

IBM® DB2® OLAP Server™ and DB2 OLAP Starter Kit



Version 7

*OLAP Integration Server Metaoutline User's
Guide*

© Copyright International Business Machines Corporation 1998, 2000. All rights reserved.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© 1998–2000 Hyperion Solutions Corporation. All rights reserved.

U.S. Patent Number: 5,359,724

Hyperion, Essbase, and Arbor are registered trademarks, and Hyperion Solutions and Hyperion Essbase are trademarks of Hyperion Solutions Corporation.

Microsoft is a registered trademark, and Windows is a trademark of Microsoft Corporation. IBM, DB2, Lotus, and 1-2-3 are registered trademarks of IBM Corporation. All other brand and product names are trademarks or registered trademarks of their respective holders.

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

Notice: The information contained in this document is subject to change without notice. Hyperion Solutions Corporation shall not be liable for errors contained herein or consequential damages in connection with the furnishing, performance, or use of this material.

Printed in the U.S.A.

Contents

Preface

Chapter 1: About Hyperion Integration Server

About OLAP	1-2
About Multidimensional Databases	1-3
User Interactions with Data	1-3
Sources for Data	1-5
About Hyperion Integration Server	1-5
Components of Hyperion Integration Server	1-7
Hyperion Integration Server Desktop	1-7
OLAP Integration Server	1-8
Workflow for Using Hyperion Integration Server	1-9
About Metaoutlines	1-10
Components of a Metaoutline	1-10
Workflow for Creating a Metaoutline	1-12

Chapter 2: Creating and Working with Metaoutlines

Understanding the Metaoutline Workflow	2-2
Working with Metaoutlines	2-3
Starting Hyperion Integration Server Desktop	2-3
Creating or Editing Metaoutlines	2-4
Correcting Connection Problems	2-6
Working in OLAP Metaoutline Assistant	2-7
Process Graphics	2-8
Inline Help Text	2-8
Tasks to Complete	2-8

Contents

Buttons	2-9
Tasks That Cannot Be Performed Using OLAP Metaoutline Assistant	2-10
Working in the OLAP Metaoutline Standard User Interface	2-11
Left Frame	2-12
Right Frame	2-12
Tools and Toolbars	2-13
Menus	2-13
Status Bar	2-13
Viewing the OLAP Metaoutline Main Window	2-13
OLAP Metaoutline Filter and Transformation Graphics	2-14
Selecting Multiple Objects in the Left Frame	2-15
Selecting Multiple Objects in the Right Frame	2-16
Validating Metaoutlines Manually	2-19
Saving Metaoutlines	2-21
Understanding Exclusive Access	2-22
Saving a Metaoutline Manually	2-23
Changing Read/Write Access	2-26
Saving While Closing	2-26
Saving a Metaoutline to a Different Name	2-27
Printing Metaoutlines	2-27
Deleting Metaoutlines	2-28

Chapter 3: Adding Dimensions, Member Levels, and Measures to a Metaoutline

Understanding the Metaoutline Workflow	3-2
About Dimensions, Member Levels, and Measures	3-3
About Dimensions	3-4
About Member Levels and Members	3-5
About Measures	3-6
About User-Defined Dimensions and Members	3-7
About Attribute Dimensions and Members	3-8
About Naming	3-8
Creating Dimensions, Member Levels, and Measures	3-11
Creating Dimensions Associated with an OLAP Model	3-12
Creating Member Levels Associated with an OLAP Model	3-15
Creating Recursive Dimensions	3-17

Creating User-Defined Dimensions and Members	3-18
Creating as Children	3-19
Creating as Siblings	3-20
Defining the Load Member for a User-Defined Dimension	3-21
Creating a Measure Dimension	3-23
Creating Attribute Dimensions and Member Levels	3-24
Comparison of UDAs and Attributes	3-25
Creating Attribute Dimensions from OLAP Model Dimensions	3-26
Naming Attribute Members	3-27
Setting the Member Names Format	3-27
Setting the Calculations Name Format	3-28
Specifying Attribute Boolean Names and Date Formats	3-31
Specifying Numeric Boundaries for Attributes	3-33
Renaming and Selecting a Data Type for an Attribute Member	3-35
Choosing a Data Type for an Attribute Dimension	3-36
Setting Base Dimension Levels for an Attribute	3-37
Setting Boolean Parameters for an Attribute	3-39
Setting Numeric Ranges for Attribute Members	3-42
Seeing Attribute Results in Spreadsheets	3-44
Renaming Dimensions, Member Levels, and Measures	3-44
Renaming in the Right Frame	3-45
Renaming in the Dimension Properties or Member Properties Dialog Box	3-46
Renaming a Single Measure	3-47
Deleting Dimensions, Member Levels, and Measures	3-49
Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines	3-50
Viewing Defined Measures	3-52

Chapter 4: Setting Load Properties

Understanding the Metaoutline Workflow	4-2
Setting Load Properties for Dimensions	4-3
About Duplicate Member Names	4-3
Understanding Rules for Shared Members	4-6
Handling Duplicate Member Names	4-6
Handling New Members During Data Load	4-8

Contents

Setting Load Properties for Member Names	4-11
Transforming Member Names	4-12
Replacing Parts of Member Names	4-13
Dropping Spaces From Around Member Names	4-16
Converting Spaces Within Member Names to Underscores	4-17
Attaching Prefixes and Suffixes to Member Names	4-18
Changing the Case of Member Names	4-22
Setting Load Properties for Members	4-24
Sorting Members	4-24
Setting the Member and Data Load Build Level	4-25
Setting Load Properties for Measures	4-27
Scaling Measures	4-28
Reversing Measure Signs	4-29
Changing the Column Associated with a Measure	4-30

Chapter 5: Working with Filters

Understanding the Metaoutline Workflow	5-2
About Filters	5-3
About Metaoutline Filters	5-4
About Member Filters	5-6
About Measure Filters	5-7
Filtering in an OLAP Model or in a Metaoutline	5-8
Working with Metaoutline Filters	5-8
Selecting a Metaoutline Filter	5-8
Creating a Metaoutline Filter	5-10
Viewing the Contents of a Metaoutline Filter	5-12
Deleting a Metaoutline Filter	5-13
Creating a Member Filter Interactively	5-14
Viewing the SQL Statement for a Filter	5-19
Typing a Member Filter	5-21
Creating a Measure Filter Interactively	5-23
Typing a Measure Filter	5-26
Creating an Attribute Filter Interactively	5-28

Chapter 6: Setting Hyperion Essbase Properties

Understanding the Metaoutline Workflow	6-2
Specifying Dimension and Member Properties	6-3
About Dimension Types	6-3
Setting a Dimension Type	6-7
Specifying Dimension Storage Properties	6-8
Specifying Data Storage Properties for Dimensions and Members	6-9
Setting Aliases for Dimensions and Members	6-12
Specifying Two-Pass Calculation on Dimensions and Members	6-14
Specifying Member Consolidation Properties	6-16
Setting Members as Dynamic Time Series	6-18
Specifying Account Properties	6-19
Specifying Time Balance Properties	6-20
Specifying Variance Reporting Properties	6-23
Specifying UDAs	6-25
Understanding Rules for UDAs	6-25
Defining a UDA	6-26
Specifying Formulas	6-27
Viewing Hyperion Essbase Properties	6-29

Chapter 7: Loading Dimensions and Members

Understanding the Metaoutline Workflow	7-2
About Loading	7-3
About Naming Applications and Databases	7-4
Procedure 1: Preparing to Load Members	7-4
Using Query Governors	7-5
Procedure 2: Starting a Member Load	7-8
Procedure 3: Loading Members Immediately	7-13
Procedure 4: Scheduling a Member Load	7-17
Starting the Windows NT Schedule Service Automatically	7-18
Starting the Windows NT Schedule Service Manually	7-19
Viewing Scheduled Member and Data Loads	7-19
Deleting Scheduled Member and Data Loads	7-22

Chapter 8: Loading Data

Understanding the Metaoutline Workflow	8-1
Procedure 1: Preparing to Load Data	8-3
Procedure 2: Starting a Data Load	8-4
Procedure 3: Loading Data Immediately	8-8
Procedure 4: Scheduling a Data Load	8-12

Chapter 9: Creating Drill-Through Reports

About Drill-Through	9-1
Adding a Report	9-2
Creating a New Report	9-3
Copying an Existing Report	9-5
Specifying the Contents of a Report	9-5
About Intersection Levels	9-6
Following the Rules for Intersection Levels	9-7
Determining Where to Set an Intersection Level	9-7
Reviewing Sample Drill-Through Reports	9-8
Adding Intersection Levels	9-11
Adding Drill-Through Columns	9-13
Setting the Size of a Query	9-15
Specifying the Format of a Report	9-16
Specifying Column Ordering	9-17
Specifying Row Ordering	9-19
Specifying Spreadsheet User Actions	9-21
Customizing a Report	9-22
Specifying the OLAP Integration Server Name	9-23
Specifying a Database Username and Password	9-24
Working with Reports	9-26
Renaming a Report in the Drill-Through Reports Dialog Box	9-26
Renaming a Report in the Report Properties Dialog Box	9-27
Deleting an Individual Report, an Intersection Level, or a Drill-Through Column	9-28
Deleting All Intersection Levels or Drill-Through Columns in a Report	9-28

Chapter 10: Troubleshooting

Problems with Creating, Validating, Saving, or Deleting Metaoutlines	10-1
Problems with Transformations	10-2
Problems with Filters	10-3
Failure of a Member Load or Data Load	10-4
Problems with Loading Members or Data	10-6

Appendix A: Creating a Sample Metaoutline Containing Attribute Dimensions

Starting up and Logging On	A-2
Opening the Standard User Interface to Create a New Metaoutline	A-3
Working in the OLAP Metaoutline Main Window	A-5
Viewing OLAP Model Dimensions and Attribute-enabled Columns	A-5
Creating Hierarchies, Dimensions, and Attribute Dimensions	A-7
Associating Level 0 Attribute Members with Members of their Base Dimension	A-10
Setting a Numeric Range for an Attribute Member	A-11
Using Attribute Calculations	A-14
Using Attribute Boolean Properties	A-16
Creating the Measures Dimension	A-21
Setting Consolidation and Calculation Methods	A-21
Assigning Aliases to Shared Members	A-26
Saving the Metaoutline	A-26

Glossary

Index

Contents

Preface

Purpose

This guide provides the information you need to build OLAP metaoutlines. It explains all Hyperion® Integration Server™ OLAP metaoutline features and options, and it contains the concepts, processes, procedures, formats, tasks, and examples that you need to use the Hyperion Integration Server Desktop software.

Audience

This guide is for Hyperion Integration Server database administrators who are responsible for installing, implementing, and deploying the system.

To use all the information in this book, you need the following skills:

- Knowledge of where the data for your business resides; for example, in a relational database.
- Knowledge of how to create and maintain Open Database Connectivity (ODBC) data sources.
- Knowledge of the data requirements for your business so you can apply the Hyperion® Integration Server product family to your specific application.
- A fundamental understanding of Microsoft Windows and basic Microsoft Windows terminology, such as dialog box, list box, and button. See Microsoft Windows documentation for more information on these terms.
- Experience with the setup and operation of Hyperion® Essbase® OLAP Server.
- A basic understanding of multidimensional concepts.

Document Structure

This document contains the following information:

Chapter 1, “About Hyperion Integration Server,” introduces basic OLAP and multidimensional concepts and describes how to use the desktop to create metaoutlines.

Chapter 2, “Creating and Working with Metaoutlines,” describes how to open a metaoutline, connect to OLAP Metadata Catalog and to a relational data source, and create, save, delete, and print a metaoutline.

Chapter 3, “Adding Dimensions, Member Levels, and Measures to a Metaoutline,” describes how to use the dimensions, member levels, and measures defined in the OLAP model to create a metaoutline.

Chapter 4, “Setting Load Properties,” describes how to set the actions that OLAP Integration Server performs on dimensions, member levels, and measures as it builds a Hyperion Essbase outline or loads data into a Hyperion Essbase outline.

Chapter 5, “Working with Filters,” demonstrates how to set filters on member levels and measures that determine what member levels or data OLAP Integration Server loads into the Hyperion Essbase database.

Chapter 6, “Setting Hyperion Essbase Properties,” describes how to set the properties that dimensions and member levels will have after you build the Hyperion Essbase outline.

Chapter 7, “Loading Dimensions and Members,” steps you through using a metaoutline to add dimensions and member levels to a Hyperion Essbase outline.

Chapter 8, “Loading Data,” steps you through loading data into a Hyperion Essbase database.

Chapter 9, “Creating Drill-Through Reports,” describes how to create Drill-Through reports that users can access using Hyperion® Essbase® Spreadsheet Add-in.

Chapter 10, “Troubleshooting,” provides tips on dealing with common problems you may encounter while building a Hyperion Essbase database and loading data into it.

Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions,” provides procedures for using the sample TBC application to create a metaoutline that contains attribute dimensions.

Glossary, lists and defines key Hyperion Integration Server terms and their definitions.

Index, lists Hyperion Integration Server terms and their page numbers. In the PDF version of this guide, select an index entry to view the relevant page.

Sample Applications

The examples used in this book are based on two sample databases provided with Hyperion Integration Server. The databases are called TBC (external data source) and TBC_MD (OLAP Metadata Catalog). The TBC_MD OLAP Metadata Catalog contains a sample OLAP model and a sample metaoutline. The OLAP model is called TBC Model, and the metaoutline is called TBC Metaoutline.

Note: The capitalization of column and table names in the sample applications depends on the relational database management system (RDBMS) that you use.

The person who installs Hyperion Integration Server is responsible for making the sample databases, the OLAP model, and the metaoutline available to end users. If any of the following problems occur when you launch Hyperion Integration Server Desktop, contact the Hyperion Integration Server administrator:

- You cannot find the sample databases, the OLAP model, or the metaoutline.
- You do not have adequate access to the sample databases, the OLAP model, or the metaoutline.
- You do not see any data in the sample databases.

Related Documentation

Hyperion provides the following documentation for this product:

- *Hyperion Integration Server Desktop OLAP Model User's Guide*, for information about basic modeling concepts and descriptions of how to design, create, and maintain one or more OLAP models on top of the relational database.
- *Hyperion Integration Server System Administrator's Guide*, for information about how to use OLAP Integration Server, OLAP Metadata Catalog, and OLAP Command Interface, and about how to create and manage ODBC connections.

- *Hyperion Integration Server Installation Guide*, for information about installing the software, about Hyperion Integration Server directories and files, and about connecting relational data sources to an OLAP Metadata Catalog.

Hyperion Essbase OLAP Server documents:

- *Hyperion Essbase OLAP Server Database Administrator's Guide*, for information about basic multidimensional concepts and descriptions of how to design, create, and maintain a Hyperion Essbase database.
- *Hyperion Essbase SQL Interface Guide*, for information about the data sources supported through Hyperion Essbase SQL Interface.
- *Hyperion Essbase Spreadsheet Add-in User's Guide*, for information about how to use Hyperion Essbase Spreadsheet Add-in with Microsoft Excel and Lotus 1-2-3 for Windows.
- The online *Quick Technical Reference* in the DOCS directory for information about Hyperion Essbase calculation and report commands and functions.

Online Help

- To access online help:
 1. In a dialog box, click **Help**.
 2. In the **Help** dialog box, to display information about an item, click the item.
- To print an online help topic, display the topic and select File > Print.

Online Guides

The online guides are electronic versions of the printed documentation.

- To display an online guide, select Help > Help Topics.

Conventions

The following table shows the conventions used in this document:

Table xi: Conventions Used in This Document

Item	Meaning
➤	Arrows indicate the beginning of a procedure consisting of sequential steps.
Brackets []	In examples, brackets indicate that the enclosed elements are optional.
Bold	Bold text indicates words or characters that you type exactly as they appear on the page. Bold in procedural steps highlights major user interface elements.
CAPITAL LETTERS	Capital letters denote commands and various IDs. (Example: CLEARBLOCK command)
Example text	Courier font indicates that the material shown is a code or syntax example.
Ctrl + 0	Keystroke combinations shown with the plus symbol (+) indicate that you should press the first key and hold it while you press the next key. Do not type the + symbol.
<i>Courier italics</i>	Courier italic text indicates a variable field in command syntax. Substitute a value in place of the variable shown in Courier italics.
<i>Italics</i>	Italics in a product-related term indicates that the term is also included in the glossary of the book.
Ellipses (...)	Ellipsis points indicate that text has been omitted from an example.
Mouse orientation	This document provides examples and procedures using a right-handed mouse. If you use a left-handed mouse, adjust the procedures accordingly.

Table xi: Conventions Used in This Document (Continued)

Item	Meaning
Menu options	Options in menus are shown in the following format: <i>Menu name > Menu command > Extended menu command</i> For example: Connections > OLAP Metadata Catalog > Connect
<i>n, x</i>	The variable <i>n</i> indicates that you must supply a generic number; the variable <i>x</i> indicates that you must supply a generic letter.

Note: The term right-click, used throughout this guide, means to click the secondary mouse button to open a pop-up menu.

About Hyperion Integration Server

Hyperion Integration Server Desktop provides a suite of graphical tools that you use to create OLAP models and metaoutlines. You use the tables of a relational data source to define a logical model that represents the relational data in an online analytical processing (OLAP) context. You then use the OLAP model to create a metaoutline that serves as a template for a Hyperion Essbase database outline.

This chapter explains basic OLAP concepts, describes a metaoutline, provides a workflow for using OLAP Metaoutline Assistant or the Hyperion Integration Server Desktop standard user interface, and provides an overview of Hyperion Integration Server.

This chapter contains the following topics:

- “About OLAP” on page 1-2
- “About Multidimensional Databases” on page 1-3
- “About Hyperion Integration Server” on page 1-5
- “Components of Hyperion Integration Server” on page 1-7
- “Workflow for Using Hyperion Integration Server” on page 1-9
- “About Metaoutlines” on page 1-10
- “Components of a Metaoutline” on page 1-10
- “Workflow for Creating a Metaoutline” on page 1-12

Note: You must create an OLAP model before you create a metaoutline. For more information on creating OLAP models, see the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.

About OLAP

OLAP is a decision-support computing environment for business managers who need to analyze consolidated enterprise data in real time. OLAP enables users to answer complex “what if” questions and to create sales and marketing scenarios to test budgeting, sales promotion, and sales planning strategies. Hyperion Essbase OLAP Server supports a multidimensional, multi-user database that enterprise users can access with standard retrieval tools, such as spreadsheets.

Hyperion Essbase OLAP Server supports multiple views of data sets for users who need to analyze the relationships between data categories. For example, a decision-support user might want answers to the following questions:

- How did Product A sell last month? How does this figure compare to sales in the same month in each of the last five years? How did Product A sell by branch, region, and territory? How will Product A sell next month, next quarter, and next year?
- Did Product A sell better in particular regions? Do regional trends exist?
- Did customers return Product A last year? Were returns due to product defects? Were defective products manufactured in a specific plant?
- Did commissions and pricing affect how salespeople sold Product A? Did particular salespeople do a better job of selling the product?

You can use Hyperion Integration Server to create Hyperion Essbase databases and thus answer these types of questions quickly and easily.

The first step in advancing from a relational database to a Hyperion Essbase database is the creation of a logical data model that represents the relational database. You use Hyperion Integration Server Desktop to create the data model. Then you use OLAP Metaoutline to create one or more metaoutlines and use a metaoutline to build a Hyperion Essbase database and load members and data into the database.

About Multidimensional Databases

A multidimensional database stores consolidated data at the intersections of its members and dimensions. For example, if a company sells a total of 20 units of all products in the East region in the first quarter, Hyperion Essbase stores 20 at the intersection of Product, East, Quarter1, and Unit Sales.

Dimensions represent core components of a business plan and often relate to business functions. Product, Region, and Year are examples of typical dimensions. Dimensions are static in most databases and rarely change over the life of an application.

A *member* is an individual component of a dimension. For example, Product A, Product B, and Product C might be members of the Product dimension. Each member has a unique name. A dimension can contain an unlimited number of members. Members in some dimensions change frequently over the life of an application.

Members can, simultaneously, be parents of members and children of members. The Hyperion Essbase outline indents members below one another to indicate a consolidation relationship. For example, sales totals for the Product dimension might be totalled by product description, broken down by product code, and further broken down by product ID.

User Interactions with Data

Hyperion Essbase OLAP Server consolidates and calculates data to provide different views of the data. With a multidimensional database, users can perform all of the following functions:

Consolidate (aggregate or roll up) data

The consolidation process computes the data relationships for all parent-child combinations within a dimension. For example, the consolidation for the Year dimension is as follows:

$$\text{Year} = \text{Quarter1} + \text{Quarter2} + \text{Quarter3} + \text{Quarter4}$$

A consolidation is typically additive, but it can be any type of calculation.

Create sophisticated “what if” scenarios

For example, assume that you set a sales goal of ten percent growth for the upcoming year on all product lines that your company manufactures. With Hyperion Essbase, you can compare the sales forecasts, as estimated by planners, with the actual data retrieved from the online transaction processing (OLTP) database to see whether you are achieving your sales goals. If actual sales are running lower than projected sales for a given period, salespeople can access the forecasted data, input new product sales scenarios (for example, What if I sell 2000 widgets to our biggest corporate customer?), update the forecasted data, and provide revised forecasted data to headquarters.

Input strategic planning assumptions

Assume that your company is planning for 50 percent growth over the next three years. You have a fairly good grasp of how many new products are required to meet your goal, but how do you know how many new engineers, salespeople, and support personnel you can afford to add while still optimizing profits and gross margin? With Hyperion Essbase, you can input projected sales and projected expenses for each product and calculate downward to determine the cost of goods sold. Thus, you obtain a realistic picture of the bottom line. If the picture does not look practical, you can create different scenarios with different mixes of products, people, and expenses until you produce the profit picture that you require.

Conduct spreadsheet operations

Drilling down or drilling up on data in Hyperion Essbase Spreadsheet Add-in retrieves progressively more detailed or progressively less detailed data relative to a selected dimension. Drilling down on a database dimension provides you with greater detail on the dimension. Drilling up provides you a higher level perspective of the dimension—a view of a higher consolidation level. For example, you drill down on the Year dimension to view the values for each quarter. Then you drill down on Quarter1 to see the values for each month of Quarter1. Conversely, you view Quarter1 sales for the Chicago office and drill up to view sales totals for the entire East region.

Pivoting data alters the user's view of retrieved data. When Hyperion Essbase retrieves a dimension, it displays a configuration of rows and columns. A user can pivot (rearrange) the data to obtain a different view.

For more information about spreadsheet operations, such as drill up, drill down, and pivot, see the *Hyperion Essbase Spreadsheet Add-in User's Guide*.

Sources for Data

The data in a multidimensional database can originate from a variety of sources, such as OLTP databases, text files, and spreadsheet files. OLTP databases contain a wealth of operational data, such as the answers to the following questions:

- How many units of Product A are on hand?
- What is the current customer's name, address, and billing status?
- What is an employee's salary, job title, and address?

Data in an OLTP database must be up to date and easy to change and must use as little space as possible. The data is stored in rows and columns in relational database tables.

Hyperion Integration Server enables you to access data for a metaoutline from a data source that is configured for Open Database Connectivity (ODBC) and that supports SQL 92 or later.

About Hyperion Integration Server

Hyperion Integration Server transfers data from relational tables to a Hyperion Essbase database quickly. To enable the transfer, you must determine which data to transfer and consolidate the selected data into a form that is useful for decision-support users. Then you must identify the tables, rows, or columns that contain the required data and determine how the tables, rows, or columns map to the structure of the multidimensional database.

About Hyperion Integration Server

As illustrated in Figure 1-1, Hyperion Integration Server provides graphical tools to help you accomplish the following tasks:

- Use the tables, views, and columns in a relational database to create an OLAP model. An OLAP model is a logical star schema that consists of a fact table surrounded by related dimension tables.
- Use the OLAP model to create a metaoutline—an outline template that contains the structure and the rules required to generate a Hyperion Essbase outline.
- Use the metaoutline to create and populate a Hyperion Essbase database.

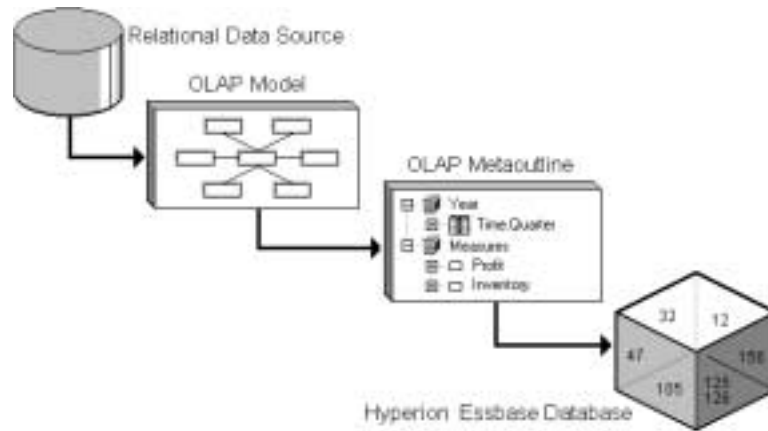


Figure 1-1: Workflow for Creating a Hyperion Essbase Database from a Relational Data Source

Components of Hyperion Integration Server

Hyperion Integration Server, pictured in Figure 1-2, consists of two major components: Hyperion Integration Server Desktop and OLAP Integration Server.

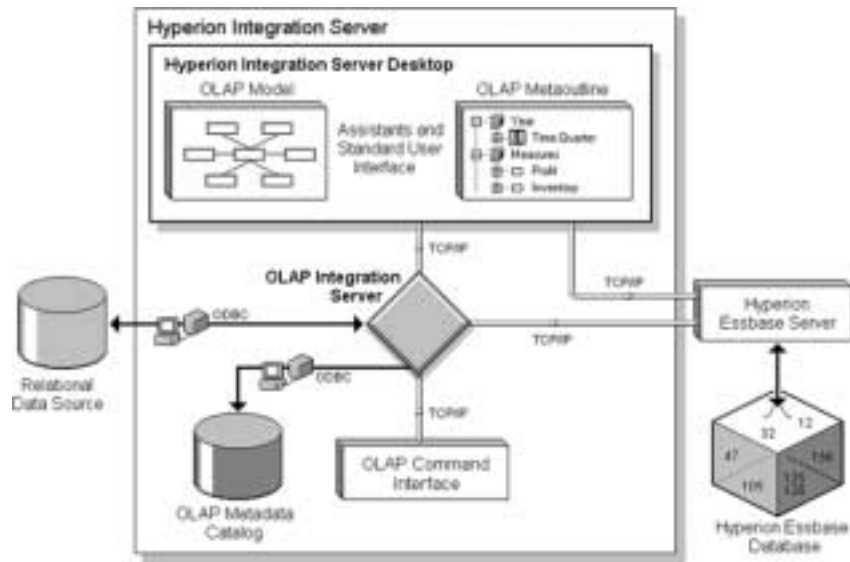


Figure 1-2: Hyperion Integration Server

Hyperion Integration Server Desktop

Hyperion Integration Server Desktop, a graphical user interface, is used for creating OLAP models and metaoutlines. Hyperion Integration Server Desktop includes the following components:

- OLAP Model Assistant:** A wizard for building centralized and reusable OLAP models that are based on a relational data source. The relational data source consists of the tables and views in a single relational database and contains the data that you want to report in Hyperion Essbase.

Use OLAP Model Assistant to create a basic OLAP model. For more complex model-building functions, such as recursive self-joins, use the standard Hyperion Integration Server Desktop user interface. Use the OLAP models that you create from information in the relational data source to build one or more metaoutlines.

- **OLAP Metaoutline Assistant:** A wizard for creating one or more metaoutlines from an OLAP model. A metaoutline determines the structure and content of the final Hyperion Essbase database outline.

