourcing. This provides easy recognition of the reports that can connect to each Intelligent Cube using dynamic sourcing. You must publish the Intelligent Cubes to allow the reports to connect to the new Intelligent Cubes using dynamic sourcing. For information on how to publish Intelligent Cubes, see Publishing Intelligent Cubes, page 40.

The results of the Cube Advisor analysis are stored in the MicroStrategy common files folder (the default is C:\Program Files\Common Files\MicroStrategy). The name of the file is in the format ProjectName.details.txt. For example, analyzing the MicroStrategy Tutorial project creates a Cube Advisor results file named MicroStrategy Tutorial.details.txt. You can use this file with Cube Advisor to review and create Intelligent Cubes at another time. This scenario is described in the procedure To review and create recommended Intelligent Cubes using Cube Advisor, page 266.

**Tracking the use of dynamic sourcing**

If you enable dynamic sourcing for a project, you can track various information related to dynamic sourcing that can help determine why dynamic sourcing succeeds or fails for reports. By determining why dynamic sourcing fails for a given report, you can modify your reports, Intelligent Cubes, and other objects to allow reports in your project to utilize dynamic sourcing.

You can track the use of dynamic sourcing with the MicroStrategy Diagnostics and Performance Logging tool. For information on how to use the Diagnostics and Performance Logging tool, see the System Administration Guide.
This tool allows you to enable or disable various logs related to dynamic sourcing, which record various information about dynamic sourcing, including why dynamic sourcing cannot be used in different scenarios. The sections listed below describe the dynamic sourcing log files and error codes that can be tracked using the Diagnostics and Performance Logging tool:

- *Dynamic sourcing log files, page 273*
- *Dynamic sourcing error codes and explanations, page 280*

## Dynamic sourcing log files

In the Diagnostics and Performance Logging tool, all of the dynamic sourcing log files are available on the Diagnostics Configuration tab as part of the Dynamic Sourcing component, as shown in the image below. For information on accessing the Diagnostics and Performance Logging tool, see the [System Administration Guide](#).

With the Diagnostics and Performance Logging tool, you can enable the following dynamic sourcing log files to track information about the use of dynamic sourcing:

- *Intelligent Cube parse log, page 274*
- *Report parse log, page 275*
- *Mismatch log, page 276*
• *Extended mismatch log, page 278*
• *Metric levels log, page 279*
• *Fact levels log, page 280*

**Intelligent Cube parse log**

Information about an Intelligent Cube is included in the Intelligent Cube parse log when the Intelligent Cube is published and made available by Intelligence Server.

This log displays whether an Intelligent Cube has been defined to be available for dynamic sourcing. It also lists any metrics included in the Intelligent Cube that are disabled for dynamic sourcing, and other issues that may prevent the Intelligent Cube from being available for dynamic sourcing. For explanations of various error codes that can be included in this log, see *Dynamic sourcing error codes and explanations, page 280*.

If the entries in this log for an Intelligent Cube end with `CMI_NO_ERROR`, this means the Intelligent Cube is available to reports for dynamic sourcing. Be aware that some of the metrics in an Intelligent Cube still may not be available for dynamic sourcing. You can review this log to determine whether any metrics for the Intelligent Cube are not available for dynamic sourcing.

You can also display this log in the SQL View of Intelligent Cubes. This can help determine which reports use dynamic sourcing to connect to the Intelligent Cube, as well as why some reports cannot use dynamic sourcing to connect to the Intelligent Cube. The steps below show you how to include this log for Intelligent Cubes in a project.

---

**To display the Intelligent Cube parse log in Intelligent Cubes**

1. In Desktop, log in to a project.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. Expand the **Project definition** category, and select the **Advanced** category.
4 Within the Analytical Engine VLDB properties area, click Configure. The VLDB Properties Editor opens.

   The VLDB property defined using the steps below can also be defined for individual Intelligent Cubes. To access the VLDB Properties Editor for an Intelligent Cube, with the Intelligent Cube open, from the Data menu, select VLDB Properties.

5 From the Tools menu, ensure that the Show Advanced Settings option is selected.

6 In the VLDB Settings list, expand Dynamic Sourcing, and then select Enable Cube Parse Log in SQL View.

7 Clear the Use default inherited value check box.

8 Select Enable Cube Parse Log in SQL View.

9 Click Save and Close to save your changes and close the VLDB Properties Editor.

10 Click OK to close the Project Configuration Editor.

Report parse log

When a report is executed, information about the report is included in the report parse log.

   Reports connected to a specific Intelligent Cube do not send information to the report parse log.

This log lists any reasons why a report cannot use dynamic sourcing. For explanations of various error codes that can be included in this log, see Dynamic sourcing error codes and explanations, page 280.

If the entries in this log for a report end with CMI_NO_ERROR, this means the report can use dynamic sourcing. However, this does not necessarily mean that dynamic sourcing is used; an Intelligent Cube that meets the data requirements for the report must be available. To review information on whether the report was able to find a matching Intelligent Cube, you can review the mismatch log described in Mismatch log, page 276.

   The report parse log can increase in size quickly and thus is best suited for troubleshooting purposes.
You can also display this log in the SQL View of reports. This log helps determine whether a report can use dynamic sourcing to connect to an Intelligent Cube. The steps below show you how to include this log for reports in a project.

**To display the report parse log in reports**

1. In Desktop, log in to a project.

2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.

3. Expand the **Project definition** category, and select the **Advanced** category.

4. Within the **Analytical Engine VLDB properties** area, click **Configure**. The VLDB Properties Editor opens.

   The VLDB property defined using the steps below can also be defined for individual reports. To access the VLDB Properties Editor for a report, with the report open, from the Data menu, select **VLDB Properties**.

5. From the Tools menu, ensure that the **Show Advanced Settings** option is selected.

6. In the VLDB Settings list, expand **Dynamic Sourcing**, and then select **Enable Report Parse Log in SQL View**.

7. Clear the **Use default inherited value** check box.

8. Select **Enable Report Parse Log in SQL View**.

9. Click **Save and Close** to save your changes and close the VLDB Properties Editor.

10. Click **OK** to close the Project Configuration Editor.

**Mismatch log**

When a report that can use dynamic sourcing is executed, which can be verified with the report parse log (**Report parse log, page 275**), and Intelligent Cubes are available for dynamic sourcing, information about
whether a matching Intelligent Cube can be found for the report is included in the mismatch log.

The mismatch log can increase in size quickly and thus is best suited for troubleshooting purposes.

This log lists any reasons why a report cannot use a specific Intelligent Cube to satisfy its data requirements. For explanations of various error codes that can be included in this log, see *Dynamic sourcing error codes and explanations, page 280*.

If the entries in this log for a report and Intelligent Cube combination end with `CMI_NO_ERROR`, this means the report can use dynamic sourcing to access the Intelligent Cube.