Use OLAP Metaoutline Assistant to create a basic metaoutline. For more complex metaoutline functions, such as scheduling member loads, use the standard Hyperion Integration Server Desktop user interface. After you create a metaoutline, you can build a Hyperion Essbase database and load data from the relational data source directly into the Hyperion Essbase database.

Note: This user's guide provides detailed procedures for creating and editing a metaoutline using the standard user interface. To create a metaoutline using OLAP Metaoutline Assistant, refer to "Working in OLAP Metaoutline Assistant" on page 2-7 (or click Overview in the Welcome dialog box), the assistant online help, and the inline help that appears within each assistant tab.

- **Hyperion Integration Server Desktop standard user interface:** A graphical user interface for creating OLAP models and metaoutlines. Use the standard user interface to perform advanced functions, such as performing recursive self-joins in an OLAP model or scheduling member loads for a metaoutline.

To create a metaoutline using the standard user interface, you must connect to OLAP Metadata Catalog and the relational data source. For more information, see Chapter 2, "Creating and Working with Metaoutlines".

OLAP Integration Server

OLAP Integration Server is the primary component of Hyperion Integration Server. OLAP Integration Server is software that uses the information stored in OLAP Metadata Catalog to extract from the relational data source the dimension and member names needed to build an associated Hyperion Essbase outline. When the Hyperion Essbase outline is complete, OLAP Integration Server extracts data from the relational data source, performs the operations specified in the associated metaoutline, and loads the data into the Hyperion Essbase database. For more information about OLAP Integration Server, see the *Hyperion Integration Server System Administrator's Guide*.

OLAP Integration Server includes the following subcomponents, as previously illustrated in Figure 1-2:

- **OLAP Metadata Catalog:** An SQL relational database that contains the following information:
 - Metadata describing the nature, source, location, and type of data to retrieve from the relational data source
 - Metadata describing the information required to generate a Hyperion Essbase outline
 - OLAP models and metaoutlines

You can create more than one OLAP Metadata Catalog to store OLAP models and metaoutlines. However, you cannot move OLAP models and metaoutlines to a different OLAP Metadata Catalog after you have created and saved them to a specific catalog.

OLAP Metadata Catalog is a data source that is configured for Open Database Connectivity (ODBC). If you do not know how to create an ODBC data source, see the *Hyperion Integration Server Installation Guide* or the ODBC user documentation.

- **OLAP Command Interface:** A command-line tool used to access OLAP Integration Server to perform operations on the Hyperion Essbase outline and the data in the Hyperion Essbase database. For more information, see the *Hyperion Integration Server System Administrator's Guide*.

Workflow for Using Hyperion Integration Server

Figure 1-3 provides an overview of Hyperion Integration Server components that you use to prepare relational data for OLAP reporting through Hyperion Essbase OLAP Server.

- To create a Hyperion Essbase database from a relational data source:
 1. Build an OLAP model that is based on the tables of a relational data source. OLAP Integration Server stores the OLAP model and the information necessary to retrieve the relevant tables and views in OLAP Metadata Catalog.
 2. Create a metaoutline from the OLAP model. OLAP Integration Server stores the metaoutline in OLAP Metadata Catalog.
 3. Load members and data into an associated Hyperion Essbase database.
 4. Update the Hyperion Essbase outline with new members and data.

About Metaoutlines

OLAP metaoutlines have the following features:

- They are reusable. You can use a metaoutline as the basis for more than one Hyperion Essbase outline.
- They are centralized. You can define a metaoutline at a central location and use it to create multiple Hyperion Essbase outlines in multiple locations.
- They enable you to create a Hyperion Essbase outline that is based on criteria that you specify.
- They enable you to create Hyperion Essbase databases on demand. You can create or update a Hyperion Essbase database whenever you want, either on the spot or by scheduling periodic updates.
- They enable you to view sample Hyperion Essbase outlines before you build them.
- They automatically generate the SQL necessary to retrieve data from an external data source.
- They enable you to filter the data from the relational data source before you build the associated Hyperion Essbase outline.
- They enable you to transform data as you create a Hyperion Essbase outline. You can transform member names as you create a Hyperion Essbase outline or transform data as you load it into a Hyperion Essbase database.

Components of a Metaoutline

A metaoutline contains all of the following components:

- One or more measures. The Hyperion Essbase database calculates data values, or *measures*, for each dimension intersection of the associated metaoutline. Measures include items such as SALES or COGS (cost of goods sold). Every metaoutline that you use to build a Hyperion Essbase outline must include at least one measure. For more information, see “About Measures” on page 3-6.
- Two or more dimensions. A *dimension* in a metaoutline creates a dimension in the associated Hyperion Essbase outline. For example, if you have a Product dimension in a metaoutline, the Hyperion Essbase outline that you generate from that metaoutline also has a Product dimension. You can drag dimensions from the OLAP model or create your own dimensions. For more information, see “About Dimensions” on page 3-4.

- One or more member levels. A *member level* in a metaoutline creates one or more members at the same level in the associated Hyperion Essbase outline. For example, if the Product dimension of a metaoutline contains a PRODUCT_DESC member level, the Product dimension in the Hyperion Essbase outline contains members, such as Birch Beer and Caffeine Free Cola, that correspond to the values in the PRODUCT_DESC member in the relational database.

You can define a set of filters and transformation rules that determine which members of a member level Hyperion Integration Server adds to the associated Hyperion Essbase outline and what transformations, if any, Hyperion Integration Server performs on the members of a member level as it builds the associated Hyperion Essbase outline. For more information, see “About Member Levels and Members” on page 3-5.

You can arrange member levels into a hierarchy. A *hierarchy* defines the organizational structure of a group of member levels. For example, a standard time hierarchy in a metaoutline might contain the Year, Quarter, and Month member levels. This hierarchy creates a Hyperion Essbase outline with members such as 1999, Quarter1, and January.

You can also create hierarchies for the metaoutline as you create an OLAP model. See the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.

Figure 1-3 shows a sample metaoutline. You can see the dimensions named Year, Product, Market, Scenario, and Supplier. The metaoutline also contains a dimension named Measures that contains OPENINGINVENTORY and ADDITIONS measures. The Year dimension contains a hierarchy—Quarter and Month.

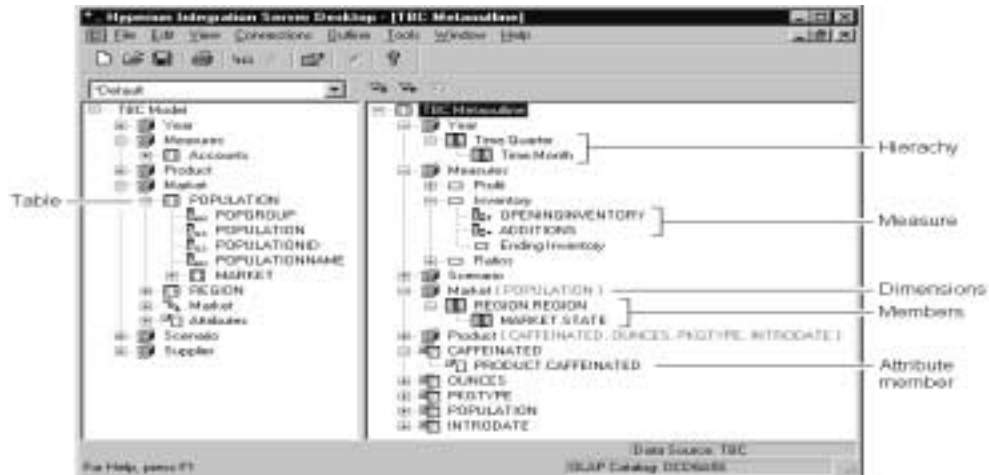


Figure 1-3: Sample Metaoutline

Workflow for Creating a Metaoutline

Before you can create a metaoutline, you must have the following pieces in place:

- An OLAP model.
See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.
- A running and properly prepared external data source.
- A server running OLAP Integration Server.
See the *Hyperion Integration Server System Administrator's Guide*.
For more information, see "Working with Metaoutlines" on page 2-3.
- A running Hyperion Essbase server if you plan to load members or data into a Hyperion Essbase database.

After you make the necessary connections, use the following workflow to create a metaoutline, as illustrated in Figure 1-4:

- 1.** Start Hyperion Integration Server Desktop and use OLAP Metaoutline Assistant or the Hyperion Integration Server Desktop standard user interface to create a metaoutline.

See “Starting Hyperion Integration Server Desktop” on page 2-3 and “Creating or Editing Metaoutlines” on page 2-4.
- 2.** Add dimensions, member levels, and measures to the metaoutline.

See Chapter 3, “Adding Dimensions, Member Levels, and Measures to a Metaoutline.”
- 3.** Set load properties, filters, and Hyperion Essbase properties on the dimensions, member levels, and measures of the metaoutline.

See Chapter 4, “Setting Load Properties,” Chapter 5, “Working with Filters,” and Chapter 6, “Setting Hyperion Essbase Properties.”

As needed, view sample dimensions or outlines to check them before building the Hyperion Essbase outline. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.

As needed, create Drill-Through reports. See Chapter 9, “Creating Drill-Through Reports.”
- 4.** Load member levels into the Hyperion Essbase outline.

See Chapter 7, “Loading Dimensions and Members.”
- 5.** Load data into the Hyperion Essbase outline.

See Chapter 8, “Loading Data.”

As needed, view the data by using Hyperion Essbase Spreadsheet Add-in. See the *Hyperion Essbase Spreadsheet Add-in User’s Guide*.

6. Update the Hyperion Essbase database with new members and data.

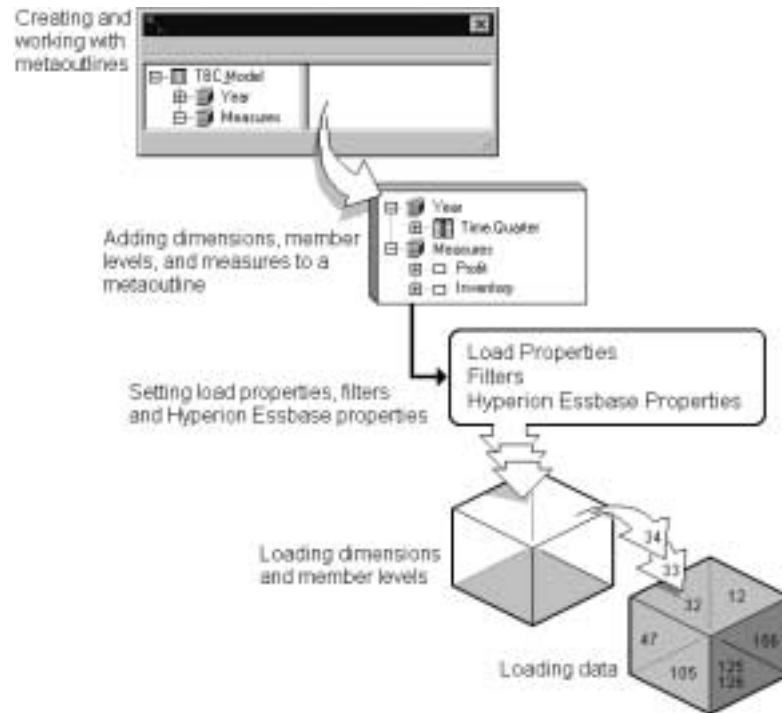


Figure 1-4: Metaoutline Workflow

When you finish, the following structures are functional:

- A metaoutline that determines the structure of the associated Hyperion Essbase outline
- A Hyperion Essbase database that contains both members and data

Creating and Working with Metaoutlines

This chapter describes how to create and work with metaoutlines—from opening or creating a metaoutline through validating and saving a metaoutline. Before you read this chapter, read Chapter 1, “About Hyperion Integration Server” to familiarize yourself with the definition for and components of a metaoutline.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 2-2
- “Working with Metaoutlines” on page 2-3
- “Working in OLAP Metaoutline Assistant” on page 2-7
- “Working in the OLAP Metaoutline Standard User Interface” on page 2-11
- “Validating Metaoutlines Manually” on page 2-19
- “Saving Metaoutlines” on page 2-21
- “Printing Metaoutlines” on page 2-27
- “Deleting Metaoutlines” on page 2-28

For information on adding dimensions, member levels, and measures to a metaoutline, see Chapter 3, “Adding Dimensions, Member Levels, and Measures to a Metaoutline.”

Understanding the Metaoutline Workflow

Figure 2-1 illustrates the workflow for creating a metaoutline. This workflow includes creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following procedures:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and member levels from a metaoutline into the Hyperion Essbase outline
- Loading data that is specified in a metaoutline into a Hyperion Essbase database

This chapter focuses on creating a metaoutline and on working with an existing metaoutline.

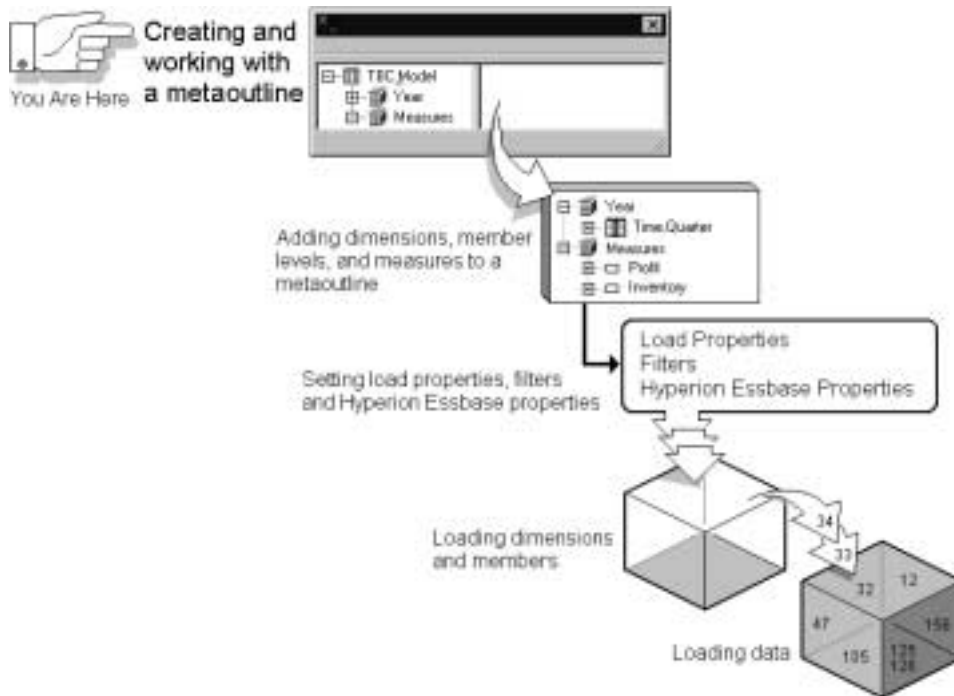


Figure 2-1: Metaoutline Workflow


Working with Metaoutlines

Before you start working with a metaoutline, make sure that the OLAP Metadata Catalog where the associated OLAP model is stored, the server on which OLAP Integration Server is running, and the data source are accessible. These items can be on servers on a network and do not need to be on your local computer to be available to Hyperion Integration Server Desktop. For more information, see the *Hyperion Integration Server System Administrator's Guide*.

The Hyperion Essbase server must be running if you need to create, change, or load data into a Hyperion Essbase database.

To work with a metaoutline, start Hyperion Integration Server Desktop and choose whether to use OLAP Metaoutline Assistant (to be guided through creating or editing a metaoutline) or to use the OLAP Metaoutline standard user interface (to create or edit a metaoutline without guidance). For more information, see “Starting Hyperion Integration Server Desktop” on page 2-3 and “Creating or Editing Metaoutlines” on page 2-4.

Starting Hyperion Integration Server Desktop

- To start Hyperion Integration Server Desktop, use any of the following methods:
 - From the Windows Start menu, select Hyperion Integration Server Desktop.
 - Double-click the Hyperion Integration Server Desktop icon, .
 - On the command line, type the executable name `olapbldr`.

For example, if Hyperion Integration Server Desktop is installed in the `\hyperion\is\bin` directory, type the following path:

```
\hyperion\is\bin\olapbldr
```

Note: The default client directory, which is specified during installation, is `\hyperion\is`. Your application may have a different installation directory. Consult the person who installed Hyperion Integration Server at your site if you are not sure of the directory specification.

Now you can start working on a metaoutline. See “Creating or Editing Metaoutlines” on page 2-4.

Creating or Editing Metaoutlines

To create or edit metaoutlines, start Hyperion Integration Server Desktop, connect to OLAP Integration Server, OLAP Metadata Catalog, and open a metaoutline. You are then prompted to connect to a data source.

For more information about connecting to OLAP Integration Server, OLAP Metadata Catalog or a data source, see the *Hyperion Integration Server Installation Guide*. If you have problems connecting, see “Correcting Connection Problems” on page 2-6.

If you started Hyperion Integration Server Desktop, the Welcome dialog box that you use to choose whether to create or edit a metaoutline opens automatically.

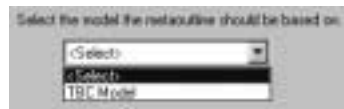
If you are already running Hyperion Integration Server Desktop and are connected to OLAP Metadata Catalog, select File > New or File > Open to open the Welcome dialog box.

Using OLAP Metaoutline Assistant is the quickest, easiest way to create basic metaoutlines or to edit metaoutlines. If, however, you want to perform more complex tasks, such as working with attributes, you must use the OLAP Metaoutline standard user interface. For more information, see “Tasks That Cannot Be Performed Using OLAP Metaoutline Assistant” on page 2-10.

- To use OLAP Metaoutline Assistant to create a metaoutline:



1. Select the **OLAP Metaoutline Assistant** icon.
2. From the drop-down list, select an existing model on which to base the metaoutline.

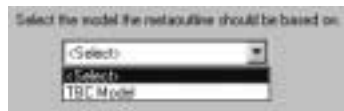


3. If you want to place write-lock privileges on the metaoutline, select **Exclusive Access**.
Exclusive Access enables you to modify and save the metaoutline. If **Exclusive Access** is not selectable, another user has write-lock privileges and you have read-only privileges or no privileges.
4. Click **Open**.

- To use the OLAP Metaoutline standard user interface to create a metaoutline:



1. Select the **OLAP Metaoutline** icon,
2. From the drop-down list box, select an existing model on which to base the metaoutline.



3. If you want to place write-lock privileges on the metaoutline, select **Exclusive Access**.

Exclusive Access enables you to modify and save the metaoutline. If **Exclusive Access** is not selectable, another user has write-lock privileges and you have read-only privileges.

4. Click **Open**.

- To use OLAP Metaoutline Assistant to edit a metaoutline:

1. In the **Welcome** dialog box, select the **Existing** or **Recent** tab.
2. Select the name of the metaoutline that you want to edit.

For more information about the metaoutline, look in the **Details** list box.

3. Select **Open with Assistant**.
4. If you want to place write-lock privileges on the metaoutline, select **Exclusive Access**.

Exclusive Access enables you to modify and save the metaoutline. If **Exclusive Access** is not selectable, another user has write-lock privileges and you have read-only privileges.

5. Click **Open**.

- To use the OLAP Metaoutline standard user interface to edit a metaoutline:
 1. In the **Welcome** dialog box, click the **Existing** or **Recent** tab.
 2. Select the name of the metaoutline name that you want to edit.
For more information about the metaoutline, look in the **Details** list box.
 3. Make sure that **Open with Assistant** is not selected.
 4. If you want to place write-lock privileges on the metaoutline, select **Exclusive Access**.
Exclusive Access enables you to modify and save the metaoutline. If **Exclusive Access** is not selectable, another user has write-lock privileges and you have read-only privileges.
 5. Click **Open**.
 6. If necessary, log on to the appropriate data source.

Correcting Connection Problems

If you have problems connecting from Hyperion Integration Server Desktop to OLAP Integration Server, OLAP Metadata Catalog, the Hyperion Essbase server, or the data source, use the following checklist to help diagnose any problems:

- Was the server temporarily down or not available? Try to connect again.
- Did you enter the correct server name? Depending on the network configuration, you may need to enter a fully-qualified internet host name; for example, **myserver.mycompany.com**.
- Did you enter the correct username and password? Passwords are usually case-sensitive. Make sure that the Caps Lock key is not activated.
- Do you have a network connection to the server? Check the network connection by using the ping command. See Microsoft Windows online help for more information about using ping.

If the above steps do not resolve the problem, contact your database administrator or system administrator. For more information about troubleshooting server connections, see the *Hyperion Integration Server System Administrator's Guide*.

Working in OLAP Metaoutline Assistant

OLAP Metaoutline Assistant is a wizard that steps you through creating or editing a metaoutline quickly and easily. OLAP Metaoutline Assistant contains multiple tabs. If you are creating a metaoutline, you complete the tabs and their tasks in order.

After you complete the required tasks in all tabs, you have a valid metaoutline that you can use to create one or more Hyperion Essbase outlines.

If you are editing an existing metaoutline, you can work through the tabs in any order to revise the metaoutline as needed. When you open OLAP Metaoutline Assistant, it displays the first tab, as shown in Figure 2-2.

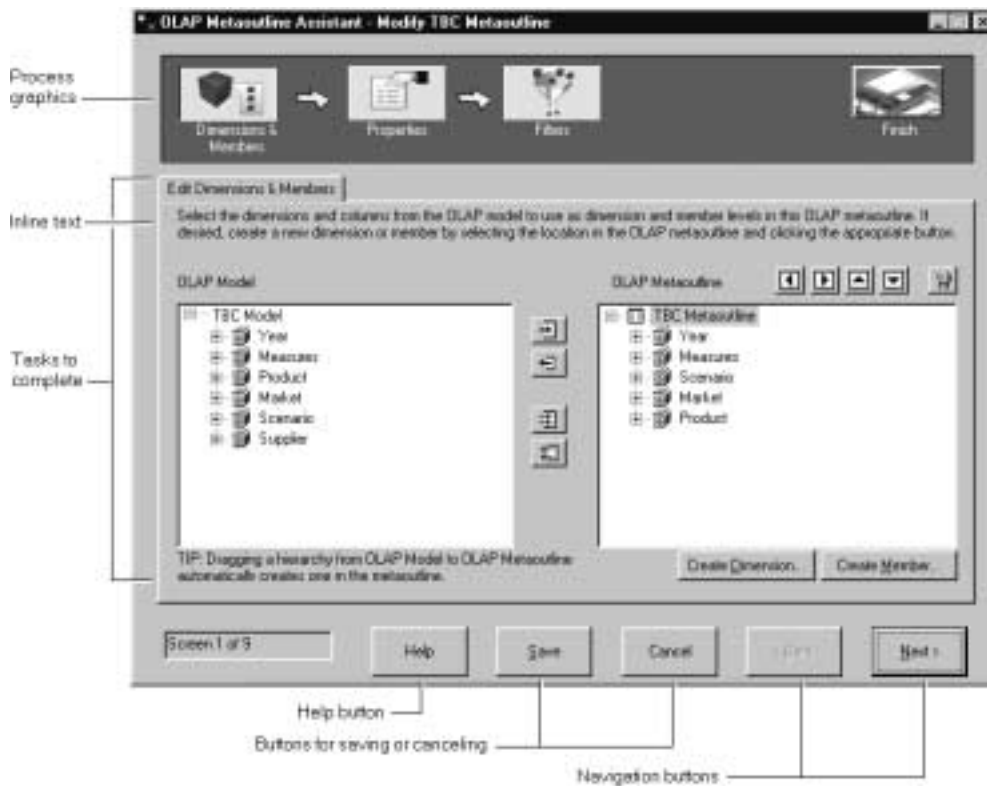


Figure 2-2: Edit Dimensions & Members Tab in OLAP Metaoutline Assistant

Process Graphics

The process graphics along the top of OLAP Metaoutline Assistant represent related tasks. When you create a metaoutline, the tasks must be completed in order from left to right. After you complete all the tabs that a process graphic represents, the next process graphic becomes accessible and you can begin work on tasks within that task group. When you edit an existing metaoutline, all process graphics are clearly displayed and all tabs are accessible.

Note: For best visibility of the OLAP Metaoutline Assistant process graphics, a color palette of 65536 colors is recommended. To set this palette, select Start > Settings > Control Panel. Double-click the Display icon, select 65536 from the Color Palette drop-down list, click Apply, and then click OK. Your PC may not support a palette of 65536 colors.

Inline Help Text

Each tab contains brief descriptions of the tasks that you need to perform. If you need additional information on a task, click the Help button.

Tasks to Complete

Each tab contains tasks that you must complete to define a specific part of a metaoutline. In the first tab, for example, you must select the dimensions and columns of the OLAP model that you want to use as dimensions and member levels in the metaoutline. For help on using any tab, click the Help button.

OLAP Metaoutline Assistant contains the following tabs:

Table 2-1: OLAP Metaoutline Assistant Tabs

Tab Name	More Information
Select Dimensions and Members	“About Dimensions, Member Levels, and Measures” on page 3-3
Set Dimension Properties	“Specifying Dimension and Member Properties” on page 6-3
Set Member Properties	“Specifying Dimension and Member Properties” on page 6-3
Set Account Properties	“Specifying Account Properties” on page 6-19

Table 2-1: OLAP Metaoutline Assistant Tabs (Continued)

Tab Name	More Information
Name Filters	“About Metaoutline Filters” on page 5-4
Assign Member Filters	“About Metaoutline Filters” on page 5-4
Assign Measures Filters	“About Measure Filters” on page 5-7
Review Filters	“Viewing the Contents of a Metaoutline Filter” on page 5-12
Preview Outline	“Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50
Load Data or Members into Hyperion Essbase	Chapter 7, “Loading Dimensions and Members” Chapter 8, “Loading Data”
Finish	“Saving Metaoutlines” on page 2-21

Buttons

The following buttons are displayed along the bottom of OLAP Metaoutline Assistant:

- **Help button.** Click to view detailed help for the selected tab. The help describes each field in the OLAP Metaoutline Assistant tabs and explains what you must do to complete the tasks in the selected tab before moving to the next tab.
- **Save button.** Click to save the metaoutline. You can save a metaoutline in any tab and then continue to move through the other tabs. Changes that you made to all previous tabs are saved.

When you save a metaoutline, it is validated automatically. You can save a metaoutline with validation problems, but you cannot use the metaoutline to create a Hyperion Essbase outline.

- **Cancel button.** Click to close the metaoutline without saving. Changes that you saved in previous tabs are not discarded. Changes that you did not save in previous tabs are discarded, as are all changes in the current tab.

- Back button. Click to move to the previous tab. You can move back through previous tabs and make changes at any time.
- Next button. Click to move to the next tab. If you are creating a metaoutline and have not completed all mandatory tasks (such as selecting dimensions and members), you cannot move to the next tab.

Tasks That Cannot Be Performed Using OLAP Metaoutline Assistant

You cannot use OLAP Metaoutline Assistant to perform the following tasks. If you need to perform these tasks in a metaoutline, use the OLAP Metaoutline standard user interface.

- Create attribute dimensions.
- Rename dimensions, members, and measures.
- Manually validate the metaoutline at any time during the creation or editing process.
- Set load properties for dimensions, members, and measures. These tasks include the following:
 - Handling duplicate member names
 - Transforming a member name
 - Removing spaces around a member level name
 - Attaching a prefix or a suffix to a member level name
 - Sorting member levels alphanumerically within a dimension
 - Scaling measures
 - Reversing the sign of a measure
 - Changing the column of a measure
- Set Dynamic Time Series
- Set query governors

- Create Drill-Through reports
- View relational source table data.

Note: This user's guide provides detailed procedures for creating and editing a metaoutline using the standard user interface. To create a metaoutline using OLAP Metaoutline Assistant, click Overview in the Welcome dialog box, or refer to the online help and the inline help that appears within each assistant tab.

Working in the OLAP Metaoutline Standard User Interface

To create a metaoutline using the standard user interface, you must connect to OLAP Metadata Catalog and to the relational data source. For more information on connecting, see “Working with Metaoutlines” on page 2-3. After you select the OLAP Metaoutline icon, Hyperion Integration Server Desktop displays the OLAP Metaoutline main window, as shown in Figure 2-3. The window displays the OLAP model in the left frame and a blank metaoutline creation area in the right frame.

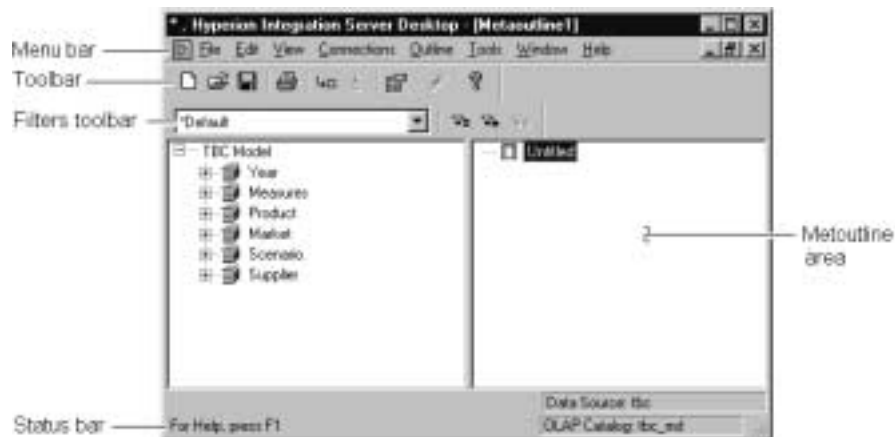






Figure 2-3: OLAP Metaoutline Main Window

Left Frame

The left frame lists the dimensions defined in the OLAP model. You can use OLAP model dimensions in the metaoutline.

Each OLAP model object or metaoutline object is displayed after a plus symbol, , or a minus symbol, .

- The plus symbol, , indicates that you can expand the branch. For example, clicking Market expands the branch to show REGION.
- The minus symbol, , indicates that you can collapse the branch. For example, when Market is expanded, clicking on Market collapses the dimension to show the dimension name but not the children of the dimension.

For information on selecting multiple dimensions, hierarchies, columns, and measures in the left frame, see “Selecting Multiple Objects in the Left Frame” on page 2-15.

Right Frame

The right frame is where you build a metaoutline. Drag a dimension, member, hierarchy, or measure from the left frame into the right frame to create a dimension, member level, hierarchy, or measure in the metaoutline.

For information on OLAP model objects that you can drag into the right frame, see “Creating Dimensions, Member Levels, and Measures” on page 3-11.

For information on selecting multiple dimensions, hierarchies, columns, and measures in the right frame, see “Selecting Multiple Objects in the Right Frame” on page 2-16.

Tools and Toolbars

You use the toolbar to perform commonly used functions. For more information on individual toolbar buttons, click Help. To hide the toolbar, deselect View > Toolbar. The check mark next to Toolbar in the menu is removed, and the toolbar disappears. To redisplay the toolbar, select View > Toolbar.

The toolbar is not attached to the edge of the program window. You can move it by selecting it and dragging it to a new location. You can also change the shape and size of the toolbar.

You use the Filters toolbar to perform actions on metaoutline filters. You can move the toolbar by selecting it and dragging it to a new location. You can also change the shape and size of the Filters toolbar.

Menus

You use the menu bar to perform commonly used commands.

You use the pop-up menu that opens when you right-click an object to perform operations on the selected object.

Status Bar

The status bar displays the data source and the OLAP Metadata Catalog to which you are connected and displays other status information. To hide the status bar, deselect View > Status Bar. The check mark next to Status Bar in the menu is removed, and the status bar disappears. To redisplay the status bar, select View > Status Bar.



Viewing the OLAP Metaoutline Main Window

You can customize the view of the OLAP Metaoutline main window to display Hyperion Essbase properties in either columnar or condensed format. Columnar format displays Hyperion Essbase properties under column headings as shown in Figure 2-4. Condensed format displays Hyperion Essbase properties in parentheses to the right of the dimension or member as they appear in a Hyperion Essbase outline. Default Hyperion Essbase settings are not displayed.

To customize the view of the OLAP Metaoutline main window, select View > Display Essbase Properties. For more information on the specific menu items listed on the Display Essbase Properties submenu, see “Viewing Hyperion Essbase Properties” on page 6-29.

OLAP Metaoutline Filter and Transformation Graphics

You can view the following filter and transformation icons in the OLAP Metaoutline main window by selecting View > Metaoutline Icons:

- The filter icon, , indicates that a filter has been set on a member level or on a measure.
- The transformation icon, , indicates that a member level or a measure has been transformed.

To hide these icons, deselect View > Metaoutline Icons.

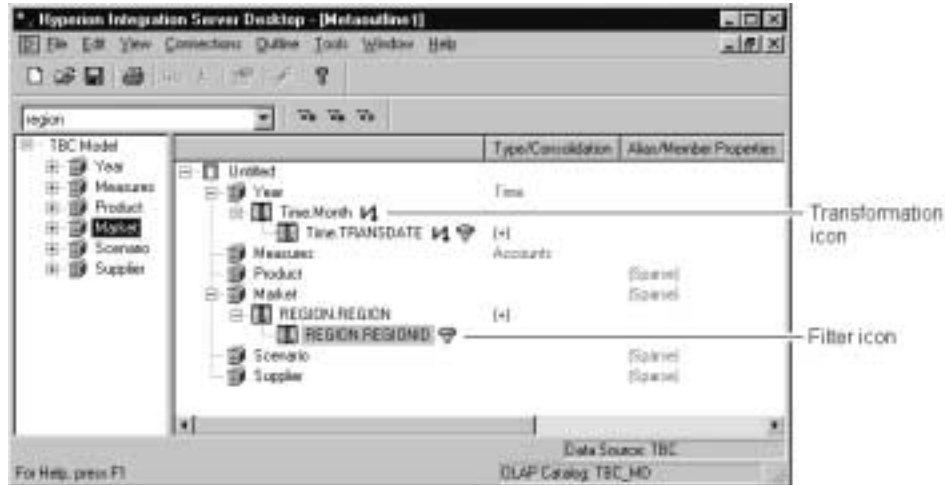


Figure 2-4: Viewing Hyperion Essbase Properties in Columnar Format

For further information on the icons displayed in the OLAP Metaoutline main window, as shown in Figure 2-4, see “Creating Dimensions Associated with an OLAP Model” on page 3-12.

Selecting Multiple Objects in the Left Frame

To select multiple, non-contiguous objects in the left frame of the OLAP Metaoutline main window, select the name of the first object, press and hold Ctrl, and select the names of the other objects.

To select multiple, contiguous objects, use one of the following methods:

- Select a first object, press and hold Ctrl, and select additional objects.
- Select a first object, press Shift, and select a final object. All objects between the two selected objects are selected.

You can perform any of the following actions to create dimensions and member levels in the right frame of the OLAP Metaoutline main window:

- Select multiple dimensions in the left frame and drag the dimensions into the right frame to create dimensions without member levels. You can then select multiple members of one or more dimensions in the left frame and drag the members into the right frame. Each member is placed under the correct dimension automatically.

Note: You may need to organize the dimension members into a hierarchy.

- Select multiple members in the left frame and drag the members into the right frame to create dimensions and member levels automatically.
- Select multiple dimensions and members in the left frame and drag the dimensions and members into the right frame to create dimensions and member levels automatically.

Creating and Working with Metaoutlines

As shown in Figure 2-5, if you drag the ADDITIONS, COGS, FAMILY, and FAMILYID members into the right frame, the Measures and Product dimensions are created automatically.

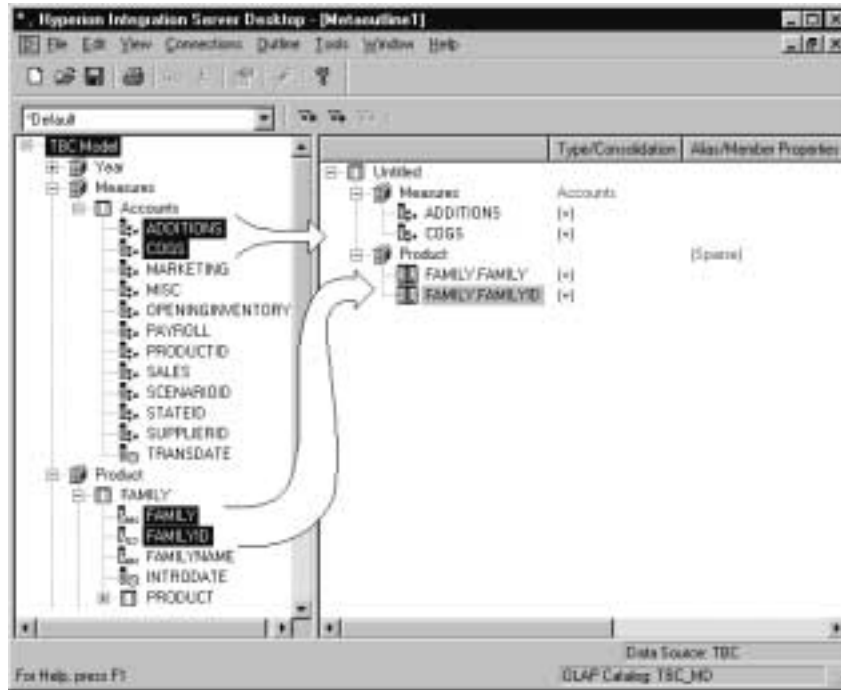


Figure 2-5: Selecting and Dragging Multiple Members into the Right Frame

Selecting Multiple Objects in the Right Frame

You can select multiple dimensions, member levels, and measures to perform the following actions:

- Set or change Hyperion Essbase properties
- Delete objects
- Create Drill-Through reports

To select multiple, non-contiguous objects in the right frame of the OLAP Metaoutline main window, select the name of the first object, press and hold Ctrl, and click the names of the additional objects.

To select multiple, contiguous objects, use one of the following methods:

- Select a first object, press Ctrl, and select the additional objects.
- Select a first object, press Shift, and select a final object. All objects between the two selected objects are selected.

When you select multiple dimensions, member levels, measures, and UDAs, the Edit and pop-up menu options vary, depending on your selection. The commands available for each type of multiple selection are shown in Table 2-2.

Table 2-2: Multiple Selection and Menu Command Availability

Selection	Menu Command Availability
Two or more dimensions	Delete command Properties command Drill-Through command
Two or more member levels	Delete command Properties command Drill-Through command
Some combination of dimensions, member levels, and measures	Delete command Drill-Through command
Two or more measures within the accounts dimension, including the accounts dimension	Delete command Drill-Through command
Two or more UDAs	Delete command Properties command Drill-Through command

Note: Selecting the metaoutline name along with a dimension, member level, or measure activates the Drill-Through command on the Edit menu. The pop-up menu is not selectable.

When you select a combination of dimensions, member levels, or UDAs, only specific tabs and commands are selectable on the Dimension Properties and Member Properties dialog boxes, as indicated in Table 2-3.

If you select two or more objects with different Hyperion Essbase properties, the default option in the dialog box is not displayed. For example, if you select two dimensions, one dimension tagged as sparse and one dimension tagged as dense, the Dimension Storage option on the Dimension Properties dialog box is selectable. In this situation, you must choose what the new Dimension Storage option should be for the selected dimensions.

Table 2-3: Multiple Selection: Tab and Command Availability

Selection	Tab Availability	Commands that Are Not Selectable
Two or more dimensions	Dimension Properties dialog box: <ul style="list-style-type: none"> • Member Info tab • UDAs tab 	<ul style="list-style-type: none"> • Alias command • Two-pass calculation is not selectable if a selected dimension is tagged as two-pass calculation and another selected dimension is not tagged as two-pass calculation
Two or more member levels	Member Properties dialog box: <ul style="list-style-type: none"> • Member Info tab • UDAs tab 	<ul style="list-style-type: none"> • Alias command • Two-pass calculation is not selectable if a selected member level is tagged as two-pass calculation and another selected member level is not tagged as two-pass calculation • Dynamic time is not selectable if two or more member levels belong to the time dimension, and a time member level is tagged as dynamic time and another time member level is not tagged as dynamic time

Table 2-3: Multiple Selection: Tab and Command Availability (Continued)

Selection	Tab Availability	Commands that Are Not Selectable
Two or more measures	Member Properties dialog box: <ul style="list-style-type: none"> Member Info tab Account Info tab UDAs tab 	<ul style="list-style-type: none"> Alias command
Two or more UDAs	Member Properties dialog box: <ul style="list-style-type: none"> Member Info tab UDAs tab 	<ul style="list-style-type: none"> Alias command Two-pass calculation is not selectable when a selected member level is tagged as two-pass calculation and another selected member level is not tagged as two-pass calculation

Note: Changing specific properties for dimensions or for member levels does not change the properties of other options. For example, selecting multiple dimensions and tagging them with two-pass calculation does not change the storage properties of the two-pass dimensions.

Validating Metaoutlines Manually

When you save a metaoutline, OLAP Integration Server automatically validates it and writes it to OLAP Metadata Catalog. If you prefer, you can validate a metaoutline manually before saving it. You can save a metaoutline that does not pass validation, but you cannot use it to load member levels or data into a Hyperion Essbase database.

To validate a metaoutline manually, select File > Validate. If the metaoutline is valid, a dialog box informs you that the metaoutline contains no errors.

If the metaoutline is invalid, Hyperion Integration Server displays a dialog box, similar to the one in Figure 2-6, that details the errors in the metaoutline.

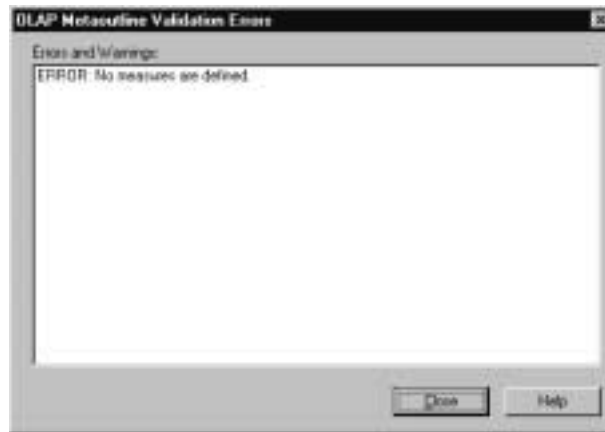


Figure 2-6: Invalid OLAP Metaoutline Validation Error Message Listing

If the metaoutline is invalid, check to make sure that the following conditions are met:

- The metaoutline contains at least one dimension.
- The metaoutline contains at least one measure. See “About Measures” on page 3-6.
- The metaoutline contains only one dimension or no dimension tagged as measure. See “Creating a Measure Dimension” on page 3-23.
- The metaoutline does not contain more measures than are allowed.
 - If you do not have a dimension tagged as measure in the metaoutline, the metaoutline can contain only one measure.
 - If you have a dimension tagged as measure in the metaoutline, the metaoutline can contain a measure for each level 0 (leaf level) member of the measure dimension.

For more information on measures, see “About Measures” on page 3-6.

- Each dimension or user-defined dimension contains one or more member levels.
- Each user-defined dimension has a default load member. For more information, see “Defining the Load Member for a User-Defined Dimension” on page 3-21.

To save a model that does not pass validation, see “Saving a Metaoutline Manually” on page 2-23.

Saving Metaoutlines

When you save a metaoutline, Hyperion Integration Server automatically validates it and writes it to OLAP Metadata Catalog. You must store the metaoutline in the OLAP Metadata Catalog that stores the OLAP model on which the metaoutline is based.

You can save a metaoutline even if it does not pass validation, but you cannot use an invalid metaoutline to load member levels or data into a Hyperion Essbase database.

You can save a metaoutline, even if multiple users are reading it, by selecting Exclusive Access in the Welcome dialog box. Opening an existing metaoutline with Exclusive Access guarantees that no other user can modify the metaoutline.

This topic describes how to save a metaoutline and contains the following subtopics:

- “Understanding Exclusive Access” on page 2-22
- “Saving a Metaoutline Manually” on page 2-23
- “Saving While Closing” on page 2-26
- “Saving a Metaoutline to a Different Name” on page 2-27

Understanding Exclusive Access

You can save a metaoutline even if multiple users are reading it by selecting Exclusive Access in the Welcome dialog box, as shown in Figure 2-7. Opening an existing metaoutline with Exclusive Access guarantees that no other user can modify the metaoutline.

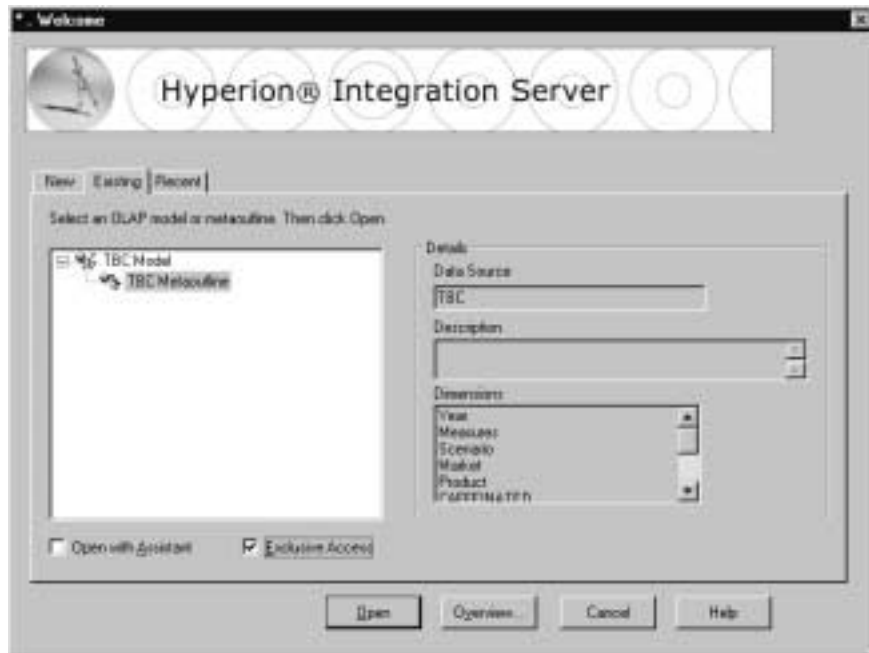


Figure 2-7: Selecting the Exclusive Access Option

When opening an existing metaoutline, select Exclusive Access to place write-lock privileges on the metaoutline. This option allows you to perform both of the following tasks:

- Save an existing metaoutline.
- Modify and save an existing metaoutline. If you open a read-only metaoutline and no other user has Exclusive Access, you can save the metaoutline to the same name.

If Exclusive Access is not selectable, another user has write-lock privileges, and you have read-only privileges.

If you want to modify and save a metaoutline but have read-only privileges, select File > Save As to save the metaoutline to a different name. See “Saving a Metaoutline to a Different Name” on page 2-27. Also see the *Hyperion Integration Server System Administrator’s Guide* for information on unlocking metaoutlines.

Note: Unless you selected Exclusive Access when you opened the metaoutline, you cannot save the metaoutline if other users are reading it. Save it later or use File > Save As to save the metaoutline to a different name. See “Saving a Metaoutline to a Different Name” on page 2-27.

Saving a Metaoutline Manually

If the metaoutline that you want to save has been previously saved and you do not want to give the metaoutline a different name, select File > Save.

OLAP Integration Server writes the changes to the OLAP Metadata Catalog to which you are connected.

If you have not previously saved the metaoutline or if you want to save the metaoutline to a different name, you must use the Save New Metaoutline dialog box.

- To use the **Save New Metaoutline** dialog box to save an outline manually:
 1. Perform one of the following tasks:
 - If you are saving a metaoutline for the first time, select File > Save.
 - If you are saving a previously saved metaoutline to a new name, select File > Save As.

The **Save New Metaoutline** dialog box, as shown in Figure 2-8, is displayed.



Figure 2-8: Saving a Metaoutline

2. In the **Metaoutline name** text box, enter the name of the metaoutline.
See “About Naming” on page 3-8.

3. From the **Security** drop-down list box, select one of the following options:

Table 2-4: Read/Write Settings

Option	Permitted Activity
Allow read/write access for other users	Other users can read and write to the metaoutline. This setting is the default.
Allow read access for other users	Other users can read the metaoutline but not write to the metaoutline; that is, another user cannot save changes to the metaoutline.
Disallow all access for other users	Other users can neither read nor write to the metaoutline. Only you can read or write to the metaoutline.

Note: To change the permissions on a metaoutline after you create it, you must use the tools provided with the relational database that contains the OLAP Metadata Catalog or select File > Save As to save the metaoutline to a different name. See the *Hyperion Integration Server System Administrator's Guide*.

4. If you want to describe the metaoutline, in the **Description** text box, enter a brief description.

The description can include up to 255 characters. Hyperion Integration Server displays the description in the following places:

- The **Save New Metaoutline** dialog box, when you select File > Save As (see “Saving a Metaoutline to a Different Name” on page 2-27)
- The **OLAP Metaoutline Properties** dialog box (You can view or change the metaoutline description in this dialog box at any time.)
- The **Existing** tab of the **Welcome** dialog box (see “Working with Metaoutlines” on page 2-3)
- The **Recent** tab of the **Welcome** dialog box (see “Working with Metaoutlines” on page 2-3)

5. Click OK.

If the metaoutline is invalid, Hyperion Integration Server displays an error message. You can save the metaoutline without correcting the error, but you cannot use an invalid metaoutline to load member levels or data into a Hyperion Essbase database.

Note: If you attempt to save a metaoutline that another user currently has open, a dialog box informs you that the metaoutline is currently locked or open. You must wait to save until all other users close the metaoutline.

Changing Read/Write Access

At any time after you save a metaoutline, you can change the access rights that you originally assigned by selecting File > Save As.

► To change read/write access to a metaoutline:

1. Select File > Save As.

The **Save New Metaoutline** dialog box is displayed.

2. In the **Security** drop-down list box, change the user access defined for the model.

3. Click **OK**.

Saving While Closing

If you try to close a metaoutline without saving it, a dialog box prompts you to save your changes.

Click Yes to save the metaoutline with changes or No to close the metaoutline without saving changes. See “Saving a Metaoutline Manually” on page 2-23.

CAUTION: If the metaoutline you are closing is a new metaoutline that has never been saved, you will lose it if you click No.

Saving a Metaoutline to a Different Name

Save a metaoutline to a different name if you want to accomplish any of the following tasks:


- Change the name of a metaoutline. (The original metaoutline retains the original name.)
- Change the access permissions of the metaoutline.
- Make a copy of the metaoutline. You cannot copy a metaoutline by using the Copy function that the operating system provides because the metaoutline resides in OLAP Metadata Catalog. Instead, you need to save the metaoutline to a different name.

To save a metaoutline to a different name, select File > Save As. Hyperion Integration Server displays the Save New Metaoutline dialog box. For more information, see “Saving a Metaoutline Manually” on page 2-23.

For information about naming metaoutlines, see “About Naming” on page 3-8.

Printing Metaoutlines

You can print the metaoutline contained in the right frame of the OLAP Metaoutline main window to use as a hard-copy reference or to retain on file. The OLAP model information in the left frame of the window does not print when you print the metaoutline. To print the OLAP model, see the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

- To print a metaoutline:
1. Select File > Print or click the **Print** toolbar button, , to open the **Standard Print** dialog box.
 2. Select the settings that you want.
 3. Click **OK**.

Deleting Metaoutlines

When you delete a metaoutline, Hyperion Integration Server removes the metaoutline from OLAP Metadata Catalog.

- To delete one or more metaoutlines:
1. Make sure that you are connected to the OLAP Metadata Catalog where the metaoutline is stored.

If you are connected, the lower-right corner of the OLAP Metaoutline main window contains the appropriate OLAP Metadata Catalog name (for example, TBC_MD in the sample TBC database).

If you are not connected, select Connections > OLAP Metadata Catalog > Connect and log on to the correct OLAP Metadata Catalog.
 2. Select File > Delete to display the **Delete an OLAP Model/Metaoutline** dialog box, shown in Figure 2-9.



Figure 2-9: Deleting a Metaoutline

3. Select the metaoutline to delete, and click **Delete**.

Note: To select multiple metaoutlines to delete, select one metaoutline, hold the **Ctrl** key, and select each additional metaoutline.

Hyperion Integration Server displays a dialog box that prompts you to confirm that you want to delete the metaoutline.

Note: If you delete a metaoutline that you used to build a Hyperion Essbase outline, you can no longer drill through on the Hyperion Essbase outline. See Chapter 9, “Creating Drill-Through Reports.”

4. Click **Yes**.

CAUTION: Use caution when deleting. You cannot undo the deletion of a metaoutline.

OLAP Integration Server deletes the selected metaoutline from OLAP Metadata Catalog.

Note: If you attempt to delete a metaoutline that you or another user currently has open, a dialog box informs you that the metaoutline is currently locked or open. If you have it open, close the metaoutline and select **Connections > OLAP Metadata Catalog > Delete Locks**. If other users have the metaoutline open, you must wait until they close it.

Creating and Working with Metaoutlines

Adding Dimensions, Member Levels, and Measures to a Metaoutline

This chapter begins by defining what the terms *dimension*, *member level*, and *measure* mean in the context of a metaoutline and by explaining how dimensions, member levels, and measures map to a Hyperion Essbase outline. Next, the chapter steps you through the process of creating, renaming, and deleting dimensions, member levels, and measures. Finally, the chapter discusses how you can preview the structures of dimensions, member levels, and measures before you create a Hyperion Essbase outline.

Note: For a tutorial on creating a metaoutline that contains attribute dimensions, see Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions.” The relational data source, OLAP Metadata Catalog, OLAP model, metaoutline, and data on the example screens in this appendix are based on the sample TBC application provided with Hyperion Integration Server.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 3-2
- “About Dimensions, Member Levels, and Measures” on page 3-3
- “Creating Dimensions, Member Levels, and Measures” on page 3-11
- “Creating Attribute Dimensions and Member Levels” on page 3-24
- “Renaming Dimensions, Member Levels, and Measures” on page 3-44
- “Deleting Dimensions, Member Levels, and Measures” on page 3-49

- “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50
- “Viewing Defined Measures” on page 3-52

Note: For information on setting properties for dimensions, member levels, and measures, see Chapter 4, “Setting Load Properties,” Chapter 5, “Working with Filters,” and Chapter 6, “Setting Hyperion Essbase Properties.” For more information on attribute dimensions, see Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions.”

Understanding the Metaoutline Workflow

Figure 3-1 illustrates the workflow for creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following procedures:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and member levels from a metaoutline into a Hyperion Essbase outline
- Loading data that is specified in a metaoutline into a Hyperion Essbase database

Chapter 3 focuses on adding dimensions, member levels, and measures to a metaoutline.