If dynamic sourcing cannot be used because of a metric (`CMI_NO_GOOD_METRIC_FOUND`), information on why the metric prevents the use of dynamic sourcing is provided in the extended mismatch log. For information on this log, see *Extended mismatch log* below.

You can also display this log in the SQL View of reports. This log helps determine why a report that can use dynamic sourcing cannot connect to a specific Intelligent Cube. The steps below show you how to include this log for reports in a project.

---

**To display the mismatch log in reports**

1. In Desktop, log in to a project.

2. Right-click the project and select *Project Configuration*. The Project Configuration Editor opens.

3. Expand the *Project definition* category, and select the *Advanced* category.

4. Within the *Analytical Engine VLDB properties* area, click *Configure*. The VLDB Properties Editor opens.

   The VLDB property defined using the steps below can also be defined for individual reports. To access the VLDB Properties Editor for a report, with the report open, from the *Data* menu, select *VLDB Properties*.

5. From the *Tools* menu, ensure that the *Show Advanced Settings* option is selected.
6 In the VLDB Settings list, expand Dynamic Sourcing, and then select Enable Mismatch Log in SQL View.

7 Clear the Use default inherited value check box.

8 Select Enable Mismatch Log in SQL View.

9 Click Save and Close to save your changes and close the VLDB Properties Editor.

10 Click OK to close the Project Configuration Editor.

Extended mismatch log

When a report that can use dynamic sourcing is executed, which can be verified with the report parse log (Report parse log, page 275), and Intelligent Cubes are available for dynamic sourcing, information about whether a matching Intelligent Cube can be found for the report is included in the mismatch log (Mismatch log, page 276).

However, if dynamic sourcing cannot be used because of a metric (CMI_NO_GOOD_METRIC_FOUND), information on why the metric prevents the use of dynamic sourcing is provided in the extended mismatch log. This information is listed for every metric that prevents the use of dynamic sourcing.

The extended mismatch log can increase in size quickly and thus is best suited for troubleshooting purposes.

You can also display this log in the SQL View of reports. This log helps determine why a metric prevents the use of dynamic sourcing is provided in the extended mismatch log. The steps below show you how to include this log for reports in a project.

To display the extended mismatch log in reports

1 In Desktop, log in to a project.

2 Right-click the project and select Project Configuration. The Project Configuration Editor opens.

3 Expand the Project definition category, and select the Advanced category.
4 Within the **Analytical Engine VLDB properties** area, click **Configure**. The VLDB Properties Editor opens.

The VLDB property defined using the steps below can also be defined for individual reports. To access the VLDB Properties Editor for a report, with the report open, from the **Data menu**, select **VLDB Properties**.

5 From the **Tools menu**, ensure that the **Show Advanced Settings** option is selected.

6 In the **VLDB Settings list**, expand **Dynamic Sourcing**, and then select **Enable Extended Mismatch Log in SQL View**.

7 Clear the **Use default inherited value** check box.

8 Select **Enable Extended Mismatch Log in SQL View**.

9 Click **Save and Close** to save your changes and close the VLDB Properties Editor.

10 Click **OK** to close the Project Configuration Editor.

**Metric levels log**

When a report that can use dynamic sourcing is executed, which can be verified with the report parse log (**Report parse log, page 275**), information about the report is included in this log, including:

- The lowest level attributes on a report, which defines the report level
- The formulas of metrics on the report

This information can help determine what attributes and metrics need to be included in an Intelligent Cube for the report to be able to connect to it through dynamic sourcing.

This log file can also be used with MicroStrategy Cube Advisor to recommend and create Intelligent Cubes that could be connected to by reports using dynamic sourcing. For information on how Cube Advisor allows you to create and support a dynamic sourcing strategy that can best support the reports in your projects, see **Using Cube Advisor to support dynamic sourcing, page 260**.
Fact levels log

When a report that can use dynamic sourcing is executed, which can be verified with the report parse log (Report parse log, page 275), information about the report is included in this log. This log tracks the same information as the metrics levels log (Metric levels log, page 279). The only difference is that the fact levels log lists the facts used for metrics rather than the metric formulas.

Dynamic sourcing error codes and explanations

There are various reasons for why a report cannot use dynamic sourcing. To help determine why reports in your projects are not able to use dynamic sourcing, the table below provides explanations of all of the error codes that can be triggered for dynamic sourcing scenarios. These error codes are included in the dynamic sourcing log files, which are described in Dynamic sourcing log files, page 273.