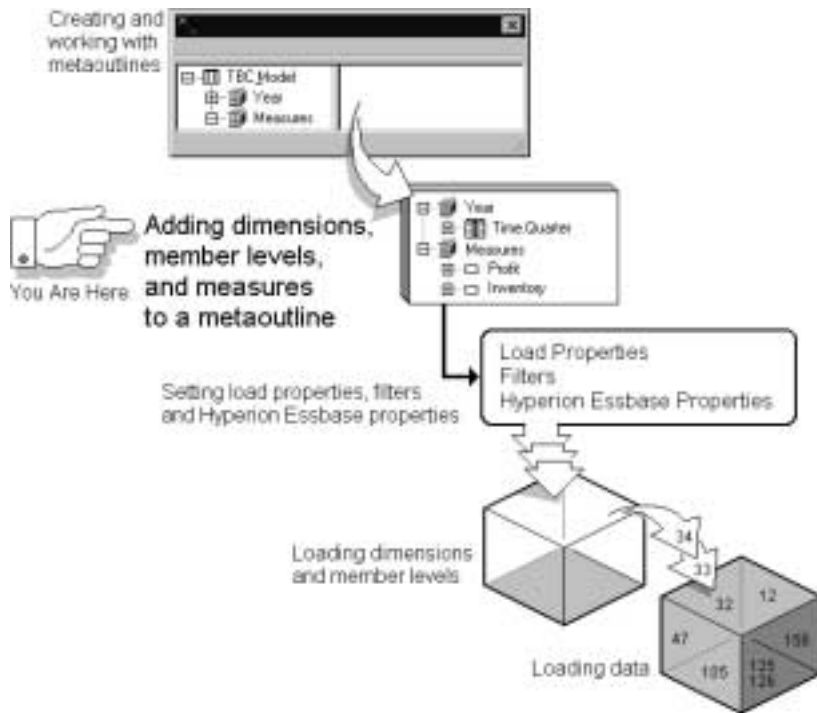


Figure 3-1: Metaoutline Workflow

About Dimensions, Member Levels, and Measures

This topic describes how dimensions, member levels, and measures of a metaoutline map to dimensions, members, and data values of a related Hyperion Essbase outline. The topic contains the following subtopics:

- “About Dimensions” on page 3-4
- “About Member Levels and Members” on page 3-5
- “About Measures” on page 3-6

Adding Dimensions, Member Levels, and Measures to a Metaoutline

- “About User-Defined Dimensions and Members” on page 3-7
- “About Attribute Dimensions and Members” on page 3-8
- “About Naming” on page 3-8

About Dimensions

In a metaoutline, the categories that represent the highest level of organization are called *dimensions*. A metaoutline dimension can have one or more member levels. Each dimension in a metaoutline corresponds to a dimension in a related Hyperion Essbase outline. For example, if a metaoutline has a Product dimension, the Hyperion Essbase outline has a Product dimension. Every dimension that you create in a metaoutline becomes a dimension in the related Hyperion Essbase outline.

TBC Metaoutline, a sample metaoutline provided with Hyperion Integration Server, contains the following dimensions: Year, Measures, Scenario, Market, and Product. It also contains the following attribute dimensions: CAFFEINATED, OUNCES, PKGTYPE, POPULATION, and INTRODATE.

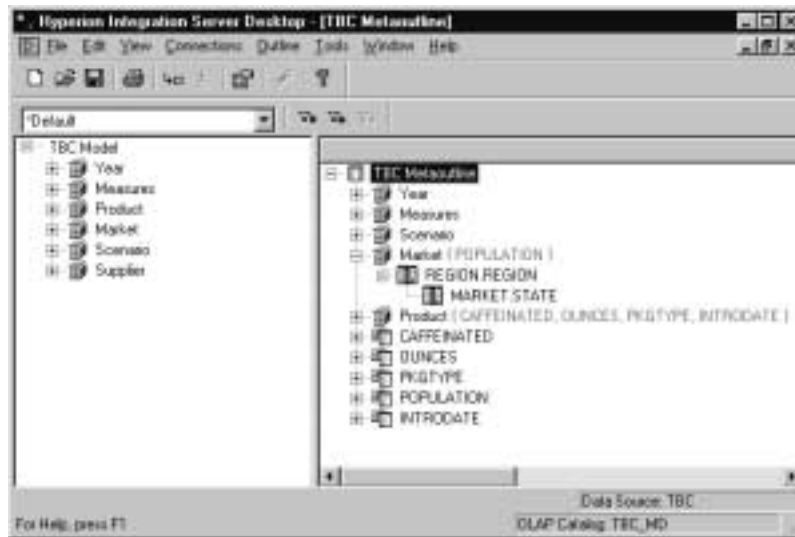


Figure 3-2: Dimensions in TBC Metaoutline

About Member Levels and Members

A *member level* in a metaoutline is based upon a column of the related OLAP model and is representative of the levels that are contained within a hierarchy of the related Hyperion Essbase outline. A *member* is a relational value within an OLAP model column, a value within a metaoutline member level, and a value within a Hyperion Essbase dimension. For example, in reference to the Market dimension, the REGION and STATE columns of the OLAP model become the REGION and STATE member levels of the metaoutline. The members of the member levels REGION and STATE then become members of the Hyperion Essbase outline (Central, East, South, West, Colorado, Illinois, and so on).

For example, in the following hierarchy created in an OLAP model, the Market dimension contains columns that become the metaoutline member levels REGION.REGION and MARKET.STATE.

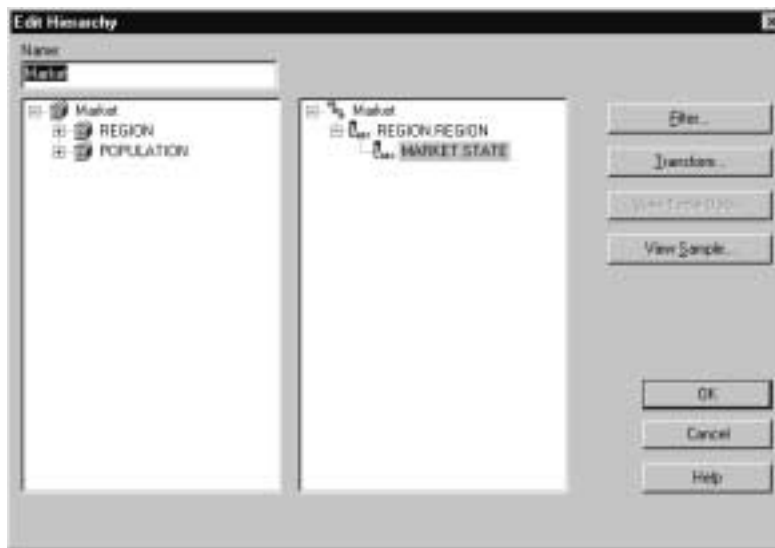


Figure 3-3: Example Hierarchy in an OLAP Model

Adding Dimensions, Member Levels, and Measures to a Metaoutline

In the following sample Hyperion Essbase outline, the Central, East, South, and West members of the Market dimension are based on values from the REGION column of the OLAP model and thus on values from the relational data source:

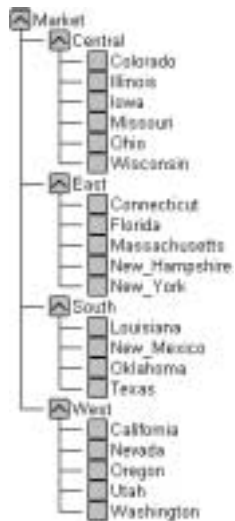


Figure 3-4: Example Hyperion Essbase Outline

Note: A measure dimension contains actual members, even though it may contain levels of the various members.

About Measures

The Hyperion Essbase database contains data values (measures) for each dimension intersection. Measures include items such as SALES and COGS (cost of goods sold). If you do not define at least one measure in a metaoutline, OLAP Integration Server cannot validate the metaoutline or use the metaoutline to build a Hyperion Essbase outline.

The Measures dimension in TBC Metaoutline contains the measures SALES and COGS.

About User-Defined Dimensions and Members

You can create user-defined dimensions and members in a metaoutline. User-defined dimensions and members correspond to dimensions and members in a related Hyperion Essbase outline. For example, if you have a user-defined TestColas member in a metaoutline, you have a TestColas member in the Hyperion Essbase outline.

User-defined dimensions and members, however, are *not* associated with an OLAP model. Create a user-defined dimension or member when you want to add a dimension or member that is not provided in the relevant OLAP model.

For example, a Hyperion Essbase database enables you to create a TestColas dimension. The TestColas dimension contains data for two types of colas: ExtraCaffeineCola and VerySweetCola. The relational data source contains the data for ExtraCaffeineCola. A worksheet created by the budget department contains the projected data for VerySweetCola.

So how do you combine the data for the TestColas in one dimension in the Hyperion Essbase database? There are no TestColas, ExtraCaffeineCola, or VerySweetCola dimensions or members in the OLAP model. Therefore, you must create the TestCola user-defined dimension and create ExtraCaffeineCola and VerySweetCola as user-defined members.

- To create ExtraCaffeineCola and VerySweetCola as user-defined members:
 1. In the metaoutline, create a TestColas dimension.
See “Creating Dimensions Associated with an OLAP Model” on page 3-12.
 2. To the TestColas dimension, add the ExtraCaffeineCola and VerySweetCola user-defined members.
See “Creating User-Defined Dimensions and Members” on page 3-18.
 3. Because ExtraCaffeineCola receives its data from the data source, tag ExtraCaffeineCola as the default load member.
See “Defining the Load Member for a User-Defined Dimension” on page 3-21.
 4. Load members and data from the budget worksheet into the Budget member of the Hyperion Essbase outline.
See the *Hyperion Essbase OLAP Server Database Administrator’s Guide* for more information about loading members and data, including loading data from worksheets.

About Attribute Dimensions and Members

Attribute dimensions created in the metaoutline are based upon attribute-enabled columns of the OLAP model. Attribute dimensions allow you to retrieve and analyze data not only from the perspective of dimensions and members, but also in terms of characteristics, or attributes, of those dimensions.

For example, the Product dimension contains various products, such as Cola and Decaffeinated Cola. Using attribute dimensions, you can filter your analysis by a Boolean attribute of the products: caffeinated true and false. Similarly, the Market dimension contains information on sales in various states. An attribute dimension of a state could be the population of the state; some soft drinks might sell better in more populated areas. The attribute dimension allows you to view results by specific attribute criteria.

There are five predesigned calculations available for each attribute member. These calculations are sums, counts, averages, minimum, and maximum values.

For more information on creating and using attributes, see “Creating Attribute Dimensions and Member Levels” on page 3-24, and Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions.”.

Refer to the *Hyperion Integration Server Desktop OLAP Model User’s Guide* for information about enabling OLAP model columns to become attributes. Refer to the *Hyperion Essbase OLAP Server Database Administrator’s Guide* for more information on calculations in attribute dimensions.

About Naming

When naming metaoutlines, dimensions, member levels, aliases, user-defined members, attributes, and metaoutline filters in the metaoutline, follow these rules:

- Do not use more than 80 characters.
- Use mixed uppercase and lowercase characters. Names are not case-sensitive in Hyperion Essbase unless there is a check mark next to the Settings > Case Sensitive Members menu item in Hyperion Essbase.

When naming dimensions, member levels, aliases, user-defined members, attributes, and metaoutline filters, observe the following rules:

- Do not use quotation marks (") or tabs anywhere in a name.

- Do not use any of the following characters in a name:

@	(at sign)	()	(parentheses)
\	(backslash)	.	(period)
{ }	(braces)	+	(plus sign)
,	(comma)	'	(single quotation mark)
-	(dash, hyphen, or minus sign)	_	(underscore)
=	(equal sign)		(vertical bar)
<	(less than sign)	/	(slash)
*	(asterisk)	"	(double quotation mark)
	tabs		

- Do not place spaces at the beginning or end of a name. (Hyperion Essbase ignores spaces at the beginning or end of a name.)
- Do not use any of the following words as a dimension or member level name:

- \$\$\$UNIVERSE\$\$\$

- #MISSING or #MI

- Calculation script commands, operators, and keywords

For a list of commands, see the online *Technical Reference* in your Hyperion\Essbase\DOCS directory.

- Report script commands

For a list of commands, see the online *Technical Reference* in your Hyperion\Essbase\DOCS directory.

- Function names and function arguments

For a list of functions, see the online *Technical Reference* in your Hyperion\Essbase\DOCS directory.

- The name of an existing dimension, a member level (unless the member is shared), an alias, a combination alias, or a generation level. User-defined dimensions and members cannot use existing member names.

Note: If you tag members as Dynamic Time Series, do not use the associated generation names: History, Year, Season, Period, Quarter, Month, Week, or Day. See the *Hyperion Essbase OLAP Server Database Administrator's Guide* for more information.

Adding Dimensions, Member Levels, and Measures to a Metaoutline

When you use a member level name in a calculation script, report script, or formula, you must enclose the member level name in quotation marks (") in the following situations:

- The name contains one or more numerals at the beginning of the name (for example, 100-Blue).
- The name contains spaces anywhere in the name.
- The name contains any of the following characters:

*	(asterisk)	!	(exclamation point)
@	(at sign)	>	(greater than sign)
{ }	(braces)	<	(less than sign)
[]	(brackets)	()	(parentheses)
:	(colon)	%	(percent sign)
,	(comma)	+	(plus sign)
-	(dash, hyphen, or minus sign)	;	(semicolon)
=	(equal sign)	/	(slash)
\	(backslash)		

- The name starts with an ampersand (&).

In calculation scripts and formulas, if the following words are used as member level names, you must enclose these words in quotation marks (""):

AND	ELSEIF	IF	OR
BEGIN	END	MACRO	RANGE
DOUBLE	ENDIF	MEMBER	STRING
ELSE	FUNCTION	NOT	THEN

Creating Dimensions, Member Levels, and Measures

As shown in Figure 3-5, the OLAP model in the left frame of the OLAP Metaoutline main window can contain dimensions, hierarchies, columns, and measures. These are the OLAP model objects that you can drag to the right frame of the OLAP Metaoutline main window to create dimensions, member levels, and measures in the metaoutline.

3

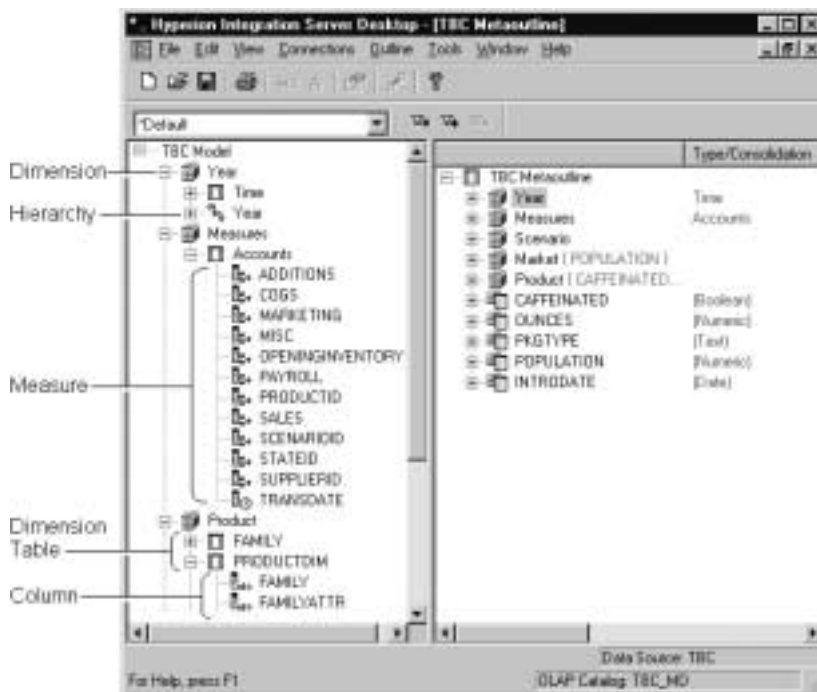


Figure 3-5: Objects in the OLAP Model

This topic contains the following subtopics:

- “Creating Dimensions Associated with an OLAP Model” on page 3-12
- “Creating Member Levels Associated with an OLAP Model” on page 3-15
- “Creating Recursive Dimensions” on page 3-17
- “Creating User-Defined Dimensions and Members” on page 3-18
- “Defining the Load Member for a User-Defined Dimension” on page 3-21
- “Creating a Measure Dimension” on page 3-23
- “Creating Attribute Dimensions and Member Levels” on page 3-24
- “Comparison of UDAs and Attributes” on page 3-25

Creating Dimensions Associated with an OLAP Model

To use an OLAP model object to create a dimension in a metaoutline, drag a dimension, hierarchy, column, or measure from the left frame of the OLAP Metaoutline main window to the right frame of the OLAP Metaoutline main window.

When you drag an object to the right frame, the object creates a new dimension at the bottom of the metaoutline. The new dimension uses the OLAP model name of the dimension in which the OLAP model object is located.

Adding Dimensions, Member Levels, and Measures to a Metaoutline

For example, if you drag the SALES measure to the right frame, the SALES measure creates the Measures dimension with the SALES measure inserted under it.



Figure 3-6: Using a Measure to Create a Dimension

Table 3-1 describes creating members and dimensions by dragging various OLAP model objects.

Table 3-1: Creating Dimensions by Dragging OLAP Model Objects








OLAP Model Object that You Drag	Metaoutline Dimension that You Create	Example
Dimension, 	New dimension.	The Product dimension of the left frame creates the Product dimension in the right frame.
Hierarchy, 	New dimension that contains the hierarchy.	The Quarter, Month hierarchy of the left frame creates in the right frame the Year dimension with the Quarter and Month member levels arranged hierarchically.

Table 3-1: Creating Dimensions by Dragging OLAP Model Objects (Continued)

OLAP Model Object that You Drag	Metaoutline Dimension that You Create	Example
Text column, 	New dimension that contains a member level that is based on the Text column.	The FAMILY column of the left frame creates in the right frame the Product dimension with the FAMILY member level.
Numeric column, 	New dimension that contains a member level that is based on the Numeric column.	The REGIONID column of the left frame creates in the right frame the Market dimension with the REGIONID member level.
Measure, 	New dimension that is tagged as Accounts and that contains a measure based on the accounts value.	The ADDITIONS column of the left frame creates in the right frame the Accounts dimension with the ADDITIONS member level.
Attribute, 	New attribute dimension containing new attribute member, new base dimension if necessary, and association of base and attribute dimension	The POPULATION.POPGROUP attribute-enabled column creates the new attribute dimension POPGROUP with the member POPULATION.POPGROUP. It creates the Market dimension and associates POPULATION.POPGROUP to Market.
Time column, 	New dimension that is tagged as Time and that contains a member level based on the Time column.	The Month column in the left frame creates in the right frame the Year dimension with the Month member level.

For information on selecting multiple dimensions, hierarchies, columns, and measures in the left frame, see “Selecting Multiple Objects in the Left Frame” on page 2-15.

After a dimension is created in the right frame, you can drag the dimension to another location within the metaoutline and rename the dimension.

See “Renaming Dimensions, Member Levels, and Measures” on page 3-44.

CAUTION: You must define at least one measure in each metaoutline. If the metaoutline does not contain a measure, you cannot use the metaoutline to build a Hyperion Essbase outline. See “Creating a Measure Dimension” on page 3-23.

Creating Member Levels Associated with an OLAP Model

To create a member level from an OLAP model object, drag a column, hierarchy, or measure from the left frame of the OLAP Metaoutline main window to the right frame of the OLAP Metaoutline main window.

For example, if you drag the predefined Year, Quarter, Month hierarchy into the right frame, the hierarchy creates a Year dimension with the hierarchy inserted under it, as illustrated in Figure 3-7.

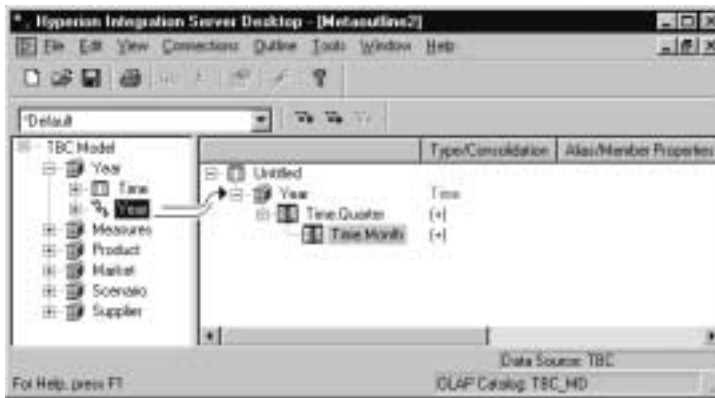


Figure 3-7: Using a Hierarchy to Create a Dimension

Table 3-2 describes dragging columns, hierarchies and measures to create metaoutline structures.

Table 3-2: Creating Metaoutline Structures

Location in Metaoutline to Which You Drag a Column, Hierarchy, or Measure	Metaoutline Structure that You Create	Example
Blank space	<p>New dimension with the member level, the hierarchy of member levels, or the measure inserted directly under the new dimension.</p> <p>This action creates the new dimension at the bottom of the metaoutline and gives the new dimension the name of the dimension in which the OLAP model object is located.</p>	<p>Dragging the ADDITIONS column onto a blank space creates a Measures dimension with an ADDITIONS member level. You can rename the Measures dimension as Accounts.</p>
Dimension	<p>New member level, hierarchy of new member levels, or measure directly under the selected dimension.</p> <p>This action creates the new member level, hierarchy of new member levels, or new measure at the bottom of the selected dimension and gives the new object or measure the name that the object had in the OLAP model.</p>	<p>Dragging the FAMILYID column onto the Product dimension creates a FAMILY.FAMILYID member level under the Product dimension.</p>
Member level	<p>New member level, hierarchy of new member levels, or new measure directly under the selected member level.</p> <p>This action creates the new member level, hierarchy of new member levels, or new measure at the bottom of the selected member level and gives the new object the name that the object had in the OLAP model.</p>	<p>Dragging the PRODUCTDIM.SKU column onto the PRODUCTDIM.FAMILY member level creates a PRODUCTDIM.SKU member level under the PRODUCTDIM.FAMILY member level.</p>

For information on selecting multiple dimensions, hierarchies, columns, and measures in the left frame, see “Selecting Multiple Objects in the Left Frame” on page 2-15.

After a new object is created in the right frame, you can, as necessary, perform any of the following tasks:

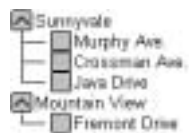
- Drag the newly-created member level or measure to the appropriate location in the dimension.
- Rename the newly-created member level or measure.
See “Renaming Dimensions, Member Levels, and Measures” on page 3-44.
- Delete unwanted member levels of a hierarchy, from the leaf level up, by selecting an unwanted member level and pressing Delete.

Creating Recursive Dimensions

You can create dimensions that contain member levels that are based on the contents of the two columns of a parent-child relationship. These dimensions are called *recursive dimensions*. For example, consider the following parent-child data source:

CHILD	PARENT
Murphy Ave.	Sunnyvale
Crossman Ave.	Sunnyvale
Fremont Drive	Mountain View
Java Drive	Sunnyvale

You can use the data source to create the following Hyperion Essbase outline:



When creating a recursive dimension, follow these rules:

- Do not set transformations on member levels in recursive dimensions.
- If you want to assign aliases or UDAs to recursive member levels, you drag from the left frame to the right frame the column to which the alias or UDA column applies. For example, if the alias column applies to the child column, use the child column in the metaoutline. If you are not sure to which column (parent or child) the alias or UDA column applies, see the information on creating aliases in the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.

- If you create a filter on the parent member level of a recursive dimension, OLAP Integration Server applies the filter only to values of the parent member level column. OLAP Integration Server returns both the parent member level that satisfies the requirements of the filter and the immediate child of that parent.

For example, if you apply a filter to retrieve only East on the MARKET.PARENT column of TBC Metaoutline, OLAP Integration Server retrieves only the rows that have East in the parent column, such as the rows that contain East, Florida and East, New York.

- If you create a filter on the child member level of a recursive dimension, Hyperion Integration Server applies the filter only to values in the child column. OLAP Integration Server returns both the child value that satisfies the filter condition and the immediate parent of that child. If a child member level is shared by multiple parents, OLAP Integration Server returns all immediate parents of the child.

For example, if you apply a filter to retrieve only East on the MARKET_CHILD column of TBC Metaoutline, OLAP Integration Server retrieves only the rows that have East in the child column, such as the row that contains USA, East. If East is shared by multiple parents, OLAP Integration Server also returns all parents; for example, England, East or Canada, East.

➤ To create a recursive dimension:

1. Set up the parent-child data source for recursive dimensions.

See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

2. Drag either the parent column or the child column (but not both) to the right frame of the OLAP Metaoutline main window.

OLAP Integration Server creates the recursive hierarchy during the member load.

Creating User-Defined Dimensions and Members

User-defined dimensions and members are not associated with an OLAP model. You can create a user-defined dimension as a child of the metaoutline or as a sibling of an existing dimension. Likewise, you can create a user-defined member as a child of a dimension (either a user-defined dimension or a dimension associated with an OLAP model) or as a sibling of an existing user-defined



member. You *cannot* create a user-defined member as a child of a database-defined member level. See “About User-Defined Dimensions and Members” on page 3-7.

This topic contains the following subtopics:

- “Creating as Children” on page 3-19
- “Creating as Siblings” on page 3-20

Creating as Children

► To create a user-defined dimension as a child of a metaoutline or a user-defined member as a child of a dimension:

1. In the right frame of the OLAP Metaoutline main window, select one of the following objects:
 - To create a dimension, select the metaoutline name; for example, TBC Metaoutline,  TBC Metaoutline .
 - To create a member, select a dimension name; for example, TESTCOLAS,  TestColas .

Note: You *cannot* create a user-defined member as a child of a database-defined member level.

To name the new dimension or member, select Edit > Add Member as Child,

or click the **Add Child** button,  . The **Enter the New dimension’s Name** dialog box is displayed.



Figure 3-8: Naming a New Dimension

2. Enter the new dimension or member name; for example, ExtraCaffeineCola. See “About Naming” on page 3-8.

3. Click **OK**.



If you create a dimension, it is displayed as the first dimension in the metaoutline. If you create a member, it is displayed as the first member in the dimension.

4. If necessary, drag the newly-created dimension or member to the appropriate location in the metaoutline.


Creating as Siblings

► To create a user-defined dimension as a sibling of an existing dimension or a user-defined member as a sibling of a user-defined member:

1. In the right frame of the OLAP Metaoutline main window, perform one of the following tasks:

- Select the dimension after which you want to add the new dimension; for example, Year,  Year .
- Select the user-defined member after which you want to add the new member; for example, ExtraCaffeineCola,  ExtraCaffeineCola .

Note: You *cannot* create a user-defined member as a child of a database-defined member level.

2. Select Edit > Add Member as Sibling, or click the **Add Sibling** button,  .

The **Enter the New Dimension's Name** dialog box is displayed.



Figure 3-9: Naming a New Dimension

3. Enter the new dimension or member name; for example, **Scenario**.
See “About Naming” on page 3-8.
4. Click **OK**.
If you create a dimension, the new dimension is placed after the selected dimension. If you create a member, the new member is placed after the selected member.
5. If necessary, drag the newly-created dimension or member to the appropriate location.

Defining the Load Member for a User-Defined Dimension

A user-defined dimension is not connected to the related OLAP model. Because the dimension is not connected to the OLAP model, OLAP Integration Server does not know into which member to load data. Therefore, you must select a member in the user-defined dimension into which OLAP Integration Server loads data. The selected member is called the *load member*. You must select one load member for each user-defined dimension.

- To set the load member in a user-defined dimension:
 1. In the right frame of the OLAP Metaoutline main window, select the metaoutline name; for example, TBC Metaoutline.
 2. Right-click the metaoutline name, and, from the pop-up menu, select Properties.

3. Select the **User Defined Dimensions** tab.



Figure 3-10: Viewing User-Defined Dimensions

4. Select the user-defined dimension for which you want to define the default load member; for example, TestColas.
5. Click **Edit**.

The **Select Default Load Member** dialog box, which lists all user-defined members of the selected dimension, is displayed.



Figure 3-11: Setting the Default Load Member

6. Select the member to use as the default load member; for example, ExtraCaffeineCola.

Note: You can also set the default load member by right-clicking a member of a user-defined dimension and selecting Default Load Member from the pop-up menu. When you select Default Load Member, a check mark is displayed next to Default Load Member in the pop-up menu.

7. Click **OK**.

OLAP Integration Server loads data from the relational data source into the default load member the next time you perform a data load.