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMI_AGG_METRIC_BRANCH_WITH_ZERO_CHECKING</td>
<td>Dynamic sourcing cannot be used because a metric included in the Intelligent Cube uses zero checking, which is enabled with the Zero Check VLDB property. Using zero checking on metrics in Intelligent Cubes causes the Intelligent Cubes to be unavailable for dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_AGGREGATION_ON_ATTRIBUTESRELATED_TOTRANSFORMATION_ATTRIBUTES</td>
<td>Dynamic sourcing cannot be used because a transformation metric is included in the Intelligent Cube and it cannot be aggregated to the level required on the report.</td>
</tr>
<tr>
<td>CMI_ALL_METRICS_REMOVED_FROM_CUBE</td>
<td>The Intelligent Cube included metrics, but the metrics were later removed or no longer enable dynamic sourcing. This Intelligent Cube cannot be used for dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_ANDNOT_ORORNOT_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because filter qualifications are combined with AND NOT or OR NOT logical operators.</td>
</tr>
<tr>
<td>CMI_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute form present in the report is not available in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_BAD_OR_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because two filter qualifications on different attributes or attribute elements are combined with the logical operator OR.</td>
</tr>
<tr>
<td>CMI_CG_OR_CONsolidATION</td>
<td>A custom group or consolidation is used on the report. Dynamic sourcing cannot be used for reports that include custom groups or consolidations.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>CMI_COMPLEX_METRIC_EXPRESSION_ON_CUBE</td>
<td>Dynamic sourcing cannot be used for the Intelligent Cube because a metric’s formula is too complex. Metrics that are too complex include compound metrics that are not defined as smart metrics, and simple metrics that use a formula other than a single aggregation function of a single fact. You can remove the metric or modify the metric’s formula so that dynamic sourcing can be supported for the Intelligent Cube. For information on simple and compound metrics, see the Advanced Reporting Guide.</td>
</tr>
<tr>
<td>CMI_COUNT_DISTINCT_FOUND</td>
<td>Dynamic sourcing cannot be used because the Count Distinct function is used.</td>
</tr>
<tr>
<td>CMI_CSI_OR_ELEMENT_BROWSING</td>
<td>The report cannot use dynamic sourcing because it is connected to a specific Intelligent Cube, or the user is browsing the elements of an attribute from within the report.</td>
</tr>
<tr>
<td>CMI_CUBE_TEMPLATE_MISSING_FILTER_ATTRIBUTE_FORM</td>
<td>Dynamic sourcing cannot be used because an attribute form used in the report’s filter qualifications is not available in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_CUBES_FOR_AD_HOC_DISABLED_FOR_REPORT</td>
<td>Dynamic sourcing is disabled for the report. To enable dynamic sourcing for a report, see Enabling or disabling dynamic sourcing for reports, page 248.</td>
</tr>
<tr>
<td>CMI_CUBES_FOR_AD_HOC_DISABLED_MASTERSWITCH</td>
<td>Dynamic sourcing is disabled for the project. To enable dynamic sourcing for a project, see Enabling or disabling dynamic sourcing for projects, page 245.</td>
</tr>
<tr>
<td>CMI_DB_ONLY_FUNCTION_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because a passthrough or database function such as ApplySimple is used in the filter.</td>
</tr>
<tr>
<td>CMI_DB_ONLY_FUNCTION_IN_METRIC_FORMULA</td>
<td>Dynamic sourcing cannot be used because a passthrough or database function such as ApplySimple is used in a metric’s formula.</td>
</tr>
</tbody>
</table>
| CMI_DIFFERENT_AGG_TABLE_HIT | Dynamic sourcing cannot be used because of one of the following scenarios:  
  - A metric retrieves its data from different aggregate tables for the report and Intelligent Cube.  
  - The metric uses an aggregation function other than Sum for the aggregate table.  
  - The metric uses the Sum aggregation function, but the aggregate table is disabled for dynamic sourcing.  
For information on disabling dynamic sourcing for aggregate tables, see Disabling dynamic sourcing for aggregate tables, page 258. |
<p>| CMI_DIFFERENT_DB_ROLES | Dynamic sourcing cannot be used because the report and Intelligent Cube use different database instances. |</p>
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMI_DIFFERENT_METRIC_FORMULA</td>
<td>This error signifies that, due to differing metric formulas, a metric in an Intelligent Cube is not a match for a metric on a report. All metrics on an Intelligent Cube are checked to determine if they are a match for a metric on a report. This process is done until a match is found, or no match is found and dynamic sourcing is not used (finding no matches for a metric returns the error CMI_NO_GOOD_CANDIDATE_METRICS_FOUND).</td>
</tr>
<tr>
<td>CMI_DIFFERENT_TRANSFORMATIONS_USED</td>
<td>Dynamic sourcing cannot be used because a metric on the report and a metric on the Intelligent Cube use different transformations.</td>
</tr>
<tr>
<td>CMI_DIMMETRIC_BRANCH_IN_CUBE</td>
<td>This error commonly occurs because a compound metric is included in the Intelligent Cube, which is preventing the Intelligent Cube from being available for dynamic sourcing. Only compound metrics that are defined as smart metrics can be included in Intelligent Cubes.</td>
</tr>
<tr>
<td>CMI_DISTINCT_FOUND_ON_AGGREGATION_FUNCTION</td>
<td>Dynamic sourcing cannot be used because the Distinct parameter is used in the aggregation of a metric and the level of data does not match between the report and the Intelligent Cube. To support dynamic sourcing when a Distinct parameter is used in an aggregation, the Intelligent Cube and report must contain the exact same level of data.</td>
</tr>
<tr>
<td>CMI_DISTINGUISH_DUPLICATED_ROWS_MISMATCH_FOR_CUBE_AND_REPORT</td>
<td>The report cannot connect to the Intelligent Cube using dynamic sourcing because the Distinguish Duplicated Rows VLDB property is not defined the same in the report and the Intelligent Cube. For information on this and other VLDB properties, see the Supplemental Reference System Administration.</td>
</tr>
<tr>
<td>CMI_FACT_ENTRY_LVL_ERROR</td>
<td>Dynamic sourcing cannot be used because a metric uses a fact with a fact extension or degradation. For information on fact level extensions, see the Facts chapter in the Project Design Guide.</td>
</tr>
<tr>
<td>CMI_FACT_ENTRY_LVL_GREATER_THAN_A_FILTER_LEVEL</td>
<td>Dynamic sourcing cannot be used because the fact entry level of a metric on the report and the filtering criteria of the report cannot be satisfied by the Intelligent Cube. For example, a report includes a metric that has a fact entry level of Year, and the filter includes a qualification on the Month attribute, which is at a lower logical level. If the lower level attribute used in the filter (for this example, Month) is not included in the Intelligent Cube, dynamic sourcing cannot be used. To support dynamic sourcing, the lower level attribute must be included in the Intelligent Cube.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CMI_FACT_NOT_SUPPORTED_AT_REQUIRED_CUBE_LEVEL</td>
<td>The report cannot connect to the Intelligent Cube using dynamic sourcing because a metric on the report is at a higher level than what is available on the Intelligent Cube. For example, a report requires metric data to displayed at a yearly level, but the Intelligent Cube can only display metric data at the quarterly level. Dynamic sourcing cannot be used to connect the report to this Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_FACT_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a fact is included in a filter. This can only occur if you use one of the SQL optimization options available with the SQL Global Optimization VLDB property.</td>
</tr>
<tr>
<td>CMI_FILTER_KEY_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute form is used in a conditional metric included on the report or in the report filter, and the attribute form is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_METRIC_EXPR_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a metric qualification is used in the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_METRIC_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a metric qualification is used in the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_NOT_RESTRICTIVE_ENOUGH</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing for the report because it cannot be verified that the Intelligent Cube filter on a particular metric is less restrictive than the report filter.</td>