Creating a Measure Dimension

If a metaoutline contains more than one measure (data value), you must define a measure dimension and add all measures to that dimension. All measures that you can use in the metaoutline are defined in the OLAP model.

You can create a measure dimension by using the OLAP model dimension that is tagged as accounts or by creating a new dimension and tagging it as measure. See “About Measures” on page 3-6.



All measures must be defined at level 0 (the leaf level) of the metaoutline. If you want to group measures under a parent, create the parent as a user-defined member. See “Creating User-Defined Dimensions and Members” on page 3-18.

- To create a measure dimension from the OLAP model dimension tagged as accounts:
 1. Drag the dimension tagged as accounts from the OLAP model to the metaoutline.

The selected dimension creates in the metaoutline a measure dimension tagged as accounts.
 2. Drag the appropriate measures, such as COGS, from the OLAP model to the metaoutline dimension tagged as accounts.

- To create a measure dimension without using the OLAP model dimension tagged as accounts:
 1. Create the dimension that you want to use as the measure dimension.
 2. Select the dimension.
 3. Right-click the dimension, and, from the pop-up menu, select Database Measure Dimension.
 4. Drag the appropriate measures, such as COGS, from the OLAP model to the measure dimension.

Note: If the dimension is not tagged as accounts in the OLAP model, the measures that you add do not use the standard measure

icon, . Instead, they use the column icon, . Regardless of the button, however, Hyperion Integration Server treats the members as measures during member and data loads.

Creating Attribute Dimensions and Member Levels

Attribute dimensions in the metaoutline are based on attribute-enabled columns in the OLAP model. After an attribute dimension and member have been created, you can define attribute properties, such as Boolean and numeric ranges, that enable you to view business data in finer detail than would otherwise be easily available.

For a tutorial on creating attribute members and dimensions, see Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions.” For a comparison of attributes and user-defined attributes (UDAs), see “Comparison of UDAs and Attributes” on page 3-25.

This topic contains the following subtopics:

- “Comparison of UDAs and Attributes” on page 3-25
- “Creating Attribute Dimensions from OLAP Model Dimensions” on page 3-26
- “Naming Attribute Members” on page 3-27
- “Setting the Member Names Format” on page 3-27
- “Setting the Calculations Name Format” on page 3-28
- “Specifying Attribute Boolean Names and Date Formats” on page 3-31
- “Specifying Numeric Boundaries for Attributes” on page 3-33

- “Renaming and Selecting a Data Type for an Attribute Member” on page 3-35
- “Choosing a Data Type for an Attribute Dimension” on page 3-36
- “Setting Base Dimension Levels for an Attribute” on page 3-37
- “Setting Boolean Parameters for an Attribute” on page 3-39
- “Setting Numeric Ranges for Attribute Members” on page 3-42
- “Seeing Attribute Results in Spreadsheets” on page 3-44

Comparison of UDAs and Attributes

Attributes and UDAs both enable analysis based on characteristics of the data. Attributes provide more capability than UDAs. However, attributes can only be associated with sparse dimensions, whereas UDAs can be used on both sparse and dense dimensions.

Both attributes and UDAs can retrieve consolidated totals. In attributes, the built-in calculation function makes this ability simple to implement. To retrieve totals in UDAs, you must build a calculation script or special-purpose commands. Similarly, it is easy to categorize attributes in a hierarchy and retrieve totals by higher levels in the hierarchy.

You can use Boolean operators, AND and OR, with attribute and UDA values to refine a query. For example, you can select decaffeinated drinks from the 100 product group. Because attributes can have date, text, Boolean, and numeric type, you can use appropriate operators and functions to work with and display attribute data.

You can group numeric attributes into ranges of values (population size, for example) and enable the dimension building process to associate the base member with the appropriate range automatically. For example, you can group sales in various regions based on ranges of their populations: less than 3 million, between 3 and 6 million, and so on.

For more information on User-Defined Attributes (UDAs), see “Specifying UDAs” on page 6-25 and the *Hyperion Essbase OLAP Server Database Administrator’s Guide*. For more information on using attributes in calculations, see “Using Attribute Calculations” on page A-14 and the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.

Creating Attribute Dimensions from OLAP Model Dimensions

To create an attribute dimension and member level in the metaoutline, you must have an attribute-enabled column in the OLAP model.

Note: An attribute-enabled column in the left frame must be dragged to an empty space in the right frame; it cannot be placed within another dimension.

- To create an attribute dimension from an OLAP model dimension:
 - Drag an attribute-enabled column (such as PRODUCT.CAFFEINATED) in the left frame to the right frame. In Figure 3-12, PRODUCT.CAFFEINATED was dragged directly under the metaoutline icon.

This action creates a dimension (for example, the Product dimension), an attribute dimension (for example, CAFFEINATED) and an attribute member (for example, PRODUCT.CAFFEINATED). If the Product dimension already exists in the metaoutline, it remains unchanged. In either case, dragging an attribute member to the right frame creates an association between the attribute and the base dimension. The new attribute dimension is positioned under the metaoutline icon, as shown in Figure 3-12.

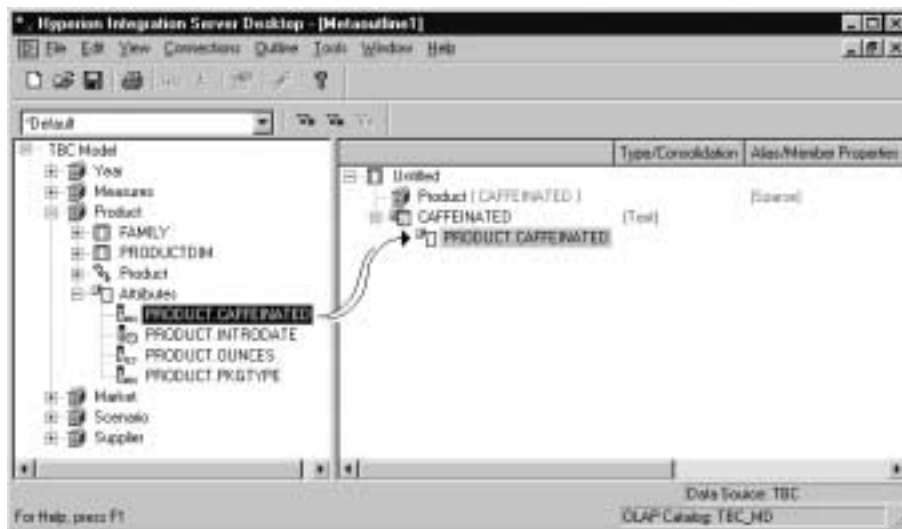


Figure 3-12: Creating an Attribute Dimension and Member

Naming Attribute Members

- To determine the format for attribute member names in a metaoutline by using the Attribute Member Names dialog box:
 1. In the right frame of the OLAP metaoutline main window, select the metaoutline icon at the top of the frame.
 2. Right click on the metaoutline icon, and select Attribute Member Names from the pop-up menu.

The **Attribute Member Names** dialog box is displayed. This dialog box has four tabs: **Member Name Format**, **Calculations**, **Boolean and Date**, and **Numeric Boundaries**, as shown in Figure 3-13.



Figure 3-13: Attribute Member Names Dialog Box

Setting the Member Names Format

You can ensure that an attribute name is unique by assigning a prefix or suffix. Note that a metaoutline does not display the full attribute member name after you have assigned a prefix or suffix identifier. You can view the full attribute names when you retrieve information; for example, full attribute names are displayed when you view a spreadsheet.

- To set the format for attribute member names:
1. Right-click the metaoutline icon, and select **Attribute Member Names** from the pop-up menu.
The **Attribute Member Names** dialog box is displayed.
 2. Select the **Member Name Format** tab (Figure 3-13).
 3. Select a **Value** such as **Parent** to be used as a prefix or suffix.
 4. Select a **Separator** to be used with the prefix or suffix.
The default separator is an underscore (_).
 5. Click **Prefix** or **Suffix** to define how the selected value is to be used.
 6. Click **OK**.

The **Sample** area displays an example attribute name based on the selections made in this dialog box.

Setting the Calculations Name Format

An attribute calculations member name cannot be the same as a member name. For example, if another member name in the metaoutline is **Sum**, then you must enter a new name for the **Sum** member name in the **Calculations** tab. You can change the names for the attribute calculations in the **Calculations** tab of the **Attribute Member Names** dialog box (Figure 3-14).

Attribute calculations have the following properties:

- **System-defined:** When you create an attribute dimension in a metaoutline, the resulting reports and spreadsheets have five predefined calculations available for all attribute members.
- **Dynamic Calc:** All attribute calculations are calculated when a user requests the calculation, and then the calculation is discarded. You cannot store the calculated data in a database.
- **Displayed in Reports:** Attribute calculation results are not displayed in the metaoutline. The calculation results are available only in spreadsheets and reports.

The following five calculations are system-defined for attribute dimensions:

- **Sum:** Sum calculates a sum, but it does not always calculate a simple sum. Sum totals members based on their consolidation property or formula. For example, you can Sum the Profit% for 12-Ounce drinks, and the Sum will be based on the Profit% formula, which is profit divided by sales, with the resulting ratio multiplied by 100.
- **Count:** Calculates the number of members with the specified attribute or combination of attributes. Count includes only those members that have data blocks in existence.
- **Min:** Calculates the minimum data value for a specified attribute or combination of attributes.
- **Max:** Calculates the maximum data value for a specified attribute or combination of attributes.
- **Average:** Calculates a mechanical mean (average) of the values for a specified attribute or combination of attributes. The average includes only those members that have data blocks in existence (sum divided by count).

➤ To rename any of the system-defined calculations:

1. Right-click the metaoutline icon, and select Attribute Member Names from the pop-up menu.

The **Attribute Member Names** dialog box is displayed.

2. Select the **Calculations** tab as shown in Figure 3-14.

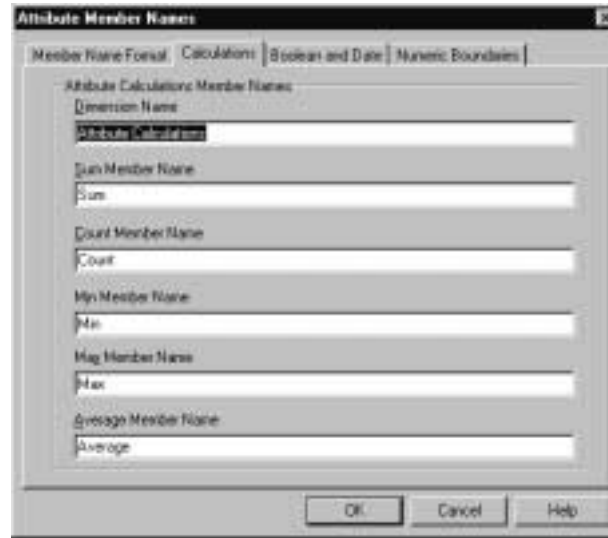


Figure 3-14: Attribute Member Names Dialog Box, Calculations Tab

3. To rename the dimension, type the new name in the **Dimension Name** text box.
4. Type the new name for the calculation into the relevant text box.
Sum can be renamed Total, **Min** can be renamed Minimum, and so forth.
5. Click **OK**.

For further information about using attribute calculations, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Specifying Attribute Boolean Names and Date Formats

After you have defined a Boolean or date type attribute, you can rename the Boolean value for Boolean attribute members and define formats for attribute date values. Boolean names cannot be the same as the names of other members of the metaoutline. For date type attribute members, a default mm-dd-yyyy format is assigned unless you specify a different value.

- To rename attribute Boolean names:
 1. Right-click the metaoutline icon, and select Attribute Member Names from the pop-up menu.
The **Attribute Member Names** dialog box is displayed.
 2. Select the **Boolean and Date** tab, as shown in Figure 3-15.



Figure 3-15: Attribute Member Names Dialog Box, Boolean and Date Tab

3. Make one of the following entries, and then click **OK**:

- If the member name True is used elsewhere in the metaoutline, type **Yes**, **Allowed**, or another appropriate name in the **True Member Name** text box.

The default member name is **True**.

- If the member name False is used elsewhere in the metaoutline, type **No**, **Not Allowed**, or another appropriate name in the **False Member Name** text box.

The default member name is **False**.

Note: If you have more than one Boolean attribute dimension, you must specify a prefix or suffix member name format to ensure unique member names; for example, Caffeinated_True and Caffeinated_False. See “Setting the Member Names Format” on page 3-27.

Note: Changing the Boolean member name setting does not automatically change the names of members of existing Boolean attribute dimensions. You must change the names of existing Boolean members individually.

For information on setting Boolean properties for an attribute dimension, see “Setting Boolean Parameters for an Attribute” on page 3-39.

For further information about using attribute calculations, see the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.

- To choose a date format for attribute members:
 - In the **Boolean and Date** tab of the **Attribute Member Names** dialog box, select one of the following options, and then click **OK**:
 - **mm-dd-yyyy** displays the month before the day; October 19, 2000 is displayed as 10-19-2000.
 - **dd-mm-yyyy** displays the day before the month; October 19, 2000 is displayed as 19-10-2000.
- Note:** If you change the date member name format, the names of existing members may be invalid. For example, if the 10-19-2000 member exists, and you change the format to dd-mm-yyyy, outline verification will find this member invalid. If you change the date format, you must rebuild the date attribute dimensions.

Specifying Numeric Boundaries for Attributes

For dimension member attributes that you have defined as numeric, you can set the numeric attribute name as the top (highest number) or bottom (lowest number) in a numeric range. After you have set a boundary for a numeric attribute, you can then set numeric ranges of values for the attribute (see “Setting Numeric Ranges for Attribute Members” on page 3-42).

- To specify the numeric boundary names for attributes:
 1. Right-click on the metaoutline icon, and select Attribute Member Names from the pop-up menu.

The **Attribute Member Names** dialog box is displayed.

2. Select the **Numeric Boundaries** tab, as shown in Figure 3-16

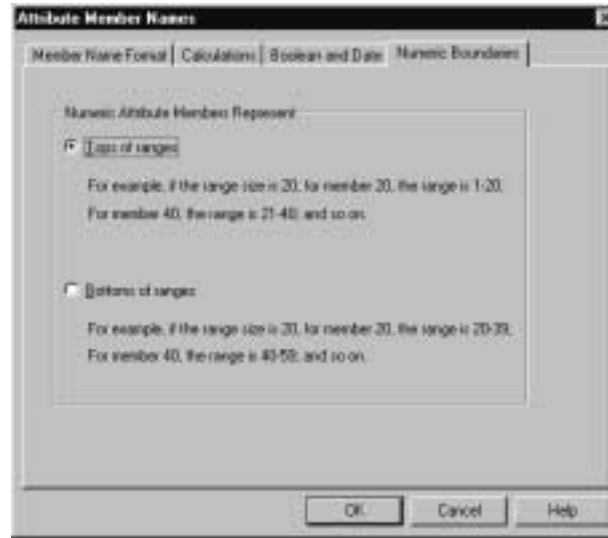


Figure 3-16: Attribute Member Names Dialog Box, Numeric Boundaries Tab

3. Select one of the following options:
 - **Tops of ranges:** Selecting this option sets the names of numeric attributes to the value at the top of the numeric range. If the range is 3,000,001 to 6,000,000, selecting this option sets the name as 6,000,000. **Tops of ranges** is the default setting.
 - **Bottoms of ranges:** Selecting this option sets the name of numeric attributes to the value at the bottom of the numeric range. If the range is 3,000,000 to 5,999,999, selecting this option sets the name as 3,000,000.
4. Click **OK**.

Renaming and Selecting a Data Type for an Attribute Member

After you create a dimension member attribute, you can rename the attribute, assign a different allowable dimension type (such as string or Boolean), and define an alias to make the attribute name easier to read in spreadsheet reports.

- To rename an attribute or to assign an alias and description to an attribute:
 1. Right-click the attribute dimension level, for example, CAFFEINATED, and select Properties from the pop-up menu.

The **Attribute Dimension Property** dialog box is displayed, as shown in Figure 3-17.

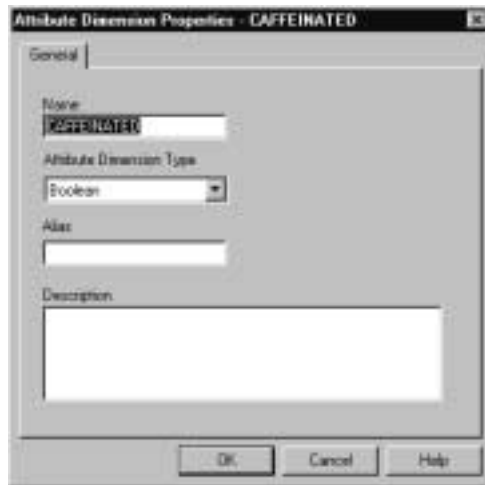


Figure 3-17: Attribute Dimension Properties Dialog Box

2. If you want to rename the dimension, type the new name into the **Name** text box.

3. Select a data type from the drop-down list in the **Attribute Dimension Type** list box in Figure 3-18.

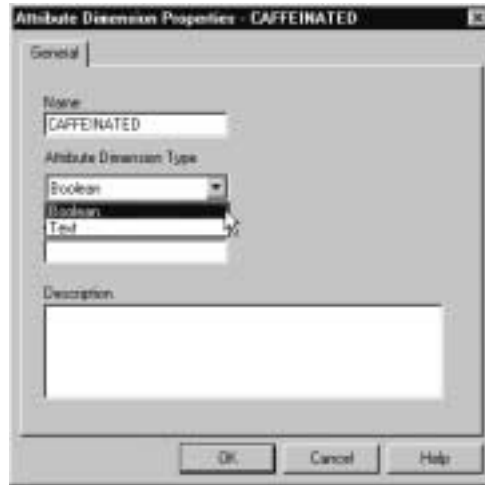


Figure 3-18: Selecting a Data Type in the Attribute Dimension Properties Dialog Box

For further information on choosing a data type for an attribute dimension, see “Choosing a Data Type for an Attribute Dimension” on page 3-36.

4. To give the dimension an alias, type an alias name in the **Alias** text box.
For example, for the PKGTYPE dimension, you might choose **Package Type** as the alias.
5. To record a description of the dimension, type it in the **Description** text box.
6. Click **OK**.

Choosing a Data Type for an Attribute Dimension

OLAP model attribute data types are based on the data type of the relational data source columns. Available attribute data types in the associated metaoutline are based on the Level 0 attribute member type.

For example, the CAFFEINATED column has a data type of string in the relational data source (identified by the ABC icon on the left side of the metaoutline window). Therefore, the available data types displayed in the Attribute Dimension Type drop-down list are Text and Boolean, with Text displayed as the default data

type selection. In Figure 3-18, the two possible data types for CAFFEINATED are shown in the drop-down list of the Attribute Dimension Properties dialog box. In Figure 3-18, the data type for CAFFEINATED is chosen to be Boolean, rather than the default data type, Text.

For each column data type in the OLAP model, only certain data types are available in the metaoutline. In the Attribute Dimension Properties dialog box, the available data types are shown in the Attribute Dimension Type list box (Figure 3-18). Data types can be chosen in the metaoutline based on the guidelines shown in Table 3-3.

Table 3-3: Allowable Data Type Changes for Attribute-enabled Columns

Column Data Type/Essbase Attribute Type	Text	Numeric	Boolean	Date
String	X		X	
Numeric	X	X	X	
Datetime	X			X

If you assign an attribute dimension to a numeric or Boolean data type, you can set further criteria for viewing the attribute. For information on setting ranges for numeric data types, see “Setting Numeric Ranges for Attribute Members” on page 3-42. For information on setting attribute Boolean properties, see “Setting Boolean Parameters for an Attribute” on page 3-39.

Setting Base Dimension Levels for an Attribute

Attribute-enabled columns in the OLAP model must be associated with base dimensions in the metaoutline. This association is automatically created when you drag an attribute-enabled column to the right frame in the OLAP metaoutline main window to form an attribute dimension (See “Creating Attribute Dimensions and Member Levels” on page 3-24).

You can also associate attribute members with base dimensions manually in either of two ways:

- By member level
- By metaoutline member

- To associate Level 0 attribute members with base dimension members:
 1. Right-click the attribute member name (for example, POPULATION.POPULATION) and select Properties from the pop-up menu to display the **General** tab of the **Attribute Member Properties** dialog box shown in Figure 3-19:



Figure 3-19: Attribute Member Properties Dialog Box, General Tab

2. Under **Alias**, optionally select the database column from the drop-down list that you want to associate as the alias.
3. Under **Association**, select one of the following options to associate an attribute member with a base dimension member:
 - Select the **By Level** option to associate the attribute with a base dimension level, and then select a level from the drop-down list.
 - Select the **By Metaoutline Member** option to associate the attribute with a base dimension name, and then select a name from the drop-down list.

- If an attribute cannot be associated with a base dimension, accept the default setting, **None**. Attribute members higher than the leaf member cannot be associated with a base dimension.

Note: Only the lowest level (leaf member) attribute can be associated with a base dimension. If an attribute member level is added below a leaf member of an attribute dimension, any association with the previous leaf member is removed. In that case, you must set up the new association with the new leaf member in this dialog box.

4. Enter optional information describing the attribute level association in the **Description** text box.
5. Click **OK**.

Setting Boolean Parameters for an Attribute

You can set flexible parameters for dimension member attributes to ensure that the data you need for critical analysis is available in spreadsheet reports. For example, you might want to see a report on how bottled soft drinks were selling in a certain state. However, you may have incomplete data, with some sales listed without information on whether the soft drink was sold in a bottle or a can. In a case such as this, you can make an assumption that soft drinks sold without information on their package type were sold in bottles and define an attribute that contains this assumption. The results of this assumption are easily visible on spreadsheet reports.

For information on setting the data type for an attribute, see “Renaming and Selecting a Data Type for an Attribute Member” on page 3-35 and “Choosing a Data Type for an Attribute Dimension” on page 3-36.

- To set Boolean values for an attribute member:
 1. Right-click the attribute member name (for example, **PRODUCT.PKGTYPE**) and select **Properties** from the pop-up menu to display the **Attribute Member Properties** dialog box.

2. Select the **Boolean Value** tab, as shown in Figure 3-20.



Figure 3-20: Attribute Member Properties Dialog Box, Boolean Value Tab

Note: In this example, the attribute dimension PKGTYPE, which has a text data type in the sample application, has been changed to a Boolean data type. For information on selecting a new data type, see “Renaming and Selecting a Data Type for an Attribute Member” on page 3-35.

- To set criteria for Boolean values:
 1. Select either the **True** or **False** option for **Apply Values As**.
Selecting **True** means that further selections in the **Boolean Value** tab will show as **True** on spreadsheets and reports. **True** is the default.
 2. In the **Operator** drop down list, select the appropriate operator; for example, you can select = (equal to).
 3. In the **Value** drop-down list, select the appropriate choice; for example, Bottle.
 4. Click the **Case** box to set case-sensitive criteria.

5. To add another line for another criterion, click **Add**.
6. Select **And** or **Or** in the **And/Or** drop-down list to determine how the new line will be used with existing lines.
7. Select an operator in the **Operator** drop-down list and a value in the **Value** drop-down list.
8. Click the **Case** box to set case-sensitive criteria.
9. Click the **Delete** box to delete any line you do not want.
10. Click **OK**.

Figure 3-21, for example, shows the Boolean Value tab selections after two criteria have been chosen. In Figure 3-21, if the package type is Bottle or Null (not reported), the results will show in reports as True for the PKGTYPE attribute member.

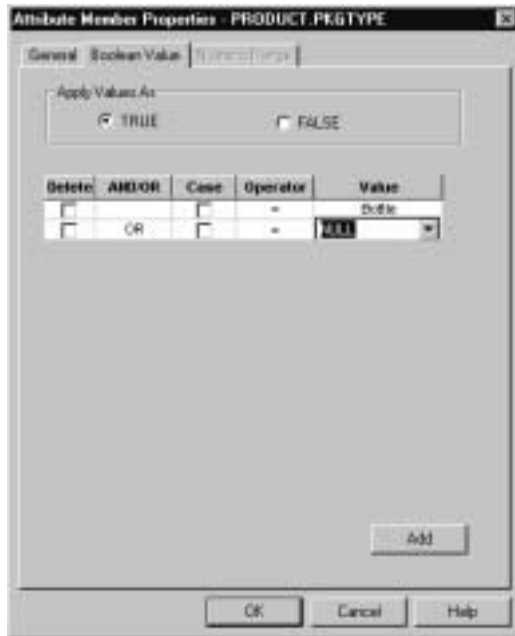


Figure 3-21: Two Criteria in the Boolean Value Tab of the Attribute Member Properties Dialog Box.

Setting Numeric Ranges for Attribute Members

After you have created a numeric data type dimension member attribute and have set numeric boundaries for the attribute, you can specify a starting point and size for numeric ranges for the data that you want reported. Numeric ranges can be either integers or float variables.

- To set numeric range values for an attribute member:
 1. Right-click the attribute member name and select Properties from the pop-up menu to display the **Attribute Member Properties** dialog box.
 2. Select the **Numeric Range** tab, as shown in Figure 3-22.



Figure 3-22: Attribute Member Properties Dialog Box, Numeric Range Tab

Note: Before setting the ranges in the **Numeric Range** tab of the **Attribute Member Properties** dialog box, you must set the numeric boundaries (**Top of ranges** or **Bottom of ranges**). For further information, see “Specifying Numeric Boundaries for Attributes” on page 3-33.

3. In the **Start** text box, enter the number to be the top value of the bottom range; for example, if you enter 2000 in the **Start** box, that will be the top number of the range “numbers below 2000.”
4. In the **Size** box, choose <below> from the drop-down list.

Setting the first **Size** as <below> sets a range for “numbers up to 2000.” It is not necessary, however, to have a <below> range. You can set **Start** to a number and **Size** to a number. In that case, numbers below the **Start** number will be not be in a defined range but will be listed individually.

Note: The **Size** box acts as a drop-down list or as a text box, depending on your choices for the range.

5. Click **Add** for a new line to be displayed.
6. In the new line, enter a number in the **Start** text box and a number in the **Size** text box, and click **Add**.

The number in the **Start** text box is the lowest number in the next range, and the number in the **Size** text box is the size of the range. However, this action defines more than one range. The ranges that are defined in this action have the same size, which is the number in the **Size** text box. The lowest number in the next range will be the number in the **Size** text box added to the number in the **Start** text box. The lowest number in the range after that will be two times the number in the **Size** text box, plus the number in the **Start** text box. Ranges of this size will continue to be generated up to the next number defined in the **Start** text box, or until the highest value is included in a range.

7. To define another set of ranges in the new line, enter a number in the **Start** text box for the bottom of the range, and enter a number in the **Size** text box for the size of the ranges.
8. To define a range for numbers above a certain value, enter the number in the **Start** text box, and choose <above> from the drop-down list in the **Size** text box.
9. Click the **Delete** box to delete any line.
10. Click **OK**.

Figure 3-22 shows an example of setting numeric ranges. In this figure, there is a numeric range for numbers below 3,000,000, followed by a set of numeric ranges of size 3,000,000, and ending with a numeric range for numbers above 20,000,000.

Seeing Attribute Results in Spreadsheets

Attributes are not included in Drill-Through reports, but they do appear in spreadsheets. An example from the sample Hyperion Essbase Spreadsheet Add-In (Figure 3-23) shows results in which the Boolean data type for CAFFEINATED equals TRUE. As shown in this example, attribute dimensions have high visibility in spreadsheet reports.

	A	B	C	D	E
1					Measures
2	Actual	Year True		100-10	22777
3	Actual	Year True		100-20	5708
4	Actual	Year True		100	28485
5	Actual	Year True		200-10	7201
6	Actual	Year True		200-20	12025
7	Actual	Year True		300	19226
8	Actual	Year True		300-10	12195
9	Actual	Year True		300-20	2511
10	Actual	Year True		300-30	11093
11	Actual	Year True		300	25799
12	Actual	Year True		400	0
13	Actual	Year True		100-20	5708
14	Actual	Year True		200-10	7201
15	Actual	Year True		200-20	12025
16	Actual	Year True		300-10	12195
17	Actual	Year True		300-20	2511
18	Actual	Year True		300-30	11093
19	Actual	Year True		Diet	28826
20	Actual	Year True		100-10	22777
21	Actual	Year True		100-20	5708
22	Actual	Year True		200-10	7201
23	Actual	Year True		200-20	12025


Figure 3-23: Spreadsheet Results for Caffeinated True

Renaming Dimensions, Member Levels, and Measures

You can change the names of metaoutline dimensions, member levels, or measures in the right frame of the OLAP Metaoutline main window or in the Properties dialog box. This topic contains the following subtopics that describe how to change names:

- “Renaming in the Right Frame” on page 3-45
- “Renaming in the Dimension Properties or Member Properties Dialog Box” on page 3-46
- “Renaming a Single Measure” on page 3-47

Renaming in the Right Frame

- To rename a dimension, member level, or measure in the right frame of the OLAP Metaoutline main window:
1. Select the dimension, member level, or measure name; for example, Quarter.
 2. Double click the dimension, member level, or measure; for example, .
 3. Enter the new name.
See “About Naming” on page 3-8.
 4. Perform one of the following tasks:
 - To save the new name, press **Enter**.
 - To rename another dimension, member level, or measure in the metaoutline, repeat steps 1 through 4.

Renaming in the Dimension Properties or Member Properties Dialog Box

- To rename a dimension, member level, or measure by using the **Dimension Properties** or **Member Properties** dialog box:
 1. In the right frame of the OLAP Metaoutline main window, select the dimension, member level, or measure name.
 2. Right-click the dimension, member level, or measure, and, from the pop-up menu, select Properties.

The **Dimension Properties** or the **Member Properties** dialog box is displayed (Figure 3-24).

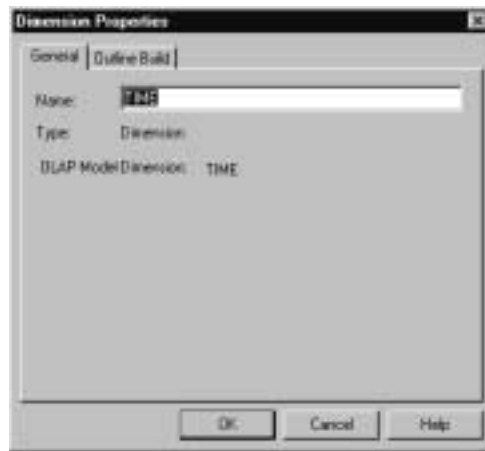


Figure 3-24: Renaming a Dimension or Member Level

3. In the **Name** text box, enter the new name.
See “About Naming” on page 3-8.
4. Click **OK**.

Renaming a Single Measure

► To rename a single measure by using the **OLAP Metaoutline Properties** dialog box:

1. In the right frame of the OLAP Metaoutline main window, select the metaoutline name; for example, TBC Metaoutline.
2. Right-click the metaoutline name, and, from the pop-up menu, select Properties.

The **OLAP Metaoutline Properties** dialog box is displayed.

3. Select the **Database Measures** tab (Figure 3-25).



Figure 3-25: Renaming a Single Measure

Adding Dimensions, Member Levels, and Measures to a Metaoutline

4. Select a measure and click **Edit**.

The **Measure Properties** dialog box is displayed (Figure 3-26).

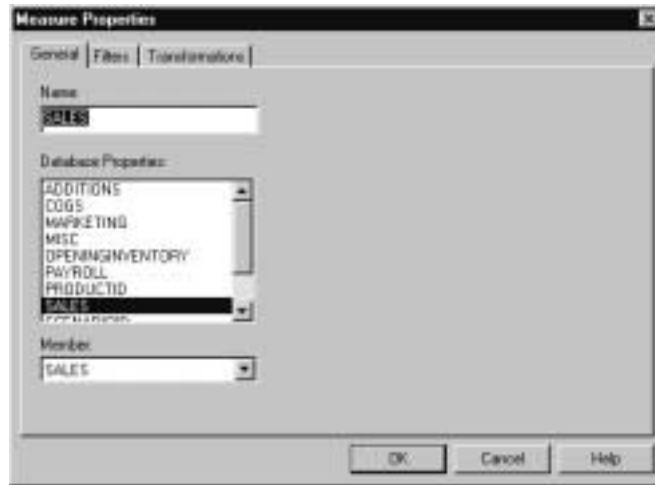


Figure 3-26: Using the Measure Properties Dialog Box

5. In the **Name** text box, enter the new name.
See “About Naming” on page 3-8.
6. Click **OK**.

Deleting Dimensions, Member Levels, and Measures

When you delete a dimension, member level, or measure from a metaoutline, you delete all build and load operations associated with the dimension, member level, or measure. You also delete all member levels located below the dimension, member level, or measure. For example, if you delete the Year dimension from TBC Metaoutline, you delete all members of the Year dimension; that is, you delete Time.Quarter and Time.Month.

3

CAUTION: You cannot undo a delete operation.

- To delete a dimension, member level, or measure from a metaoutline, choose any one of the following options:
 1. Select the dimension, member level, or measure and press the **Delete** key.
 2. Right-click the dimension, member level, or measure and select Delete from the pop-up menu.
 3. Select the dimension, member level, or measure and, from the Edit menu, select Delete.

Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines

After you create a metaoutline, you may want to see what the dimensions and members will look like in the Hyperion Essbase outline before you build the entire Hyperion Essbase outline. Before you build the Hyperion Essbase outline, you can view the metaoutline in Hyperion Essbase outline format to ensure that you have defined the dimensions and the member levels correctly, set filters and transformations correctly, and optimized the SQL generated by OLAP Integration Server.

CAUTION: OLAP Integration Server truncates sample outlines if they contain more than 32,767 members.

- To view a dimension, member, or metaoutline in Hyperion Essbase outline format:
1. Be sure there is enough RAM on the computer on which Hyperion Integration Server Desktop is installed to view the entire sample.

If you are not sure of the available RAM, set a query governor to control the duration and the size of the query that OLAP Integration Server makes to the relational data source.
 2. Within the right frame of the OLAP Metaoutline main window, use one of the following methods to preview a sample outline:
 - For a sample *outline*, select the metaoutline name—for example, TBC Metaoutline—and select View > Sample.
 - For a sample *dimension* or *member*, right-click the dimension or member level—for example, Product—and, from the pop-up menu, select Sample.
 - For results of filtering or transformation operations on either the **Edit OLAP Metaoutline Transformations** dialog box or the **Member Load Filtering** dialog box, select **Preview Results**.

Hyperion Integration Server Desktop opens the appropriate dialog box: either the **Sample Outline** dialog box (Figure 3-27), **Preview Filter Results** dialog box, or the **Preview Transformation Results** dialog box.



Figure 3-27: Viewing a Sample Hyperion Essbase Outline

The sample outline contains the sample dimensions. You can click dimension and member names to expand dimensions and members and examine the structure of the Hyperion Essbase outline; for example, you can examine the Product dimension.

The **Sample Build SQL** box displays the SQL statements that OLAP Integration Server generates to create the sample Hyperion Essbase outline. Use the sample SQL to perform any of the following functions:

- Make sure that the metaoutline refers to the intended data in the external data source.
 - Tune the external data source to speed the process of building and loading data into the Hyperion Essbase outline.
 - Cut and paste the SQL statement to an external file to view or print.
3. When you finish viewing the sample dimensions and the sample SQL, click **Close**.

Viewing Defined Measures

After you complete a metaoutline, you can review the measures it contains.

- To view all measures defined in a metaoutline:
 1. In the right frame of the OLAP Metaoutline main window, select the metaoutline name; for example, TBC Metaoutline.
 2. Right-click the metaoutline name and, from the pop-up menu, select Properties.

The **OLAP Metaoutline Properties** dialog box is displayed.

3. Select the **Database Measures** tab, as shown in Figure 3-28.



Figure 3-28: List of Measures in the Metaoutline

A list of the measures defined in the metaoutline is displayed in the list box.

4. When you finish viewing the measures, click **OK**.

You can also use the **Database Measures** tab of the **OLAP Metaoutline Properties** dialog box to edit or delete a measure. In the **Database Measures** tab, select the measure—for example, COGS—and click **Edit** or **Delete**.

If you edit a measure, you can change any of the following features:

- The name of the measure. See “Renaming a Single Measure” on page 3-47.
- The column on which the measure is based. Use the same steps as for creating a single measure. See “Changing the Column Associated with a Measure” on page 4-30.
- Transformations to perform on the measure. See “Scaling Measures” on page 4-28 and “Reversing Measure Signs” on page 4-29.
- Filters defined on the measure. See “Creating a Measure Filter Interactively” on page 5-23.

Adding Dimensions, Member Levels, and Measures to a Metaoutline

Setting Load Properties

This chapter describes how to set load properties for dimensions, member levels, and measures. *Load properties* determine what actions OLAP Integration Server performs on dimensions, member levels, and measures as it builds a Hyperion Essbase outline or loads data into a Hyperion Essbase outline. For example, OLAP Integration Server can transform member level names as it loads them from the relational data source to the related Hyperion Essbase outline.

Note: Before setting load properties, you should perform the following transformations in the OLAP model: splitting one column into two or more columns; creating new columns that are based on existing columns; and fixing data inconsistencies, such as making sure that California is used as either a region or a state, but not as both. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 4-2
- “Setting Load Properties for Dimensions” on page 4-3
- “Setting Load Properties for Member Names” on page 4-11
- “Setting Load Properties for Members” on page 4-24
- “Setting Load Properties for Measures” on page 4-27

For information about setting other properties, see Chapter 5, “Working with Filters” and Chapter 6, “Setting Hyperion Essbase Properties.”

Understanding the Metaoutline Workflow

Figure 4-1 illustrates the workflow for creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following processes:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and members from a metaoutline into a Hyperion Essbase outline
- Loading the data specified in a metaoutline into a Hyperion Essbase database

This chapter focuses on setting load properties. You can set load properties before or after setting filters and Hyperion Essbase properties.

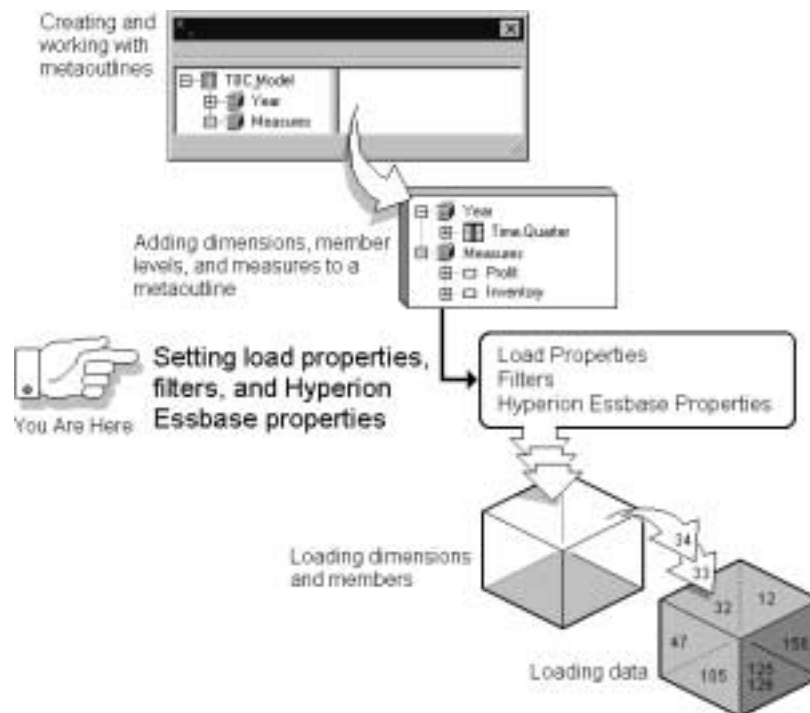


Figure 4-1: Metaoutline Workflow

Setting Load Properties for Dimensions

The following topics describe the load properties that you can set for dimensions, such as determining what to do with duplicate member names and with new member levels.

- “About Duplicate Member Names” on page 4-3
- “Understanding Rules for Shared Members” on page 4-6
- “Handling Duplicate Member Names” on page 4-6
- “Handling New Members During Data Load” on page 4-8

About Duplicate Member Names

A data source and a metaoutline can contain duplicate member names. For example, TBC Metaoutline uses the 100_20 column twice, once under 100 and once under Diet.

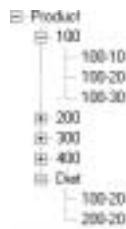


Figure 4-2: 100_20 Column

When OLAP Integration Server performs a member load, it performs one of the following actions when it encounters the members under Diet:

- It ignores the duplicate members. OLAP Integration Server does not load the members into the Hyperion Essbase outline.

Setting Load Properties

- It creates the members under Diet as *shared members*, as shown in Figure 4-3. The data associated with a shared member is stored in the real member. For example, the data for the second 300_30 member (under Diet) is stored in the first 300_30 member (under Cream Soda).



Figure 4-3: Shared Members

- In a recursive hierarchy, it moves duplicate members within the recursive hierarchy.
- In a recursive hierarchy, it moves generation 2 members that have no existing parent within the recursive hierarchy.

For example, suppose that you have the following manager-employee organizational chart under the People dimension:

Manager	Employee
Dotty	Donna
Nancy	Dotty
Dotty	Linda
Nancy	Jack
ABC Project	Dotty
ABC Project	Donna
XYZ Project	Jack

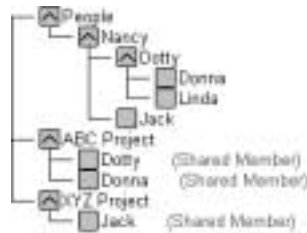
The hierarchy created by loading the first record is as follows:



Because the first hierarchy contains a generation 2 member without a parent, the first hierarchy is moved down a level when the second record is loaded, as shown in the following example:



The following Hyperion Essbase outline is generated from the previously described recursive hierarchy:



For more information on recursive hierarchies, see “Creating Recursive Dimensions” on page 3-17.

You can prevent the creation of duplicate member names by transforming data when you create an OLAP model. For more information, see the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

Understanding Rules for Shared Members

Hyperion Essbase follows these rules for shared members:

- Shared members must be in the same dimension. For example, both 100-20 members are in the Product dimension.
- A dimension can contain both shared and non-shared members.
- Shared members cannot have children.
- An unlimited number of shared members can have the same name.
- User-defined attributes (UDAs), formulas, consolidation properties, and account properties cannot be assigned to shared members.
- Shared members can be assigned aliases.
- In a Hyperion Essbase outline, a shared member should not be located before its real member. For example, the 100-20 shared member (under Diet) must be located *after* the 100-20 real member (under Colas) in the Hyperion Essbase outline.
- Any transformation performed on a member name must also be performed on the duplicate member name. For example, if you change the original member level name from New York to NY, you must perform the same change on the duplicate member name.

Handling Duplicate Member Names

➤ To handle duplicate names in a metaoutline:

1. In the metaoutline, select a dimension that you expect to contain duplicate names; for example, Product.
2. Right-click the dimension and, from the pop-up menu, select Properties to open the **Dimension Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Outline Build** tab.



Figure 4-4: Handling Duplicate Member Names

4. From the **Duplicate Member Names** group, select one of the following options:
 - **Ignore** to ignore duplicate member names during the build of the Hyperion Essbase outline.
OLAP Integration Server does not load the members into the Hyperion Essbase outline.
 - **Add as shared members** to create the duplicate member names as shared members in the Hyperion Essbase outline.
 - **Move members** to move duplicate members of recursive hierarchies.

- Select **Share members/move if no parent(Gen2)** to move, within recursive hierarchies, generation 2 members that have no existing parent, even if share is specified.

For more information on this option, see “About Duplicate Member Names” on page 4-3.

Any transformation performed on a member name must also be performed on the duplicate member name. For example, if you change the original member level name from New York to NY, you must perform the same change on the duplicate member name.

If you are not sure which option to select, see “About Duplicate Member Names” on page 4-3.

5. Click OK.

Tip: You can also create shared members by setting the Shared Member Hyperion Essbase property. See “Specifying Data Storage Properties for Dimensions and Members” on page 6-9.

Handling New Members During Data Load

During the time lapse between the build and the data load of a Hyperion Essbase outline, the information on which the outline structure is based can change. For example, someone may add a new member to the fact table or to a reference table.

Assume that the Hyperion Essbase outline build produces an outline that looks like the one in Figure 4-5:



Figure 4-5: Hyperion Essbase Outline

What if someone adds Root_Beer to the fact table and to a reference table of the data source? When you perform the data load, the new member is available.

If the Hyperion Essbase outline includes miscellaneous members, OLAP Integration Server adds the new member to the miscellaneous member of the appropriate dimension. With the addition of the Root_Beer member, for example, the Hyperion Essbase outline looks like the one in Figure 4-6:



Figure 4-6: Miscellaneous Member Example

After the data load is complete, you can view the members of Miscellaneous.

During data load, OLAP Integration Server ignores any new member whose Hyperion Essbase dimension does not include a miscellaneous member. However, OLAP Integration Server records that the new member exists and that it was ignored during the load. For more information on viewing the log file in which this information is recorded, see the *Hyperion Integration Server System Administrator's Guide*.

- To add new members to a miscellaneous member level in a metaoutline:
 1. In the metaoutline, select the dimension that you expect to contain new member levels; for example, Product.
 2. Right-click the dimension and, from the pop-up menu, select Properties to open the **Dimension Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Outline Build** tab.



Figure 4-7: Setting the Miscellaneous Member

4. From the **Default Miscellaneous Member** group, select **Enabled**.
5. In the **Name** box, enter the name of the miscellaneous member; for example, **Misc**.

See “About Naming” on page 3-8.

You may ignore the dimension name. However, it is recommended that you include the dimension name as a prefix or suffix of the miscellaneous member name. Prefixing or suffixing a member name with the name of the related dimension helps you to identify the dimension with which the member is associated.

For example, if the new member, **Root_Beer**, is in the **Product** dimension, and you prefix the miscellaneous member name with the dimension name, the Miscellaneous member looks as it does in Figure 4-8:



Figure 4-8: Miscellaneous Dimension

6. If you want to set a miscellaneous member on more than one dimension, return to step 1.

Note: If you set a miscellaneous member on more than one dimension, you must prefix or suffix the dimension name to the name of each additional miscellaneous member so that each miscellaneous member name is unique. If you do not prefix or suffix the names of subsequent miscellaneous members, you will have multiple members named Miscellaneous; for example, one under Product and one under Market.

7. Click **OK**.

Setting Load Properties for Member Names

4

This topic describes how to set load properties for member names, including how to replace parts of member names, drop spaces around member names, convert spaces to underscores within member names, attach text strings to member names, and change the case of member names.

When setting load properties, follow these general guidelines:

- If you define load properties on a real member that you expect to be shared, you must define the same load properties on each shared member.
- You cannot perform transformations on aliases.
- Do not set transformations on members of recursive dimensions. See “Creating Recursive Dimensions” on page 3-17.

This topic contains the following subtopics:

- “Transforming Member Names” on page 4-12
- “Replacing Parts of Member Names” on page 4-13
- “Dropping Spaces From Around Member Names” on page 4-16
- “Converting Spaces Within Member Names to Underscores” on page 4-17
- “Attaching Prefixes and Suffixes to Member Names” on page 4-18
- “Changing the Case of Member Names” on page 4-22

Transforming Member Names

You can transform a member name so that the member name loaded into the Hyperion Essbase outline differs from the member name in the relational data source. For example, suppose that the data source stores product stock unit numbers.

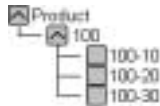


Figure 4-9: Product Dimension

You can prefix each number with SKU- so that the product codes are recognizable as stock units.

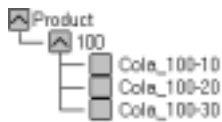


Figure 4-10: Product Members with Prefixes

Note: If you sort members, they are sorted before OLAP Integration Server applies transformations to member names. See “Sorting Members” on page 4-24.

OLAP Integration Server performs the specified transformations on member names in the following order:

1. It drops spaces from around a member name.
See “Dropping Spaces From Around Member Names” on page 4-16.
2. It converts spaces to underscores.
See “Converting Spaces Within Member Names to Underscores” on page 4-17.
3. It changes case.
See “Changing the Case of Member Names” on page 4-22.

4. It performs all replace operations in the order that you created the operations. If you are not sure of the order of the replace operations, right-click the member level and, from the pop-up menu, select Transformation to open the Edit Transformations dialog box.
See “Replacing Parts of Member Names” on page 4-13.
5. It applies prefix operations.
See “Attaching Prefixes and Suffixes to Member Names” on page 4-18.
6. It applies prefix separator operations.
7. It applies suffix operations.
8. It applies suffix separator operations.

Replacing Parts of Member Names

As a member is loaded into a Hyperion Essbase outline, its name can be changed. For example, assume that the data source abbreviates New York as NY. You can direct OLAP Integration Server to replace each NY with New York during the build of the Hyperion Essbase outline.

- To change a member name during loading:
 1. In the metaoutline, select the member level to change; for example, REGION.REGION.

Setting Load Properties

2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit Transformations** dialog box.

You can also open the dialog box by selecting Edit > Transformation.

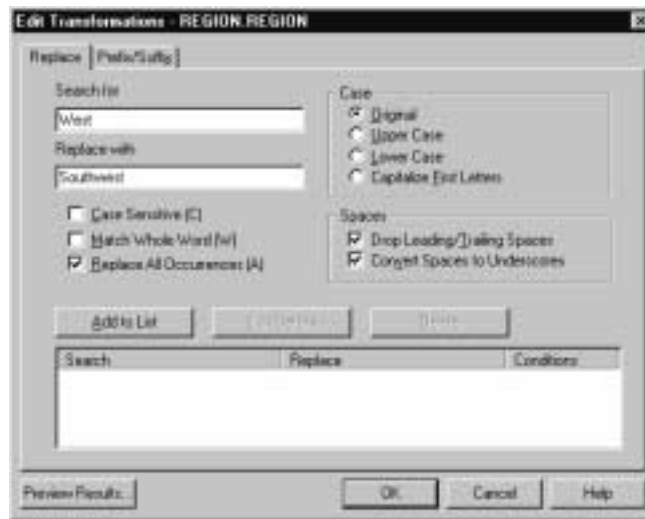


Figure 4-11: Replacing Text Strings

3. In the **Search for** text box, enter the text string that you want to replace; for example, **West**.
4. In the **Replace with** text box, enter the text with which to replace the text string; for example, **Southwest**.

Note: Do not use double quotation marks (") in the **Replace with** text box.

If you want to replace a text string with an empty string, leave the **Replace with** text box empty.

Note: When replacing a member name, make sure that you do not use any of the following characters at the beginning of the name:

@	(at sign)	()	(parentheses)
\	(backslash)	.	(period)
{ }	(braces)	+	(plus sign)
,	(comma)	'	(single quotation mark)
-	(dash, hyphen, or minus sign)	_	(underscore)
=	(equal sign)		(vertical bar)
<	(less than sign)	"	(double quotation mark)

5. Specify which of the following options the replacement operation should observe:
 - Be case-sensitive; that is, replace only text strings that match the capitalization of the string in the **Search for** text box.
 - Replace the text string only when it occurs as a whole word. For example, to replace the 10 in the string 100 10 1 with an A, select the **Match Whole Word** option. If you do not select the **Match Whole Word** option, the string is changed to A0 10 1. By default, Hyperion Essbase changes only the first occurrence.
 - Replace all occurrences of the string. For example, if you replace every occurrence of 10 in the string 100 10 1 with A, the string changes to A0 A 1.
6. Click **Add to List** to add the replacement operation to the list.
7. If you want to change or delete an existing operation, complete the appropriate action:
 - To change an existing operation, select the operation and click **Edit Details**.
 - To delete an operation, select the operation and click **Delete**.

8. Click **Preview Results** to open the **Preview Transformation Results** dialog box to view your changes in a sample Hyperion Essbase outline.
See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.
9. Click **OK**.

You can use the search and replace operation of the **Edit Transformations** dialog box to sort months chronologically.

- To sort months chronologically:
 1. In the metaoutline, select the appropriate member level, right-click the member level, and, from the pop-up menu, select Transformation.
The **Edit Transformations** dialog box is displayed.
 2. Replace each month name with a number (January=1, February=2, and so on).
See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.
 3. Sort the month field in ascending order.
See “Sorting Members” on page 4-24.
 4. Select the **Match Whole Word** option.
 5. Replace the month number with the month name (1=January, 2=February, and so on).

Dropping Spaces From Around Member Names

Some relational data sources pad a field by adding spaces at the beginning or end of the field. You must direct OLAP Integration Server to remove these spaces during the build of the Hyperion Essbase outline.

- To remove spaces from around a member name:
 1. In the metaoutline, select the member level that is surrounded by spaces; for example, Month.
 2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit Transformations** dialog box, shown in Figure 4-11.

3. If the **Drop Leading/Trailing** option, **Drop Leading/Trailing Spaces**, is not selected, select it.
By default, the **Drop Leading/Trailing Spaces** box is selected.
4. Click **OK**.

Converting Spaces Within Member Names to Underscores

In some relational data sources, the names of information categories contain spaces. You may want to direct OLAP Integration Server to change these spaces to underscores during the build of the Hyperion Essbase outline.

- To convert spaces within member names to underscores:
1. In the metaoutline, select the member level that contains names that include spaces; for example, `PRODUCT_GROUP_DESC` contains names with spaces, such as Root Beer.
 2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit Transformations** dialog box, shown in Figure 4-11.
 3. If the **Convert Spaces to Underscores** box, **Convert Spaces to Underscores**, is not selected, select it.
By default, the **Convert Spaces to Underscores** box is selected.
 4. Click **OK**.

Attaching Prefixes and Suffixes to Member Names

You can direct OLAP Integration Server to attach a prefix or a suffix to a member name. For example, you can attach the prefix SKU_ to all member names in the Product dimension.

Note: When attaching a prefix to a member name, make sure that you do not use any of the following characters at the beginning of the name:

@	(at sign)	()	(parentheses)
\	(backslash)	.	(period)
{ }	(braces)	+	(plus sign)
,	(comma)	'	(single quotation mark)
-	(dash, hyphen, or minus sign)	_	(underscore)
=	(equal sign)		(vertical bar)
<	(less than sign)	"	(double quotation mark)

- To attach a prefix or suffix to a member name:
1. Select the member level to attach the prefix or suffix to; for example, STATE.
 2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit Transformations** dialog box.
You can also open the dialog box by selecting Edit > Transformation.

3. Select the **Prefix/Suffix** tab.



Figure 4-12: Adding Prefixes or Suffixes

4. Decide whether to use the **Prefix** option group or the **Suffix** option group.
 - **Prefix** attaches the selected value in front of the member name; for example, SKU_100-10.
 - **Suffix** attaches the selected value after the member name; for example, 100-10_SKU.
5. In the **Prefix** option group or the **Suffix** option group, as appropriate, select the source of the value that Hyperion Essbase attaches to the member name in the Hyperion Essbase outline.

Some of the following examples are based on transforming the member STATE of the Market dimension, as shown in the following sample TBC Metaoutline:



Setting Load Properties

- **None** uses the metaoutline name as the Hyperion Essbase outline name. Nothing is attached.
- **Parent Name** attaches the name of the immediate parent; for example, selecting **Parent Name** in the **Prefix** option group and **Pipe** in the **Prefix Separator** option group changes Colas to Product|Colas.