</tr>
<tr>
<td>CMI_FILTER_PRED_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a metric qualification is used in the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_RELATIONSHIP_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a relationship qualification that cannot be supported is used in the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_TABLE_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a relationship qualification that cannot be supported is used in the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTER_UNIT_ROOT_NODE_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because a report is used as a filter of the report or Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTERING_KEY_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute form is used in a conditional metric included on the report or in the report filter, and the attribute form is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_FILTERS_CANNOT_ENSURE_ALL_DATA_NEEDED_IS_IN_CUBE</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing for the report because it cannot be verified that the Intelligent Cube filter is less restrictive than the report filter.</td>
</tr>
<tr>
<td>CMI_FULL_OUTER_JOIN_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because full outer joins are not supported or enabled in the project or by the database.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CMI_GROUP_BY_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute form used to define the level for a level metric is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_IF_NO_METRIC_REPORT_AND_CUBE_LOWEST_LVL_MUST_BE_THE_SAME</td>
<td>Dynamic sourcing cannot be used for the report because the report contains no metrics, and the Intelligent Cube does not use the same level as the report. For reports with no metrics, Intelligent Cubes must have the same level as these reports to use dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_INCOMPATIBLE_DATA_TYPES_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because a filter qualification includes two incompatible data types. One scenario in which this can occur is when identity forms are used to filter data, and smaller identity numbers use an integer type, while larger identity numbers use big decimal.</td>
</tr>
<tr>
<td>CMI_INCOMPLETE_HIERARCHY_DANGEROUS_ATTRIBUTE</td>
<td>The report or Intelligent Cube includes an attribute that is defined as possibly containing null values, which disables dynamic sourcing for the report or Intelligent Cube. For information on how to disable or enable dynamic sourcing for an attribute, see Disabling dynamic sourcing for attributes, page 252.</td>
</tr>
<tr>
<td>CMI_INVALID_BETWEEN_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because an invalid filter qualification that uses the Between operator is used. The first value must be less than or equal to the second value.</td>
</tr>
<tr>
<td>CMI_JOINT_PARENT_OR_JOINT_CHILD_OR_MANY_TO_MANY_NOT_SUPPORTED</td>
<td>Joint child relationships and many-to-many relationships are not supported for dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_METRIC_AND_TEMPLATE_LVL_NOT_MATCHING_IN_CUBE</td>
<td>The metric in an Intelligent Cube cannot be used for dynamic sourcing because it is a level metric. For information on level metrics, see the Advanced Reporting Guide.</td>
</tr>
<tr>
<td>CMI_METRIC_BREAK_BY_NOT_ON_CUBE_TEMPLATE</td>
<td>Dynamic sourcing cannot be used because the break by parameter for a metric on the Intelligent Cube is not included in the Intelligent Cube. The break by parameter determines when calculations such as running summations or moving averages restart their calculations. For example, a running sum of revenue uses a break by parameter of the Year attribute. Without the Year attribute on the Intelligent Cube, the metric cannot be calculated correctly.</td>
</tr>
<tr>
<td>CMI_METRIC_CROSS_JOINED_NO_GROUP_BY</td>
<td>Dynamic sourcing cannot be used because a cross join is used for a metric.</td>
</tr>
<tr>
<td>CMI_METRIC_FILTER_KEY_NOT_IN_TEMPLATE</td>
<td>Dynamic sourcing cannot be used because an attribute form is used in a conditional metric included on the report or the report filter, and the attribute form is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CMI_METRIC_GROUP_BY_KEY_NOT_IN_TEMPLATE</td>
<td>Dynamic sourcing cannot be used because an attribute form used to define the level for a level metric is not included in the Intelligent Cube. This can also be caused by defining the filtering of a target for a level metric to be absolute or ignore.</td>
</tr>
<tr>
<td>CMI_METRIC_NOT_OJ</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing because it includes a metric that does not use outer joins. Intelligent Cubes of this type can be made available for dynamic sourcing by using outer joins for all metrics in the Intelligent Cube, or selecting the Allow dynamic sourcing even if outer join properties are not set option for the Intelligent Cube (see Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250).</td>
</tr>
<tr>
<td>CMI_METRIC_NOT_SAFE</td>
<td>The metric has been disabled for dynamic sourcing, as described in Disabling dynamic sourcing for metrics, page 256.</td>
</tr>
<tr>
<td>CMI_METRIC_WITH_CONDITIONALITY_IN_CUBE</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing because it includes a conditional metric. For information on conditional metrics, see the Advanced Reporting Guide.</td>
</tr>
<tr>
<td>CMI_METRICS_WITH_NESTED_AGGREGATION_NOT_SUPPORTED</td>
<td>Dynamic sourcing cannot be used because nested aggregation is used for a metric. Nested aggregation is not supported for dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_NO_ERROR</td>
<td>No error occurred, the report can use dynamic sourcing if an Intelligent Cube with the correct data requirements can be found.</td>
</tr>
<tr>
<td>CMI_NO_GOOD_CANDIDATE_METRICS_FOUND</td>
<td>Dynamic sourcing cannot be used because no suitable metric can be found in the Intelligent Cube for a metric included in the report.</td>
</tr>
<tr>
<td>CMI_NON_AGG_FUNCTION_FOUND</td>
<td>Dynamic sourcing cannot be used because a non-aggregation function such as First is used, and the metric level in the report is different than the metric level in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_NON_AGG_NOT_REPORT_LEVEL_METRIC_FOUND</td>
<td>Dynamic sourcing cannot be used because a metric uses a function that cannot use dynamic aggregation by default. This prevents the metric from being able to display data at the level defined for the Intelligent Cube. For information on functions that are not dynamically aggregated by default, see Metrics that are not dynamically aggregated by default, page 152.</td>
</tr>
<tr>
<td>CMI_NON_HDA_USER_CANNOT_USE_HDA_CUBE</td>
<td>Dynamic sourcing cannot be used because the Intelligent Cube uses MultiSource Option to connect to multiple data sources, but the user running the report is not licensed to use MultiSource Option. For information on MultiSource Option, see the Project Design Guide.</td>
</tr>
<tr>
<td>CMI_NONAGGREGATABLE_METRIC</td>
<td>Dynamic sourcing cannot be used because a metric that cannot be aggregated is used, and the metric level in the report is different than the metric level in the Intelligent Cube.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>CMI_NONAGGREGATION_IN_METRIC</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing because it includes a metric that cannot be aggregated, and the metric level in the report is different than the metric level in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_NULL_CHECKING_FOR_ANAL_ENGINE_MISMATCH</td>
<td>Dynamic sourcing cannot be used because the option selected for the Null checking for Analytical Engine VLDB Property is different for the report and Intelligent Cube. The same option must be used to match a report and Intelligent Cube for dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_OLD OLAP_FUNCTION_IN_METRIC</td>