Hyperion Essbase requires every member name to be unique. To satisfy this requirement, you can prefix the name of every member with the name of the parent value of the member. For example, if you store information for two years, you have two members named Quarter1. By prefixing the name of every member with the name of the parent of the member, you produce unique member names such as 1997Quarter1 and 1998Quarter1.

- **Grandparent Name** attaches the name of the grandparent; that is, the name of the parent of the parent. Using PRODUCT_ID as an example, selecting **Grandparent Name** in the **Suffix** option group and **Caret** in the **Suffix Separator** option group changes Colas to 100-10^Colas.
- **All Ancestors' Names** attaches the names of all higher level generations. For example, selecting **All Ancestors' Names** in the **Prefix** option group and **Underscore** in the **Prefix Separator** option group changes 100-10 to Product_Colas_100_100-10.
- **Dimension Name** attaches the dimension name. For example, selecting **Dimension Name** in the **Suffix** option group and **Caret** in the **Suffix Separator** option group changes Colas to Colas^Product.
- **Custom Prefix** and **Custom Suffix** attach, respectively, the prefix or suffix that you enter in the appropriate text box. For example, typing SKU in the **Custom Prefix** text box and selecting **Underscore** in the **Prefix Separator** option group changes Colas to SKU_Colas.

Note: Do not use double quotation marks (") in the **Custom Prefix** and **Custom Suffix** text box.

- **Custom Prefix** and **Custom Suffix** combined with a **Prefix** option or a **Suffix** option attach, respectively, the prefix or suffix that you enter in the appropriate text box and the selected prefix or suffix. For example, typing SKU in the **Custom Prefix** text box, selecting **Parent** in the **Prefix Separator** option group, and selecting **Underscore** in the **Prefix Separator** option group changes Colas to Product_SKU_Colas.

Note: Do not use double quotation marks (") in the **Custom Prefix** and **Custom Suffix** text box.

6. In the **Prefix Separator** or **Suffix Separator** option group, select the separator character to insert between the member name and the attached value.
 - **Underscore [_]** inserts an underscore between the affix and the member name; for example, SKU_100-10.
 - **Pipe [|]** inserts a pipe between the affix and the member name; for example, Product|100-10.
 - **Caret [^]** inserts a caret between the affix and the member name; for example, Product^100-10.
 - **No separator** does not insert a character separator.
7. Click **Preview Results** to open the **Preview Transformation Results** dialog box to preview your changes in a sample Hyperion Essbase outline.
See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.
8. Click **OK**.

Changing the Case of Member Names

You can direct OLAP Integration Server to change the case of member names. For example, you can change the member name JANUARY to January.

► To change the case of a member name:

1. In the metaoutline, select the member level for which you want to change the case; for example, State.

OLAP Integration Server changes the case of *all* member names that the selected member level maps to in the Hyperion Essbase outline; for example, Illinois, Ohio, and so on.

2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit Transformations** dialog box.

You can also open the dialog box by selecting Edit > Transformation.



Figure 4-13: Capitalizing Member Names

3. From the **Case** group, select the case to which the member name will change:
 - **Original** does not change the case.
 - **Upper Case** makes all letters of the member name uppercase; for example, Illinois to ILLINOIS or illinois to ILLINOIS.
 - **Lower Case** makes all letters of the member name lowercase; for example, ILLINOIS to illinois or Illinois to illinois.
 - **Capitalize First Letters** makes the first letter of the member name and the first letter following a space or an underscore uppercase and makes all subsequent letters lowercase; for example, central_illinois or CENTRAL_ILLINOIS to Central_Illinois
4. Click **Preview Results** to open the **Preview Transformation Results** dialog box to view your changes in a sample Hyperion Essbase outline.

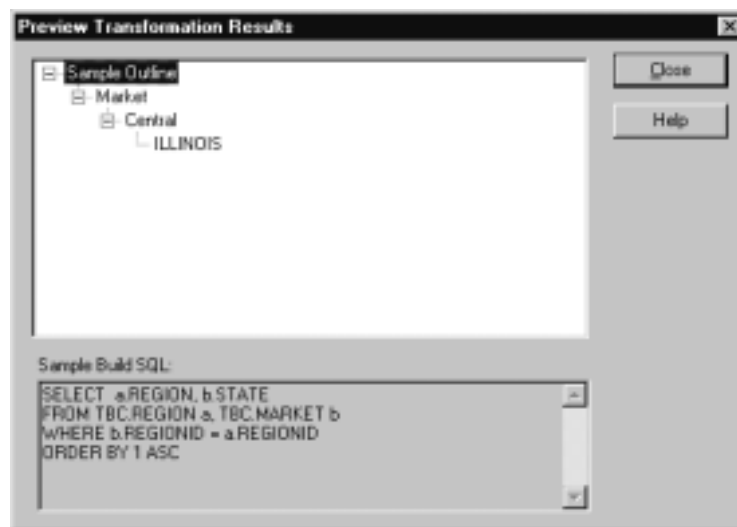


Figure 4-14: Previewing Transformation Results

Also see “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.

5. Click **Close** in the **Preview Transformation Results** dialog box to return to the **Edit Transformations** dialog box.
6. Click **OK**.

Setting Load Properties for Members

This topic describes how to set load properties that determine the order in which members are displayed in the Hyperion Essbase outline and the level to which OLAP Integration Server builds the outline. This topic contains the following subtopics:

- “Sorting Members” on page 4-24
- “Setting the Member and Data Load Build Level” on page 4-25

Sorting Members

You can set OLAP Integration Server to sort member names alphanumerically within a dimension. For example, you can sort all states in the Market dimension from A to Z so that a United States Market dimension starts with Alabama and ends with Wyoming.

Note: The relational database performs the sort. Some sorts may not work the way you intend. For example, if you sort alphabetically by month, you get April, August, December, and so on. To sort months chronologically, see “Replacing Parts of Member Names” on page 4-13.

Note: If you sort members, they are sorted before OLAP Integration Server applies any transformations to the member names.

- To sort the members in a dimension:
1. In the metaoutline, select the member level that you want to sort; for example, MARKET.STATE.
 2. Right-click the member level and, from the pop-up menu, select Filter to open the **Edit Filter** dialog box.

You can also open the dialog box by selecting Edit > Filter.

3. Select the **Ordering** tab.

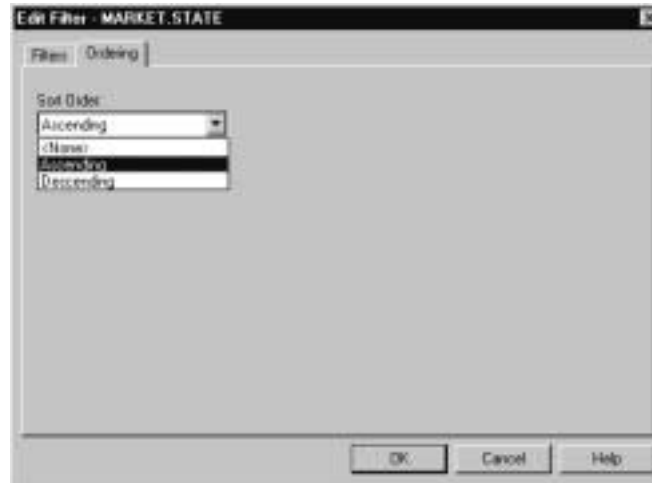


Figure 4-15: Setting Member Ordering

4. From the **Sort Order** drop-down list, select a sort order.

You can select **Ascending** (A to Z, 0 to 9), **Descending** (Z to A, 9 to 0), or **None**. **None** is the default value. Selecting **Ascending** or **Descending**, instead of **None**, improves member load performance.

5. Click **OK**.

Setting the Member and Data Load Build Level

By default, when OLAP Integration Server creates a Hyperion Essbase outline, it loads all member levels specified in the metaoutline. You can, however, set OLAP Integration Server to build only to a specified level. Building down to a specified level produces a smaller Hyperion Essbase outline that contains less data. Smaller outlines can be useful for users with limited disk space who do not need to see the lowest level of detail.

Setting Load Properties

For example, if the Market dimension in the metaoutline contains three levels—REGION, STATE, and CITY—the Market dimension of the default Hyperion Essbase outline contains three levels, as shown in Figure 4-16.



Figure 4-16: Members of the Market Dimension

If you set OLAP Integration Server to build only to the STATE member level, none of the CITY members are displayed in the Hyperion Essbase outline, as shown in Figure 4-17.

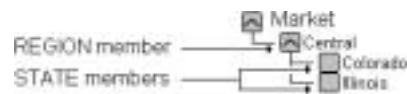


Figure 4-17: Reduced Number of Levels

Likewise, when you load data into the Hyperion Essbase outline, OLAP Integration Server loads the measures only down to the specified build level.

- To specify a build level:
 1. In the metaoutline, select the member level that you want to use as the lowest level member (leaf member); for example, PRODUCTDIM.FAMILY.
 2. Right-click the member level and, from the pop-up menu, select Build down to here.

In the pop-up menu, next to Build down to here, a check is displayed. In the metaoutline, for all member levels that lie below the specified level, a red X is displayed. During member and data loads, OLAP Integration Server sets the specified member level (the member level with a check) as the level 0 member (that is, as the leaf member). For example, Figure 4-18 displays the original level 0 member (PRODUCTDIM.SKU) with a red X, and PRODUCTDIM.FAMILY becomes the lowest member for the build.



Figure 4-18: Build Down to Here

Note: Setting the Build down to here function on a level 0 member has no effect. OLAP Integration Server builds down to level 0 by default.

Setting Load Properties for Measures

This topic describes how to set load properties for measures, including how to change a measure, how to reverse the sign of a measure, and how to change the column with which a measure is associated. This topic contains the following subtopics:

- “Scaling Measures” on page 4-28
- “Reversing Measure Signs” on page 4-29
- “Changing the Column Associated with a Measure” on page 4-30

Scaling Measures

You can direct OLAP Integration Server to scale measures (data values) as it loads them into the Hyperion Essbase database. For example, you may want to scale a measure if the values in the data source and the values in the database use different types of measurement, such as when the data source tracks sales in hundreds but the Hyperion Essbase database tracks sales in units. In this case, you want to multiply incoming values by 100.

► To scale an incoming measure:

1. In the metaoutline, select a measure; for example, SALES.
2. From the pop-up menu, select Transformation to open the **Measure Properties** dialog box.

You can also open the dialog box by selecting Edit > Transformation.



Figure 4-19: Scaling Measures

3. From the **Operator** drop-down list, select the operator to use for scaling the incoming measure; for example, * (Multiplication).
4. In the **Value** text box, enter the value to use; for example, type **10** to multiply the incoming value by 10.
5. Click **OK**.

OLAP Integration Server scales the measures and loads the scaled values into the Hyperion Essbase database.

Reversing Measure Signs

You can direct OLAP Integration Server to reverse the sign of a measure (from positive to negative or vice versa). For example, the data source can track expenses as negative numbers, but you can track them as positive numbers in the Hyperion Essbase database.

- To flip (reverse) the sign of an incoming measure:
 1. In the metaoutline, right-click the measure to flip; for example, Expenses.
 2. From the pop-up menu, select Transformation to open the **Transformation** tab of the **Measure Properties** dialog box.

You can also open the tab by selecting Edit > Transformation.

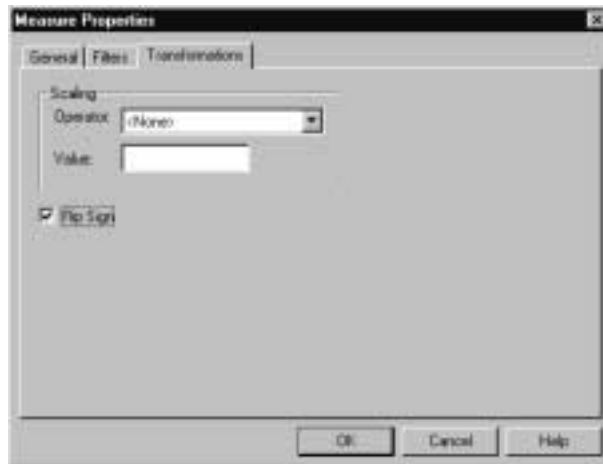


Figure 4-20: Reversing Signs for Measures

3. Select the **Flip Sign** box.
4. Click **OK**.

If you later decide to load values into the measure without reversing the sign, deselect the **Flip Sign** box.

Changing the Column Associated with a Measure

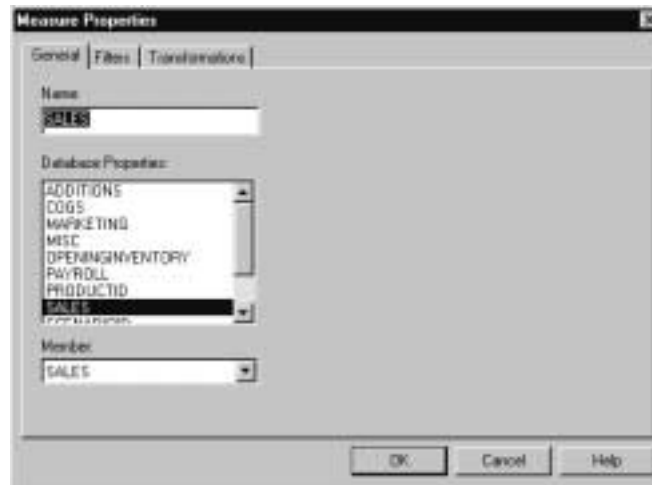
When data is loaded into a Hyperion Essbase database, every measure in the related metaoutline retrieves its data values from the relational data column with which it is associated. For example, assume that you have a SALESINVACT.PRICE column in the relational data source. The column contains all prices for all products. The related OLAP model contains the SALESINVACT.PRICE column in its fact table. When you create a PRICE measure in the metaoutline, the PRICE measure is based on the SALESINVACT.PRICE column of the relational data source.

When you load data into the Hyperion Essbase database, OLAP Integration Server retrieves the data values for price from the SALESINVACT.PRICE column of the relational data source and loads the retrieved values into the Price measure of the Hyperion Essbase outline.

If the default column that is associated with a measure is not correct, you can change the column.

- To change the column with which a measure is associated:
 1. In the metaoutline, select a measure; for example, SALES.
 2. Right-click the measure and, from the pop-up menu, select Filter or Transformation to open the **Measure Properties** dialog box.

3. Select the **General** tab.



4

Figure 4-21: Changing the Column Associated with a Measure

4. From the **Database Properties** list, select the column with which you want to associate the member.
5. Click **OK**.

Setting Load Properties

This chapter describes how you set a filter on a member level and on a measure. A *filter* determines what members or what data values OLAP Integration Server loads from the external data source into the Hyperion Essbase database.

Note: A change that you make to a filter for a member level or a measure does not affect the Hyperion Essbase database until you perform a member or data load.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 5-2
- “About Filters” on page 5-3
- “Filtering in an OLAP Model or in a Metaoutline” on page 5-8
- “Working with Metaoutline Filters” on page 5-8
- “Creating a Member Filter Interactively” on page 5-14
- “Typing a Member Filter” on page 5-21
- “Creating a Measure Filter Interactively” on page 5-23
- “Typing a Measure Filter” on page 5-26
- “Creating an Attribute Filter Interactively” on page 5-28

For information about setting other properties, see Chapter 4, “Setting Load Properties” and Chapter 6, “Setting Hyperion Essbase Properties.”

Understanding the Metaoutline Workflow

Figure 5-1 illustrates the workflow for creating a metaoutline and for using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following processes:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and members from the metaoutline into a Hyperion Essbase outline
- Loading the data specified in the metaoutline into a Hyperion Essbase database

This chapter focuses on setting filters. You can set filters before or after setting load properties and Hyperion Essbase properties.

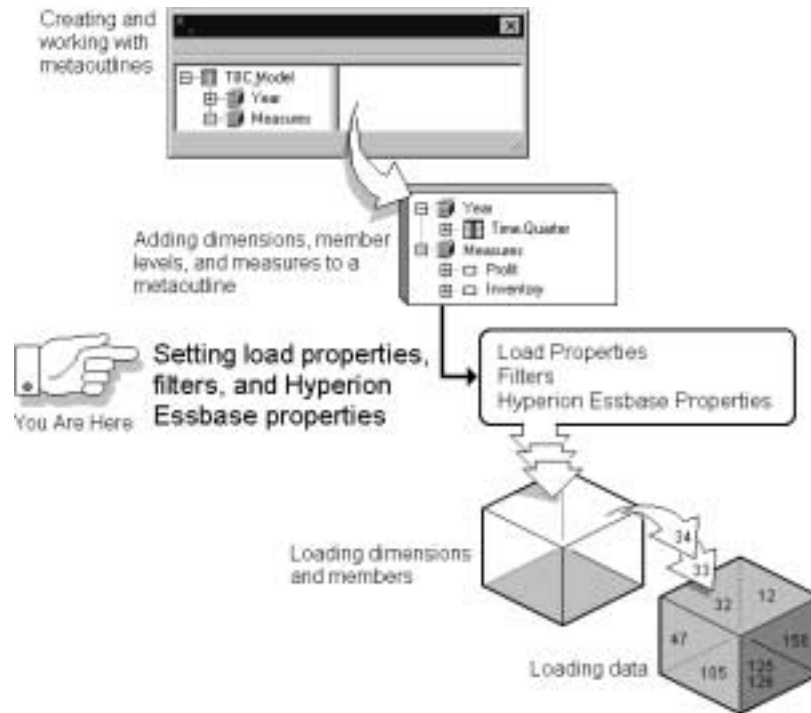


Figure 5-1: Metaoutline Workflow

About Filters

You can filter members and data values when you load members and data values into a Hyperion Essbase outline. For example, assume that a relational data source contains data for 1999 and 2000.

If you want to create a Hyperion Essbase outline that tracks the 1999 data, you create a filter that loads only the members related to 1999.

Working with Filters

Any filter that you create becomes part of an automatically-generated SQL WHERE clause. For example, a typical filter might have the following WHERE clause:

```
SELECT a.FAMILY, a.FAMILYNAME
FROM TBC.PRODUCTDIM a
WHERE (a.FAMILY IN ('200'))
ORDER BY 1 ASC, 2 ASC
```

Note: You can also filter data in an OLAP model. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

You can create filters on one or more member levels or measures of a metaoutline. All filters defined on member levels and measures in a metaoutline are stored in one or more *metaoutline filters*.

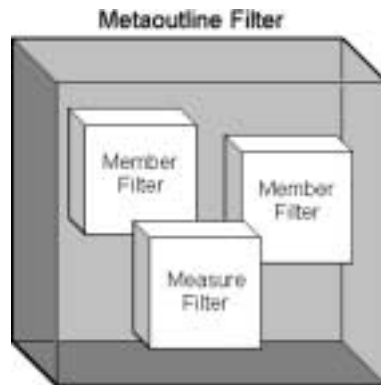


Figure 5-2: Contents of a Metaoutline Filter

About Metaoutline Filters

OLAP Integration Server automatically stores all member level and all measure filters in a metaoutline filter named *Default. Therefore, when you make a new member or measure filter, it will be added to the default filter. You can, however, create one or more additional metaoutline filters and define a specific set of member level and measure filters for each filter. Then you can select a particular metaoutline filter to determine what OLAP Integration Server loads into the Hyperion Essbase database.

Defining multiple metaoutline filters enables you to create multiple Hyperion Essbase databases from a single metaoutline—just select a different metaoutline filter before performing the member and data load. See “Selecting a Metaoutline Filter” on page 5-8.

OLAP Integration Server loads into the Hyperion Essbase database only the members and data values that are specified by the metaoutline filter. For example, you can create a metaoutline filter for each year—a 99 DATALOAD filter and a 00 DATALOAD filter. Similarly, you can create a large_market filter, and build a Hyperion Essbase database that tracks only information for large markets, as illustrated in Figure 5-3.

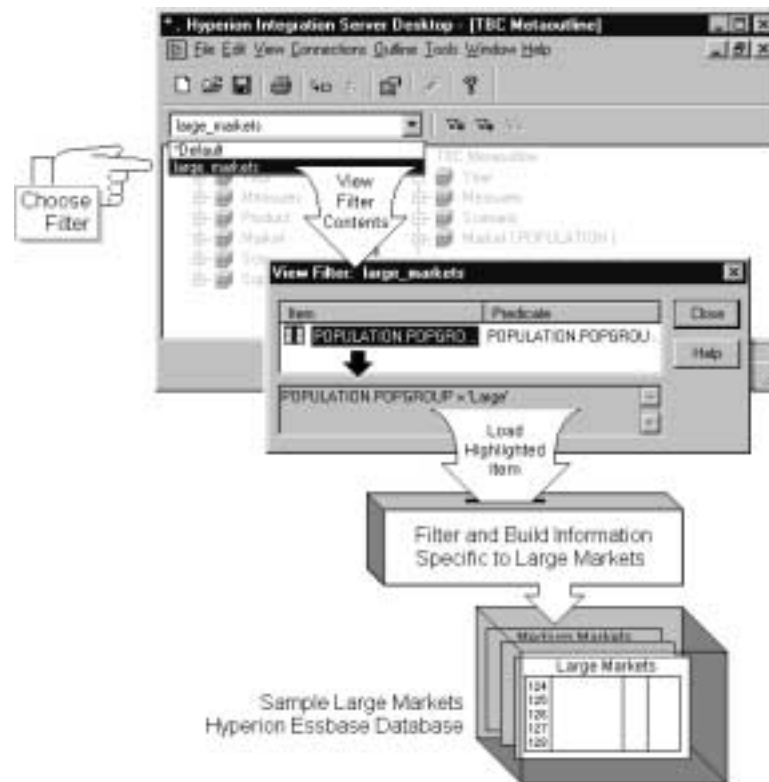


Figure 5-3: Metaoutline Filter Example

About Member Filters

A member filter is applied to member names as the member names are loaded from the external data source into the Hyperion Essbase database.



Figure 5-4: Member Filter Example

You can create a *row filter* on a member name.

OLAP Integration Server applies the *row filter* to each row. The filter is based on a member name. For example, if you filter out all years that are not 1999, OLAP Integration Server loads the 1999 values but not the values from other years, such as 2000.

Note: If you are creating a filter on a shared member, see “Handling Duplicate Member Names” on page 4-6.

When creating a member filter, keep the following guidelines in mind:

- You cannot use substitution variables in a filter.
- When you type the filter syntax, you can use the pattern-matching characters that are supported by the relational database management system (RDBMS) that you are using.
- If you define a filter on a member that you expect to be shared, you must define the same filter on each shared member.
- If you define a filter on a recursive dimension, see “Creating Recursive Dimensions” on page 3-17.

About Measure Filters

A measure filter is applied to data values as the data values are loaded from the external data source to the Hyperion Essbase database. For example, if you specify a filter to load only SALES measures that have an aggregate value greater than 1000, OLAP Integration Server does not load into the Sales member of the Hyperion Essbase database any data values that have an aggregate value equal to or less than 1000. If no data value meets the filtering condition, OLAP Integration Server loads a value of #MISSING into the Hyperion Essbase database.



Figure 5-5: Measure Filter Example

Note: If you are creating a filter on a shared member, see “Handling Duplicate Member Names” on page 4-6.

When creating a measure filter, keep the following guidelines in mind:

- You cannot use substitution variables in a filter.
- When you type the filter syntax, you can use the pattern-matching characters supported by the RDBMS.

Filtering in an OLAP Model or in a Metaoutline

You can filter out unwanted data in an OLAP model or in a metaoutline. In general, you should create high-level filters in an OLAP model and more specific filters in a metaoutline.

Create a filter in an OLAP model to prevent data from appearing in either the OLAP model or the metaoutline; for example, you might want to create an OLAP model that includes only the last two years of data even though the related warehouse contains five years of history. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

Create filters in a metaoutline to create specific Hyperion Essbase databases (for example, apply a member filter to view the data for 1998) or to customize a Hyperion Essbase database for a certain set of users (for example, filter out everything but sales and inventory).

Working with Metaoutline Filters

This topic describes how to use metaoutline filters. It contains the following subtopics:

- “Selecting a Metaoutline Filter” on page 5-8
- “Creating a Metaoutline Filter” on page 5-10
- “Viewing the Contents of a Metaoutline Filter” on page 5-12
- “Viewing the SQL Statement for a Filter” on page 5-19
- “Deleting a Metaoutline Filter” on page 5-13

Selecting a Metaoutline Filter

The currently selected metaoutline filter is the metaoutline filter to which OLAP Integration Server adds new member and measure filters and the one that is used for member and data loads.

The name of the currently selected metaoutline filter is displayed at the top of the metaoutline filter drop-down list. The metaoutline filter drop-down list contains the *Default filter, plus any metaoutline filters that you have created. If you have not created any metaoutline filters, the only filter that you can select is *Default.

► To select a metaoutline filter:

1. Open the metaoutline.

See “Creating or Editing Metaoutlines” on page 2-4.

2. In the OLAP Metaoutline main window, from the drop-down list, select a filter, as shown in Figure 5-6.



Figure 5-6: Setting a Metaoutline Filter

If no filter other than *Default is listed, you may want to define a filter. See “Creating a Metaoutline Filter” on page 5-10.

Creating a Metaoutline Filter

A metaoutline filter contains all member level and measure filters that are defined while it is selected. See “About Metaoutline Filters” on page 5-4.

► To create a new metaoutline filter:

1. Open the metaoutline.

See “Creating or Editing Metaoutlines” on page 2-4.

2. Click the **Add Filter** button, .

The **Enter the new filter's name** dialog box, in which you can name the new filter, is displayed.



Figure 5-7: Creating a New Metaoutline Filter

3. Type the name of the new filter, for example, **large_markets**.

See “About Naming” on page 3-8.

4. Click **OK**.

OLAP Integration Server creates a new metaoutline filter and sets it as the current filter.




Figure 5-8: Viewing the Current Filter Name

5. In the metaoutline, create member and measure filters. These filters will be added to the metaoutline filter that is currently selected in the metaoutline filter drop-down list.

See “Creating a Member Filter Interactively” on page 5-14 and “Creating a Measure Filter Interactively” on page 5-23.

Viewing the Contents of a Metaoutline Filter

You can view all member and measure filters that are defined in the current metaoutline filter.

- ▶ To view the contents of the default metaoutline filter:
 1. Open the metaoutline.
See “Creating or Editing Metaoutlines” on page 2-4.
 2. In the drop-down list at the upper-left side of the screen, where `large_markets` is displayed in Figure 5-8, select `*Default`.
 3. Click the **View Filter** button, .

The **View Filter** dialog box is displayed. The dialog box lists all member and measure filters that are defined in the current metaoutline filter.

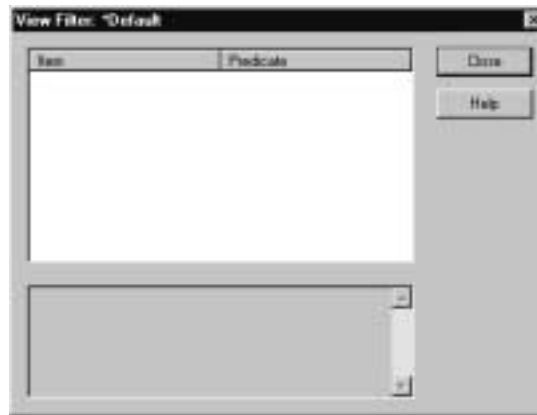


Figure 5-9: Viewing the Contents of the Current Metaoutline Filter

4. To view the conditions defined for a specific filter, select the filter.

In this case, there is no filter to select. After you have built a filter, you will be able to view it in the **View Filter** dialog box, as shown in Figure 5-10.