<td>Dynamic sourcing cannot be used because old OLAP Services behavior is enabled. To use the new OLAP Services behavior, select the Use OLAP's new behavior option for the OLAP New Behavior VLDB Property.</td>
</tr>
<tr>
<td>CMI_PERFORM_OR_FAILED</td>
<td>Dynamic sourcing cannot be used because filtering qualifications could not be combined correctly with the logical operator OR. To support dynamic sourcing you can remove or modify filter qualifications. For information on how filter qualifications can be used with dynamic sourcing, see Best practices for supporting dynamic sourcing, page 238.</td>
</tr>
</tbody>
</table>
| CMI_REPORT_IS_XDA | The report is either a:  
  • OLAP cube report connected to SAP BI, Microsoft Analysis Services, or Hyperion Essbase  
  • Query Builder report  
  • Freeform SQL report  
These types of reports cannot use dynamic sourcing. |
<p>| CMI_REPORT_LIMIT_ON_CUBE_FOUND | A report limit is present on the Intelligent Cube. Intelligent Cubes with report limits cannot be used for dynamic sourcing. |
| CMI_REPORT_OLAP_METRIC_BREAK_BY_MISSING_FROM_CUBE_TEMPLATE | Dynamic sourcing cannot be used because the break by parameter for a metric on a report is not included in the Intelligent Cube. The break by parameter determines when calculations such as running summations or moving averages restart their calculations. For example, a running sum of revenue uses a break by parameter of the Year attribute. Without the Year attribute on the Intelligent Cube, the metric cannot be calculated correctly. |
| CMI_REPORT_OLAP_METRIC_SORT_BY_MISSING_FROM_CUBE_TEMPLATE | Dynamic sourcing cannot be used because the sort by parameter for a metric is not included in the Intelligent Cube. The sort by parameter determines the order in which calculations such as running summations or moving averages perform their calculations. For example, a running sum of revenue uses a sort by parameter of the Quarter attribute. Without the Quarter attribute on the Intelligent Cube, the metric cannot be calculated correctly. |</p>
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMI_REPORT_TEMPLATE_ATTRIBUTE_MISSING_FROM_CUBE_TEMPLATE</td>
<td>Dynamic sourcing cannot be used because an attribute included in the report is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_REPORTS_WITH_ZERO_RL_METRICS_CAN_ONLY_HIT_CUBE_WITH_FOJ_SUPPORT</td>
<td>Dynamic sourcing cannot be used because the report uses metrics that are not at the report level. To support dynamic sourcing for these types of reports, Intelligent Cubes must support outer joins. For information on outer join support with Intelligent Cubes, see Enabling or disabling dynamic sourcing for Intelligent Cubes, page 250.</td>
</tr>
<tr>
<td>CMI_SECURITY_FILTER_KEY_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute used to define the security filter of the user executing the report is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_SECURITY_FILTER_LEVEL_ATTRIBUTE_UNAVAILABLE</td>
<td>Dynamic sourcing cannot be used because an attribute used to define the security filter of the user executing the report is not included in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_STRING_COMPARISON_NOT_ALLOWED</td>
<td>Dynamic sourcing cannot be used because the report uses a filter qualification that qualifies on the text data of an attribute form, and these types of qualifications have been disabled for dynamic sourcing. For information on the String Comparison Behavior VLDB property which defines this functionality, see Supporting filtering on attributes for dynamic sourcing, page 254.</td>
</tr>
<tr>
<td>CMI_STRING_COMPARISON_NOT_ALLOWED_IN_SORT_BY</td>
<td>Dynamic sourcing cannot be used because the report uses a sort on the text data of an attribute form, and these types of sorts have been disabled for dynamic sourcing. For information on the String Comparison Behavior VLDB property which defines this functionality, see Supporting filtering on attributes for dynamic sourcing, page 254.</td>
</tr>
<tr>
<td>CMI_UNEXPECTED_REPORT_TYPE</td>
<td>The report is of a type that cannot use dynamic sourcing, which includes: • MDX cube report • Query Builder report • Freeform SQL report • Data marts</td>
</tr>
<tr>
<td>CMI_UNSATISFIABLE_CONDITION_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because an unsupported condition is included in the filter. For a list of features that are not supported for dynamic sourcing, see Features that prevent the use of dynamic sourcing, page 241.</td>
</tr>
<tr>
<td>CMI_UNSUPPORTED_ELEMENT_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because an element that is not supported for dynamic sourcing is included in the filter. For a list of features that are not supported for dynamic sourcing, see Features that prevent the use of dynamic sourcing, page 241.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CMI_UNSUPPORTED_EXPR_TYPE_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because an unsupported expression is included in a filter. For a list of features that are not supported for dynamic sourcing, see <em>Features that prevent the use of dynamic sourcing, page 241</em>.</td>
</tr>
<tr>
<td>CMI_UNSUPPORTED_FILTER_NODE</td>
<td>Dynamic sourcing cannot be used because an unsupported filter type is used. For a list of features that are not supported for dynamic sourcing, see <em>Features that prevent the use of dynamic sourcing, page 241</em>.</td>
</tr>
<tr>
<td>CMI_UNSUPPORTED_FUNCTION_IN_FILTER</td>
<td>Dynamic sourcing cannot be used because an unsupported function is included in a filter. Supported functions in filters are listed in <em>Features that can be used with dynamic sourcing, page 240</em>.</td>
</tr>
<tr>
<td>CMI_USER_LOCALE_NOT_SUPPORTED_IN_CUBE</td>
<td>The report cannot connect to the Intelligent Cube using dynamic sourcing because the Intelligent Cube is not available for the locale of the user viewing the report. For MicroStrategy projects that support multiple languages and character sets, users can view reports that display data for their locale. To use dynamic sourcing, an Intelligent Cube must be available for the user’s locale.</td>
</tr>
<tr>
<td>CMI_USER_DOESNT_HAVE_DYNAMIC_SOURCING_PRIVILEGE</td>
<td>The report cannot connect to the Intelligent Cube using dynamic sourcing because the user account being used does not have the Use Dynamic Sourcing privilege. This privilege is required to use dynamic sourcing.</td>
</tr>
<tr>
<td>CMI_VLDB_SETTING_MISMATCH_FOR_CUBE_AND_REPORT</td>
<td>The report cannot connect to the Intelligent Cube using dynamic sourcing because the options used for various VLDB properties do not match between these two objects. For a list of VLDB properties that prevent reports from using dynamic sourcing, see <em>Features that prevent reports from using dynamic sourcing, page 242</em>. For information on VLDB properties, see the <em>Supplemental Reference for System Administration</em>.</td>
</tr>
<tr>
<td>CMI_ZERO_METRICS_NEEDED</td>
<td>The report cannot use dynamic sourcing because it does not contain any metrics, and no matching Intelligent Cube with no metrics can be found.</td>
</tr>
<tr>
<td>CMI_UNSUPPORTED_INCREMENTAL_REFRESH</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing due to an unsupported incremental refresh that modifies the data in the Intelligent Cube.</td>
</tr>
<tr>
<td>CMI_RAGGED_HIERARCHY_NO_METRICS_NOT_SUPPORTED</td>
<td>The report cannot be used for dynamic sourcing because it has no metrics, and attributes from ragged hierarchies. For more information, see the <em>MDX Cube Reporting Guide</em>.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CMI_NONAGG_BY_LKP_ON_INCR_REFR_CUBE_NOT_SUPPORTED</td>
<td>The Intelligent Cube cannot be used for dynamic sourcing because it uses non-aggregation by lookup and is incrementally refreshed</td>
</tr>
<tr>
<td>CMI_DIFFERENT_MDX_SOURCE_TABLE_USED</td>
<td>The MDX-based Intelligent Cube cannot be used for dynamic sourcing because the report and Intelligent Cube point to different MDX data sources.</td>
</tr>
</tbody>
</table>
attribute  A data level defined by the system architect and associated with one or more columns in a data warehouse lookup table. Attributes include data classifications like Region, Order, Customer, Age, Item, City, and Year. They provide a means for aggregating and filtering data at a given level. See also: attribute element, attribute form.

attribute element  A value of any of the attribute forms of an attribute. For example, New York and Dallas are elements of the attribute City; January, February, and March are elements of the attribute Month.

attribute form  One of several columns associated with an attribute that are different aspects of the same thing. For example, ID, Name, Last Name, Long Description, and Abbreviation could be forms of the attribute Customer. Every attribute supports its own collection of forms.

cache  A special data store holding information that was recently accessed from the data source, used for quick future access. Caching is normally done for frequently requested reports, whose execution is faster because they need not run against the data source. Cached results from the data source are stored separately and can be used by new job requests that require the same data.
**compound metric**  A metric that cannot have a level placed on the entire metric, although it can be set separately on each of the components.

**drilling**  1) A method of obtaining supplementary information after a report has been executed. The new data is retrieved by re-querying the Intelligent Cube or database at a different attribute or fact level. 2) A link from one document to another document, report, or HTML document. Prompt answers for the destination can be included in the drill.

**dynamic aggregation**  An OLAP Services feature. Roll-up of metric values that occurs when an attribute is moved from the report grid to the Report Objects pane. Whenever the attributes in the Report Objects pane are not the same as the attributes on the grid, dynamic aggregation has occurred. Dynamic aggregation happens on-the-fly, in memory.

**dynamic sourcing**  Extends the accessibility of Intelligent Cubes by allowing standard reports to access any published Intelligent Cubes that can satisfy the data requirements of the report. This connection is performed automatically without the user having to explicitly connect their report to a published Intelligent Cube. This allows users performing ROLAP analysis to work completely within a pre-created set of data without having to manually find and select the Intelligent Cube that fits their analysis.

**fact**  1) A measurement value, often numeric and typically aggregatable, stored in a data warehouse. 2) A schema object representing a column in a data warehouse table and containing basic or aggregated numbers—usually prices, or sales in dollars, or inventory quantities in counts. See also metric.
**filter** (also called a report filter) A MicroStrategy object that specifies the conditions that the data must meet to be included in the report results. Using a filter on a report narrows the data to consider only the information that is relevant to answer your business question, since a report queries the database against all the data stored in the data warehouse.

A filter is composed of at least one qualification, which is the actual condition that must be met for the data to be included on a report. Multiple qualifications in a single filter are combined using logical operators. Examples include "Region = Northeast" or "Revenue > $1 million".

A filter is normally implemented in the SQL WHERE clause.

**Grid/Graph** A control placed in a document that displays information in the same way a MicroStrategy report does.

**Intelligent Cube** A set of data that can be shared as a single in-memory copy, among many different reports created by multiple users. Rather than returning data from the data warehouse for a single report, Intelligent Cubes return sets of data from your data warehouse and save them directly to Intelligence Server memory. The reports accessing Intelligent Cubes can use all of the OLAP Services features for analysis and reporting purposes.

Intelligent Cubes provide access to a single set of data that can be shared among multiple reports and analyzed with OLAP Services, while reducing access to the data warehouse.

**level** 1) In a data warehouse, facts are said to be stored at a particular level defined by the attribute IDs present in the fact table. For example, if a fact table has a Date column, an Item_ID column, and a fact column, that fact is stored at the Date/Item level. 2) In a metric calculation, the level is the granularity of where an attribute appears in its hierarchy, where that attribute defines how a related metric is calculated. For example, a metric on a report with Year and Store attributes would be calculated at the Year/Store level.
**metric**  
1) A business calculation defined by an expression built with functions, facts, attributes, or other metrics. For example:  
\[ \text{sum(dollar\_sales)} \text{ or } \text{[Sales]} - \text{[Cost]} \]  
2) The MicroStrategy object that contains the metric definition. See also fact.

**OLAP Services**  
(OLAP stands for online analytical processing) A MicroStrategy product, which is an extension of Intelligence Server. OLAP Services uses Intelligent Cube Technology, an in-memory version of report data that can be manipulated by the MicroStrategy Analytical Engine. MicroStrategy Desktop, Web, and Office users can slice and dice data in reports within Intelligent Cubes without having to re-execute SQL against the data warehouse.

**page-by**  
Segmenting the data in a grid report by placing available attributes, consolidations, and metrics on a third axis called the Page axis. Since a grid is two-dimensional, only a slice of the cube can be seen at any one time. The slice is characterized by the choice of elements on the Page axis. By varying the selection of elements, the user can page through the cube.

**personal Intelligent Cubes**  
The Intelligent Cube functionality available in all pre-9.0 MicroStrategy releases. When a report returns a new set of data, Intelligence Server creates a cached copy of the report data. The view of the report can be manipulated with OLAP Services features, such as derived metrics, view filters, and so on. These manipulations take advantage of MicroStrategy Intelligent Cube Technology by executing against the cached copy of the report data, rather than requiring execution against the data warehouse. For reports that use Intelligent Cube Technology, the cached copy of report data that the report executes against is referred to as a personal Intelligent Cube.

Personal Intelligent Cubes allow a single report to make use of OLAP Services while still being able to seamlessly access the data warehouse.
**prompt**  MicroStrategy object in the report definition that is incomplete by design. The user is asked during the resolution phase of report execution to provide an answer that completes the query. A typical example is choosing a specific attribute on which to qualify the report results.

**report**  The central focus of any decision support investigation. A report allows users to query for data, analyze that data, and then present it in a visually pleasing manner.

**report limit**  A MicroStrategy feature that restrict the final result set of a report by setting criteria on the report metrics. You can define report limits on the value, rank, or percent of the metric value returned.

**smart metric**  A property of a compound metric that allows you to change the default evaluation order. Smart metrics calculate subtotals on the individual elements of the compound metric. For example, a smart metric uses the formula Sum(Metric1)/Sum(Metric2) rather than Sum(Metric1/Metric2) when calculating subtotals on a report.

**threshold**  Used to create conditional formatting for metric values. For example, if dollar sales is greater than $200, format that cell to have a blue background with bold type.
INDEX

A
accessing
   Derived Elements Editor 100
   Intelligent Cube with a report 55
aggregate table in dynamic sourcing 239, 258
aliasing 10
All Other derived element 79, 127
attribute
   compound, in dynamic sourcing 240
   disabling dynamic sourcing 252
   in a derived metric 167
   in dynamic sourcing 239
   inner join in dynamic sourcing 252
   outer join in dynamic sourcing 252
attribute element
   derived element, not included in 79
   in dynamic sourcing 239, 252
   report, displaying names in 125
attribute form qualification
   in a view filter 198
   in dynamic sourcing 255
attribute qualification in a view filter 195
average in a derived element 87

B
banding 11
   in dynamic sourcing 241
best practices
   derived metric creation 167
   dynamic sourcing 238
break by
   in a rank shortcut metric 179
   in a report limit 193

C
cache 16
   Intelligent Cube 39
Calculation derived element 77
display 126
   example 78
comparing
   metric values over time 202
   text strings 200
   values 199
comparison operator 199
compound attribute in dynamic sourcing 240
compound metric
dynamic aggregation and 158
example 158
connecting a report to an Intelligent Cube 55
connection mapping 25
consolidation alternative to 28
in an Intelligent Cube 28
in dynamic sourcing 242
creating
Calculation derived element 112
derived metric 167
Filter derived element 108
Group derived element 104
Intelligent Cube from an existing report 31
Intelligent Cube from scratch 29
percent range view filter 212
percent-to-total shortcut metric 174
quick calculation derived element 86
quick group using a derived element 83
rank range view filter 210
rank shortcut metric 177
report to use an Intelligent Cube 55
shortcut metric 173
smart metric 160
stand-alone derived element 117
view filter 194
Cube Advisor. See MicroStrategy Cube Advisor.
custom group
alternatives to 28
in dynamic sourcing 242
in Intelligent Cube 28
data warehouse
forcing a report to execute against 155
naming conventions 170
deleting
derived metric 181
view filter 219
derived element defined on 69
All Other derived element 79, 127
attribute element names,
displaying 125, 127, 128
calculating average in 87
calculating with 111
Calculation derived element 112
character format 133
consolidation and custom group 28
Derived Elements Editor 97
derived metric effect on 183
evaluation order 142
example 103, 107, 111
Filter derived element 108
formatting 130
Group derived element 104
in Intelligent Cube 28
quick calculation 86
quick group 83
quick sort 93
stand-alone 118
subtotals 122
value format 133
view filter effect on 232
Derived Elements Editor 97
accessing 100
creating a derived element 108
example 98
formatting with 130
derived metric defined on 165, 221
as an object 166
attribute in 167
data mart report in dynamic sourcing 243
best practices 167
creating 167
creating a shortcut metric 173
creating with the Input Metric Formula dialog box 169
deleting 181
derived element effect on 135, 183
editing 180
example 165, 171
formatting 181
function 168
in dynamic sourcing 241
in Intelligent Cube 28
level 168
operator 168
rank shortcut metric 177
removing from a report layout 181
threshold 181
troubleshooting 170
view filter effect on 220
Diagnostics and Performance Logging Tool 272
disabling
drilling in an Intelligent Cube 34
dynamic sourcing 245
displaying attribute element names in a report 125
division in a derived element 87
drilling
derived element interaction with 139
example for a report connected to an Intelligent Cube 60
in a report connected to an Intelligent Cube 33, 60
dynamic aggregation defined on 145
compound metric in 158
data warehouse 155
data warehouse access 155
estimating values 156
example 146, 153, 155
function 149
function, editing the 161
functions that are not supported by 152
functions that are supported by 150
in dynamic sourcing 241
null values 154
troubleshooting 158
view filter effect on 163, 228
dynamic sourcing defined on 39, 235, 272
aggregate table in 239, 258
attribute element in 239
attribute in 239
banding in 241
best practices 238
compound attribute in 240
consolidation in 242
custom group in 242
data mart report in 243
derived metric in 241
disabling 245
dynamic aggregation in 241
enabling 245
filter in 238, 241, 242, 243, 255
Freeform SQL report in 242
functions in 239, 240, 243
inner join in 252, 257
Intelligent Cube tuning 237
joint element list in 240
MDX cube report in 242
metric in 242, 244, 256, 279
MicroStrategy Cube Advisor 260
null values in 239, 253
operator in 242
outer join in 252, 257
outline mode in 241
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>page-by interaction with a derived element</td>
<td>136</td>
</tr>
<tr>
<td>percent range view filter</td>
<td>211</td>
</tr>
<tr>
<td>percent-to-total shortcut metric</td>
<td>174</td>
</tr>
<tr>
<td>rank range view filter</td>
<td>208</td>
</tr>
<tr>
<td>rank shortcut metric</td>
<td>177</td>
</tr>
<tr>
<td>sorting using a derived element</td>
<td>93</td>
</tr>
<tr>
<td>subtotals with a derived element</td>
<td>122</td>
</tr>
<tr>
<td>threshold interaction with a derived element</td>
<td>138</td>
</tr>
<tr>
<td>view filter effect on derived element</td>
<td>232</td>
</tr>
<tr>
<td>view filter on a metric</td>
<td>200, 203, 230</td>
</tr>
<tr>
<td>view filter on an attribute</td>
<td>195, 198, 229</td>
</tr>
<tr>
<td>view filter vs. report filter</td>
<td>191</td>
</tr>
<tr>
<td>view filter, basic</td>
<td>186</td>
</tr>
<tr>
<td>expression</td>
<td></td>
</tr>
<tr>
<td>Calculation derived element</td>
<td>78, 113</td>
</tr>
</tbody>
</table>

**E**

- **editing**
  - derived metric | 180 |
  - dynamic aggregation function | 161 |
  - evaluation order of a derived element | 143 |
  - evaluation order of a smart metric | 143 |
  - null values | 160 |
  - rank order in a rank shortcut metric | 179 |
  - stand-alone derived element | 121 |
- **enabling**
  - drilling in an Intelligent Cube | 34 |
  - dynamic sourcing | 245 |
- **error message** | 65 |
- **evaluation order**
  - derived element | 142 |
  - for view filters and derived metrics | 216 |
  - smart metric | 142 |
- **example** | |

Calculation derived element | 78, 111 |
compound metric in dynamic aggregation | 158 |
Derived Elements Editor | 98 |
derived metric | 165, 171 |
displaying a derived element as a single item | 125, 127 |
drilling in a report connected to an Intelligent Cube | 60 |
drilling interaction with a derived element | 140 |
dynamic aggregation | 146, 153, 155 |
evaluation order for view filters and derived metrics | 214 |
evaluation order with a derived element | 142 |
evaluation order with a smart metric | 142 |
Filter derived element | 75, 107 |
Group derived element | 72, 103 |
OLAP Services | 8 |
page-by interaction with a derived element | 136 |
percent range view filter | 211 |
percent-to-total shortcut metric | 174 |
rank range view filter | 208 |
rank shortcut metric | 177 |
sorting using a derived element | 93 |
subtotals with a derived element | 122 |
threshold interaction with a derived element | 138 |
view filter effect on derived element | 232 |
view filter on a metric | 200, 203, 230 |
view filter on an attribute | 195, 198, 229 |
view filter vs. report filter | 191 |
view filter, basic | 186 |
expression | |
derived metric 169

defining a filter 48
defining a report 50
filter 47, 48
prerequisites 45
report 47, 50
republish settings 46

inner join
for a metric in dynamic sourcing 257
for an attribute in dynamic sourcing 252

Freeform SQL report in dynamic sourcing 242

Input Metric Formula dialog box 169

Intelligent Cube defined on 13
accessing with a report 55
caching 39
changing for a report 57
connecting to a report 55
consolidation in 28
creating a report that uses 56
creating from an existing report 31
creating from scratch 29
creating using Cube Advisor 266
custom group in 28
derived element in 28
derived metric in 28
disabling dynamic sourcing 250
drilling 33
drilling in a report connected to 60
dynamic sourcing, tuning for 237
enabling dynamic sourcing 250
failure of the report connection 65
filter in 30
incremental refresh 45
localizing 36
memory resources 40, 42
personal 13, 17
personal Intelligent Cube compared to 14
prompt in 28

F

defined metric 169

defining a filter 48
defining a report 50
filter 47, 48
prerequisites 45
report 47, 50
republish settings 46

inner join
for a metric in dynamic sourcing 257
for an attribute in dynamic sourcing 252

Freeform SQL report in dynamic sourcing 242

Input Metric Formula dialog box 169

Intelligent Cube defined on 13
accessing with a report 55
caching 39
changing for a report 57
connecting to a report 55
consolidation in 28
creating a report that uses 56
creating from an existing report 31
creating from scratch 29
creating using Cube Advisor 266
custom group in 28
derived element in 28
derived metric in 28
disabling dynamic sourcing 250
drilling 33
drilling in a report connected to 60
dynamic sourcing, tuning for 237
enabling dynamic sourcing 250
failure of the report connection 65
filter in 30
incremental refresh 45
localizing 36
memory resources 40, 42
personal 13, 17
personal Intelligent Cube compared to 14
prompt in 28

G

greatest function (derived element) 87

Greatest function (derived element) 87

Grid/Graph defined on 81

Group derived element 72
example 72

I

incremental refresh for an Intelligent Cube 45
defining 45
publishing manually 40  
publishing on a schedule 41  
scheduling an update 32  
security 21  
shared 13  
troubleshooting 65  
unpublishing 44  
view filter in 28, 190  
Intelligent Cube Monitor 43  
international support xxii  

J  
joint element list in dynamic sourcing 240  

L  
least function (derived element) 88  
level  
  derived metric 168  
  percent-to-total shortcut metric 176  
  view filter 213  
localizing an Intelligent Cube 36  
log file for dynamic sourcing 273  

M  
MDX cube report in dynamic sourcing 242  
memory resources and Intelligent Cubes 40, 42  
metric  
  dynamic aggregation functionality 149  
  in dynamic sourcing 256, 279  
  inner join in dynamic sourcing 257  
  metric to metric qualification 202  
  outer join in dynamic sourcing 257  
  shortcut metric 173  
  smart metric 160  
  values, comparing over time 203  
  view filter effect on 226  
metric qualification in a view filter 200  
MicroStrategy Cube Advisor 260  
creating an Intelligent Cube 266  
prerequisites 261  

N  
null values  
  dynamic aggregation 154  
  dynamic sourcing, in 252  
  editing 160  
  in a view filter 201, 207  
  in dynamic sourcing 239, 253, 257  
  troubleshooting 252, 257  

O  
OLAP Services defined on 1  
  benefits 5  
  example 8  
  license 2  
  standard features 10  
operator  
  comparison 199  
  in a derived metric 168  
  in a Filter derived element 79  
  in a joined view filter qualification 218  
  in a metric qualification for a view filter 200  
  in a percent metric view filter 206  
  in a rank metric view filter 206  
  in an attribute form qualification for a view filter 199  
  in dynamic sourcing 242, 259  
  pattern 200  
  order in a rank shortcut metric 179  
  outer join for dynamic sourcing 252, 257
outline mode 11
  in dynamic sourcing 241

P
page-by 11, defined on 136
  derived element interaction with 136
    example 136
  in dynamic sourcing 241
  percent-to-total metric 176
pattern operator 200
percent range view filter 205
  creating 212
    example 211
percent-to-total shortcut metric
  creating 174
    example 174
  level 175
  page-by 176
  rules 175
personal Intelligent Cube defined on 13
  compared to an Intelligent Cube 14
  privileges 19
  prompt and 59
  security 19
pivoting 12
  in dynamic sourcing 241
privileges
  in a personal Intelligent Cube 19
  OLAP Services 7
project
  disabling for dynamic sourcing 246
  enabling for dynamic sourcing 246
  tuning for dynamic sourcing 237
prompt
  in a report accessing a personal Intelligent Cube 59
  in a report accessing an Intelligent Cube 58
in a report limit 193
  in dynamic sourcing 240
  in Intelligent Cube 28
publishing an Intelligent Cube 40
  manually 40
  on a schedule 41
Q
Query Builder report in dynamic sourcing 242
  quick calculation using a derived element 86
  quick group using a derived element 83
  quick sort using a derived element 93
R
rank range view filter 205
  creating 210
    example 208
rank shortcut metric
  break by 179
  creating 177
  example 177
  order 179
  troubleshooting 178
report
  accessing an Intelligent Cube 55
  changing the Intelligent Cube 57
  connecting to an Intelligent Cube 55
  connection failure to Intelligent Cube 65
  converting to an Intelligent Cube 31
  creating to use an Intelligent Cube 55
  disabling for dynamic sourcing 249
  drilling in an Intelligent Cube 33
  dynamic sourcing 238
  enabling for dynamic sourcing 249
Index

in dynamic sourcing 275
report limit
  in dynamic sourcing 243
view filter compared to 192

S
scheduling
  Intelligent Cube publication 41
  Intelligent Cube update 32
security
  connection mapping and 25
  Intelligent Cube 21
permission groups for Intelligent Cubes 25
personal Intelligent Cube 19
security filter
  in dynamic sourcing 240
  Intelligent Cube 22
shortcut metric defined on 173
  creating 173
  percent-to-total 174
  rank 173
  transformation 173
smart metric defined on 142
  creating 160
derived element interaction 142
sorting 12
  attribute elements using a derived element 93
  in dynamic sourcing 241
stand-alone derived element 117
  adding to a report or Grid/Graph 119
editing 121
Subscription Manager 43
subtotals 12
  derived element value, applying to 122
  example 123
  in dynamic sourcing 241
subtraction in a derived element 87
sum in a derived element 87
support
  international xxii
support. See technical support.

T
technical support xxii
text string comparison 200
threshold 12, 138
  derived element effect on 138
  in a derived metric 181
  in dynamic sourcing 241
transformation in dynamic sourcing 240
transformation shortcut metric 173
troubleshooting 272
  derived metric 170
dynamic aggregation 158
dynamic sourcing 238, 241, 252
Intelligent Cube 65
null values 160, 252, 257
  rank shortcut metric order 178

U
unpublishing an Intelligent Cube 44
updating Intelligent Cube against the data source 32

V
value comparison 199
values, null 154
view filter defined on 4
  attribute form qualification in 198
  attribute qualification in 195
  creating with a metric qualification 200
  creating with an attribute
qualification 197
deleting 219
derived element effect on 135, 232
derived metric effect on 182, 220
design considerations 190, 194
dynamic aggregation effect on 163, 228

element
  attribute qualification 195
  attribute qualification on a hidden attribute 229
  basic view filter 186
  business scenario 187
  metric qualification 200, 203, 230
  view filter vs. report filter 191
in dynamic sourcing 241
in Intelligent Cube 28
Intelligent Cube interaction with 190
joining qualification 218
metric effect on 226
operator 206, 218
percent metric qualification 206
rank metric qualification 205
report filter compared to 188
report limit compared to 192
VLDB properties in dynamic sourcing 247, 253, 255