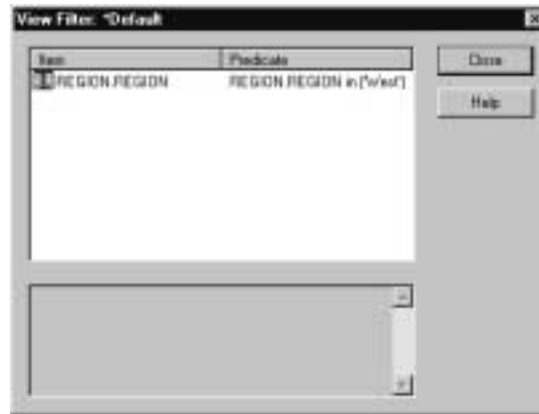



Figure 5-10: Viewing a Filter in the View Filter Dialog Box

5. When you finish viewing the filters, click **Close**.

Deleting a Metaoutline Filter

- To delete a metaoutline filter:
 1. Open the metaoutline.
See “Creating or Editing Metaoutlines” on page 2-4.
 2. From the **Filter** drop-down list, select the filter to delete.
See “Selecting a Metaoutline Filter” on page 5-8.
 3. Click the **Delete Filter** button, .
A dialog box prompts you to confirm deleting the filter.
 4. Click **Yes**.

Creating a Member Filter Interactively

You can select the parts of a member filter interactively from list boxes and dialog boxes, or you can type a member filter. This topic describes how to create a member filter interactively. To type a member filter, see “Typing a Member Filter” on page 5-21.

► To create a member filter interactively:

1. In the metaoutline, select a filter, for example, *Default, in the filter drop down list.

Figure 5-6 displays the metaoutline with *Default selected.

2. In the metaoutline, select the member level to attach to the filter; for example, REGION.REGION.
3. To create a row filter, right-click the member level and, from the pop-up menu, select Filter to display the **Edit Filter** dialog box.
4. Select the **Filters** tab.

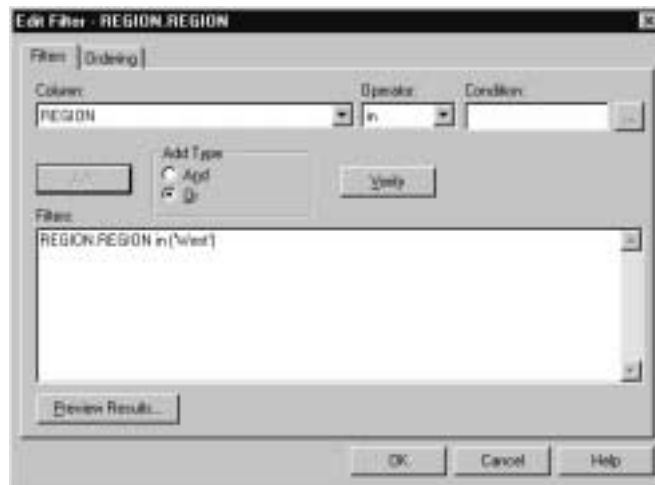


Figure 5-11: Setting a Member Load Filter Interactively

5. From the **Column** drop-down list, select the column to which the condition applies; for example, REGION.
6. From the **Operator** drop-down list, select the operator to use.

Table 5-1: Filter Operators

Operator	Description
in	Sets one or more conditions. For example, REGION in UNITED STATES or EUROPE selects both the UNITED STATES and EUROPE regions.
not in	Opposite of in. For example, REGION not in UNITED STATES selects all regions except for the regions that are part of the United States.
like	Looks for similar characters. For example, if you want to select all products that begin with P and do not want to specify other characters of the product name, you can set the condition, like "P" (Different RDBMSs may use different wildcards).
not like	Opposite of <i>like</i> . Selects conditions that exclude similar characters.
=	Equal to.
<>	Not equal to.
>	Greater than.
>=	Greater than or equal to.
<	Less than.
<=	Less than or equal to.

7. To enter a condition in the **Condition** text box, type the condition; for example, **West**, or complete the following three steps:

- a. To open the **Select Values From** dialog box, click the  button.

The **Select Values From** dialog box lists all possible values for the selected column. OLAP Integration Server retrieves the listed values from the external data source. If the external data source contains many values, a dialog box is displayed. The dialog box asks if you want to view all values.



Figure 5-12: Setting Values for Conditions

- b. Select the value you want to use (such as **West**).

To select multiple values, hold down the **Ctrl** key and select each value.

You can select more than one value only if you have selected **in** or **not in** as the operator. For a description of all operators, see Table 5-1.

- c. Click **OK**.

8. In the **Edit Filter** dialog box, click **Add** to add the condition to the **Filters** list.

9. If you want to define an additional condition, perform the following steps:
 - a. Repeat steps 5 through 7.
 - b. Perform one of the following tasks:
 - If you want the filter to apply when any one of the listed conditions is met, in the **Add Type** box, select **Or**. **Or** is the default.
 - If you want the filter to apply only if all the listed conditions are met, in the **Add Type** box, select **And**.
 - If you want to define the order in which the relational data source executes the multiple conditions, in the **Filters** box, enclose the conditions in parentheses.

How you define ordering depends on the relational data source. Consult the documentation for the relational data source.

➤ To edit and view your filter:

1. If you want to edit a filter, complete these tasks:
 - a. Use the SQL syntax required by the relational data source.
 - b. In the **Edit Filter** dialog box, click **Verify** to check the SQL syntax.
2. Click **Preview Results** to open the **Preview Filter Results** dialog box.
 In the **Preview Filter Results** dialog box, you can view your changes in a sample Hyperion Essbase outline. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.
3. Click **OK**.
4. If you want to view your filter in the metaoutline filter, complete the following tasks:
 - a. Select the filter name, for example, *Default, from the drop-down list in the filter toolbar (see Figure 5-6).

Working with Filters

- b. Follow the procedures in “Viewing the Contents of a Metaoutline Filter” on page 5-12.

For example, the *Default metaoutline filter now contains a member filter, as shown in Figure 5-13.



Figure 5-13: Viewing the New Default Filter

Viewing the SQL Statement for a Filter

- To view the SQL statement for the current *Default filter:
 1. In the metaoutline, select a member in the right frame, for example, REGION.REGION as shown in Figure 5-14.



Figure 5-14: Selecting the Metaoutline Member to View Filter

2. Right-click and select Filter from the drop-down list.
The **Edit Filter** dialog box is displayed.

3. In the **Edit Filter** dialog box shown in Figure 5-15, click **Preview Results**.

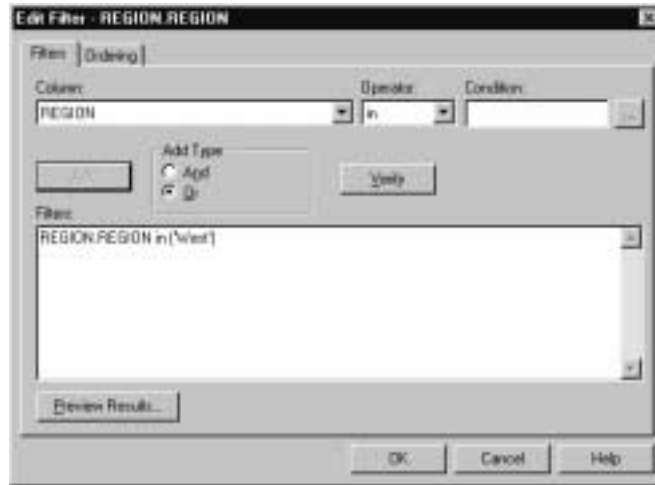


Figure 5-15: Viewing a Filter in the Edit Filter Dialog Box

The **Preview Filter Results** dialog box is displayed, as shown in Figure 5-16. The SQL statement for the filter is displayed in the lower portion of the dialog box.



Figure 5-16: Viewing the SQL Statement in the Preview Filter Results Dialog Box

4. When you finish viewing the **Preview Filter Results** dialog box, click **Close**.
The **Edit Filter** dialog box is displayed.
5. When you finish viewing the **Edit Filter** dialog box, click **OK**.

Typing a Member Filter

You can type a member filter or select the parts of a member filter interactively from list boxes and dialog boxes. This topic describes how to type a member filter. To create a member filter interactively, see “Creating a Member Filter Interactively” on page 5-14. To add this filter to the default filter, be sure the default filter is selected in the filter selection drop-down list at the upper left of the metaoutline (see Figure 5-6).

- To type a member filter:
1. In the metaoutline, select the member level to attach to the filter; for example, REGION.REGION.
 2. To create a row filter, right-click the member level and, from the pop-up menu, select Filter.
The **Edit Filter** dialog box is displayed.

3. Select the **Filters** tab.

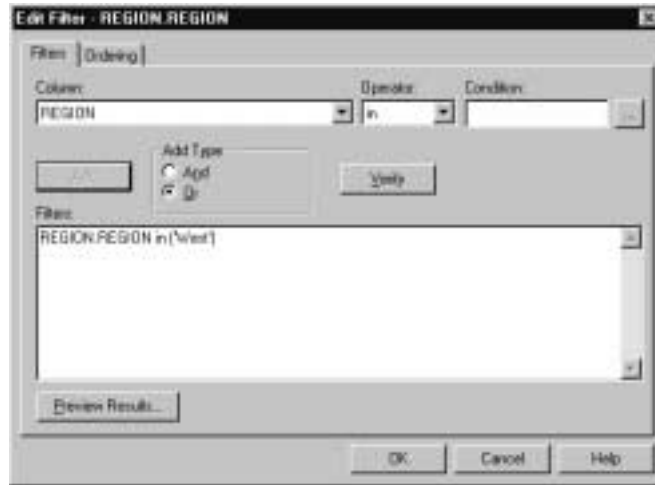


Figure 5-17: Typing a Member Load Filter

To enter a filter, in the **Filters** box, type the filter information according to the following syntax:

TableName.ColumnName Operator Condition

For example, you might type a row filter as follows:

REGION.REGION in ('West')

For a list of valid operators, see the documentation for the data source.

4. Click **Verify** to check the SQL syntax.
5. If you want to define an additional condition on the filter, complete the following steps:
 - a. Repeat steps 4 and 5.
 - b. Review step 8b of the previous procedure, “Creating a Member Filter Interactively” on page 5-14, and complete the steps appropriate to your situation.

6. Click **Preview Results** to open the **Preview Filter Results** dialog box.

In the **Preview Filter Results** dialog box, you can view your changes in a sample Hyperion Essbase outline. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.

7. Click **OK**.

Creating a Measure Filter Interactively

You can select the parts of a measure filter interactively from list boxes and dialog boxes, or you can type a measure filter. This topic describes how to create a measure filter interactively. To type a measure filter, see “Typing a Measure Filter” on page 5-26.

- To create a measure filter interactively:
 1. In the metaoutline, select the measure to apply to the filter; for example, **OPENINGINVENTORY**.
 2. Right-click the measure and, from the pop-up menu, select **Filter**.
The **Measure Properties** dialog box is displayed.
 3. Select the **Filters** tab.

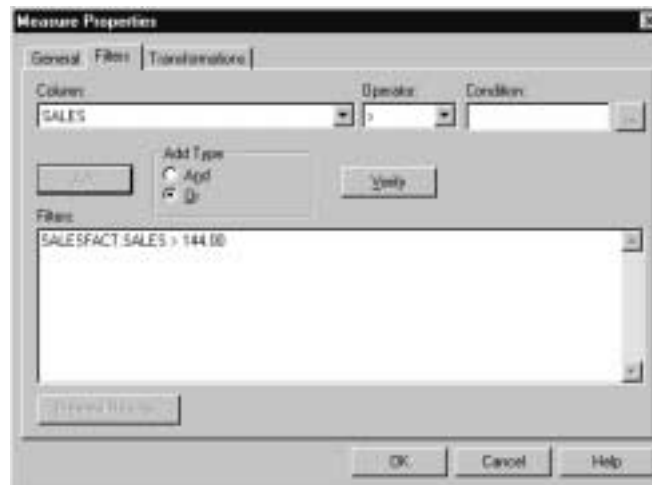



Figure 5-18: Creating a Measure Filter Interactively

Working with Filters

4. From the **Column** drop-down list, select the column to which the condition applies; for example, SALES.
5. From the **Operator** drop-down list, select the operator to use.
For a description of all operators, see Table 5-1.
6. To enter a condition, in the **Condition** text box, type the condition; for example, **114**, or complete the following steps:

- a. Click the  button to open the **Select Values From** dialog box.

The **Select Values From** dialog box lists all possible values for the selected column. OLAP Integration Server retrieves the values from the external data source. If the external data source contains many values, a dialog is displayed. The dialog box asks if you want to view them all.

Note: When the database contains hundreds of thousands of records, viewing the values for these records in the **Select Values from** dialog box is impractical. In this case, type the value of interest directly into the **Condition** text box.

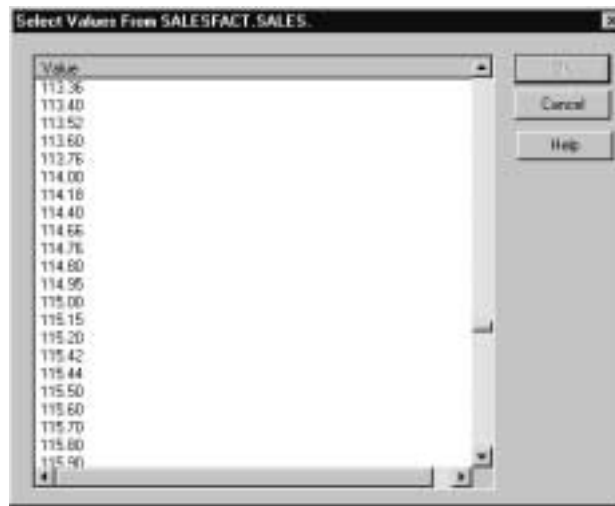


Figure 5-19: Setting Values for Conditions

- b. Select the desired value (such as 114).

To select multiple values, hold down the **Ctrl** key as you select each value.

You can select more than one value only if you have selected **in or not in** as the operator. For a description of all operators, see Table 5-1.

- c. Click **OK**.

7. In the **Measure Properties** dialog box, click **Add** to add the condition to the **Filters** list.

8. To define an additional condition on the filter, complete the following steps:

- a. Repeat steps 5 through 7.

- b. Perform one of the following tasks:

- If you want the filter to apply when any one of the listed conditions is met, in the **Add Type** box, select **Or**. **Or** is the default.
- If you want the filter to apply only if all the listed conditions are met, in the **Add Type** box, select **And**.
- If you want to define the order in which the relational data source executes the multiple conditions, in the **Filters** box, enclose the conditions in parentheses.

How you define ordering depends on the relational data source. Consult the documentation for the relational data source.

9. If you want to edit a filter, complete these tasks:

- a. Use the SQL syntax required by the relational data source.

- b. Click **Verify** to check the SQL syntax.

10. Click **Preview Results** to open the **Preview Filter Results** dialog box.

In the **Preview Filter Results** dialog box, you can view your changes in a sample Hyperion Essbase outline. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.

11. Click **OK**.

Note: If the *Default metaoutline filter is still selected, this filter is automatically added to the *Default metaoutline filter. For more information, see step 1 of “Creating a Member Filter Interactively” on page 5-14.

Typing a Measure Filter

You can type a measure filter or select the parts of a measure filter interactively from list boxes and dialog boxes. This topic describes how to type a measure filter. To create a measure filter interactively, see “Creating a Measure Filter Interactively” on page 5-23.

- To type a measure filter:
 1. Select the measure to apply to the filter; for example, SALES.
 2. Right-click the measure and, from the pop-up menu, select Filter.
The **Measure Properties** dialog box is displayed.
 3. Select the **Filters** tab.

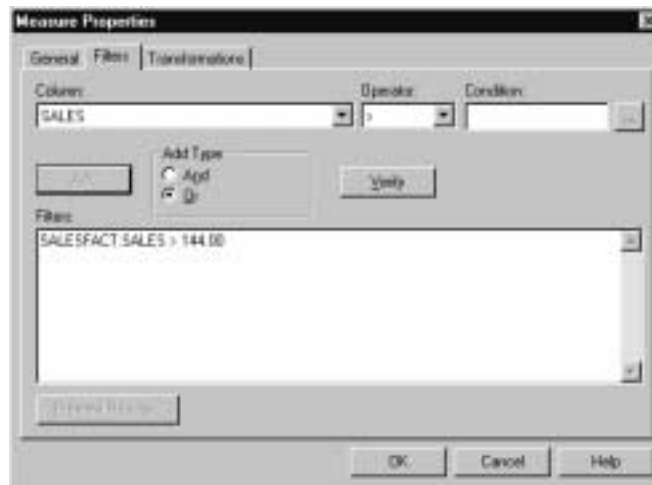


Figure 5-20: Typing a Measure Filter

4. In the **Filters** box, type the filter information according to the following syntax:

TableName.ColumnName Operator Condition

For example, type the following syntax:

SALESFACT.SALES > 114

For a list of valid operators, see the documentation for the data source.

5. Click **Verify** to check the SQL syntax.
6. To define an additional condition on the filter, complete the following steps:
- a. Repeat steps 4 and 5.
 - b. Perform one of the following tasks:
 - If you want the filter to apply when any one of the listed conditions is met, in the **Add Type** box, select **Or**. **Or** is the default.
 - If you want the filter to apply only if all the listed conditions are met, in the **Add Type** box, select **And**.
 - If you want to define the order in which the relational data source executes the multiple conditions, in the **Filters** box, enclose the conditions in parentheses.

How you define ordering depends on the relational data source. Consult the documentation for the relational data source.

7. Click **Preview Results** to open the **Preview Filter Results** dialog box.

In the **Preview Filter Results** dialog box, you can view your changes in a sample Hyperion Essbase outline. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.

8. Click **OK**

Note: If the *Default metaoutline filter is still selected, the new filter is automatically added to the *Default metaoutline filter. For more information, see step 1 of “Creating a Member Filter Interactively” on page 5-14.

Creating an Attribute Filter Interactively

Creating an attribute filter follows the same pattern as creating a member filter. To create a member filter interactively, see “Creating a Member Filter Interactively” on page 5-14. To type a member filter, see “Typing a Member Filter” on page 5-21. This topic describes how to create an attribute filter interactively.

- To create an attribute filter:
 1. Select the metaoutline filter you want to use from the filters drop-down list in the upper left corner of the Hyperion Integration Server Desktop.

For example, in Figure 5-21, the filter `large_markets` is selected. (For information on selecting a metaoutline filter, see “Selecting a Metaoutline Filter” on page 5-8).

Select the member you want to use in the attribute dimension. For example, in Figure 5-21, `POPULATION.POPGROUP` under the `POPULATION` attribute dimension is selected.



Figure 5-21: Choosing the Attribute Member for a Filter

2. Right-click the selected member, and select Filter from the pop-up menu.
The **Edit Filter** dialog box is displayed.
3. Select the **Filters** tab, shown in Figure 5-22.
4. Select an operator from the operator drop-down menu, for example, =.

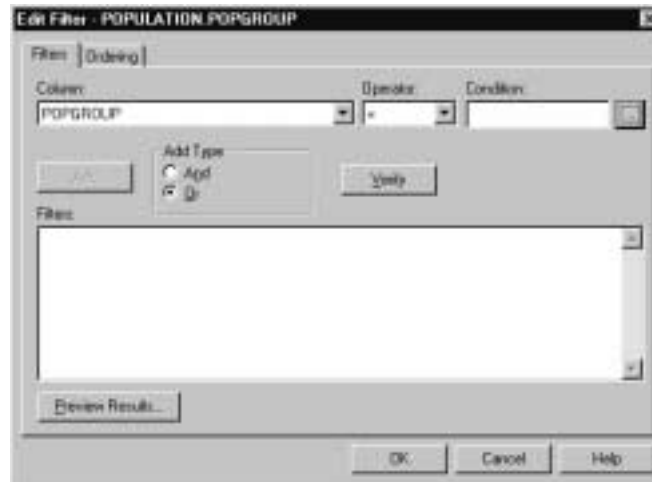



Figure 5-22: Attribute Filter in the Edit Filter Dialog Box

5. Click the conditions button, , to select the condition.
The **Select Values From** dialog box is displayed (see Figure 5-23).

6. In the **Select Values From** dialog box, select the condition you want and click **OK**.

For example, Figure 5-23 shows the **Select Values From** dialog box with the condition **Large** selected.

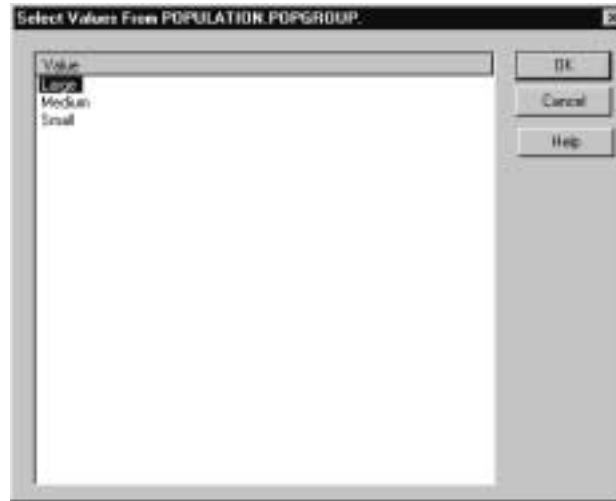



Figure 5-23: Selecting Value for Attribute Conditions

7. Click **Add** in the **Edit Filter** dialog box.
8. Click **OK**.

Note: You can create a filter on any member of the attribute dimension. For example, you can create a filter on POPULATION.POPULATION that appears below POPULATION.POPGROUP in the sample metaoutline.

- To view the new filter:
 1. Select the metaoutline filter you want to view from the metaoutline filter drop-down list (see “Viewing the Contents of a Metaoutline Filter” on page 5-12).
 2. Click the **View Filter** button, , on the metaoutline tool bar.
 3. Select the filter, for example, **POPULATION.POPGROUP**, and the filter criteria appear in the lower box of the **View Filter** dialog box.

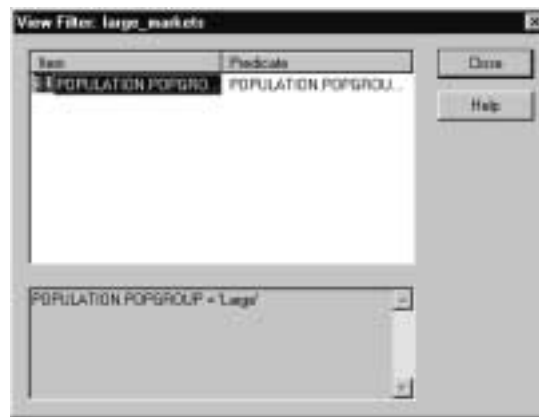


Figure 5-24: New Metaoutline Filter Containing Attribute Member Filter

4. When you finish viewing the **View Filter** dialog box, click **Close**.

Working with Filters

Setting Hyperion Essbase Properties

Hyperion Integration Server enables you to set the properties of the dimensions and members of a Hyperion Essbase outline; for example, you can determine how members consolidate into a dimension. Changes that you make to the properties of dimensions, members, and measures do not affect the Hyperion Essbase database until you perform a member or data load.

This chapter describes the Hyperion Essbase properties that you can set and outlines the methods that you use to set them. For more detailed descriptions and examples of Hyperion Essbase properties, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 6-2
- “Specifying Dimension and Member Properties” on page 6-3
- “Specifying Account Properties” on page 6-19
- “Specifying UDAs” on page 6-25
- “Specifying Formulas” on page 6-27
- “Viewing Hyperion Essbase Properties” on page 6-29

Note: For information about setting load properties, see Chapter 4, “Setting Load Properties.” For information about setting filters, see Chapter 5, “Working with Filters.”

Understanding the Metaoutline Workflow

Figure 6-1 illustrates the workflow for creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following processes:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and members from the metaoutline into a Hyperion Essbase outline
- Loading the data specified in the metaoutline into a Hyperion Essbase database

This chapter focuses on setting Hyperion Essbase properties. You can set Hyperion Essbase properties before or after setting load properties and filters.

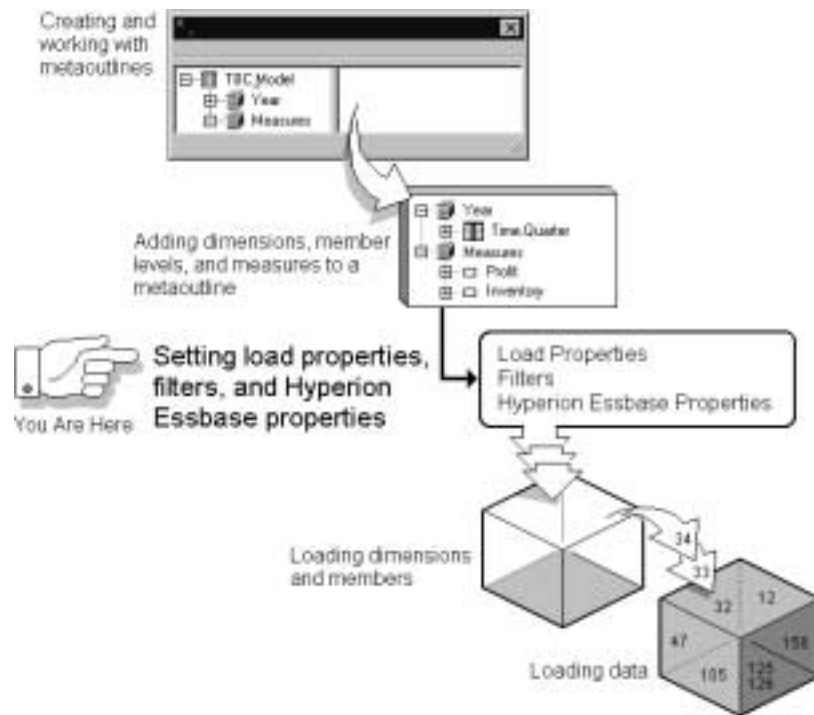


Figure 6-1: Metaoutline Workflow

Specifying Dimension and Member Properties

This topic describes how to set Hyperion Essbase properties for dimensions and for member levels. It contains the following subtopics:

- “About Dimension Types” on page 6-3
- “Setting a Dimension Type” on page 6-7
- “Specifying Dimension Storage Properties” on page 6-8
- “Specifying Data Storage Properties for Dimensions and Members” on page 6-9
- “Setting Aliases for Dimensions and Members” on page 6-12
- “Specifying Two-Pass Calculation on Dimensions and Members” on page 6-14
- “Specifying Member Consolidation Properties” on page 6-16
- “Setting Members as Dynamic Time Series” on page 6-18

Note: You cannot use the member levels of a metaoutline to create partitions.

About Dimension Types

When you tag a dimension as a specific Hyperion Essbase type, the dimension can access the built-in functionality that is designed for that type. For example, if you define a dimension type as Accounts, you can specify accounting operations for the member levels of the dimension. By default, OLAP Integration Server tags dimensions tagged as Time or Accounts in the OLAP model as Time or Accounts in the metaoutline. Attribute-enabled columns in the OLAP model are tagged as attribute dimension in the metaoutline. OLAP Integration Server tags all other dimensions as standard.

Note: Naming a dimension Time or Accounts does not automatically tag the dimension as type time or type accounts.

Measures values map to the accounts dimension in the OLAP model, which maps to the measure dimension in the metaoutline, which in turn maps to the accounts dimension in the Hyperion Essbase outline.

Table 6-1 describes each dimension type.

Table 6-1: Dimension Types

Type	Description
standard	The dimension has no special type.
accounts	<p>Contains items that you want to measure, such as profit or inventory. The dimension tagged as Measures in a metaoutline corresponds to the dimension tagged as Accounts in the related Hyperion Essbase database.</p> <p>Follow these rules when tagging a Measures (Accounts) dimension in a metaoutline:</p> <ul style="list-style-type: none"> • You can tag only one dimension in a metaoutline as Measures (Accounts). • When you tag a dimension as Measures (Accounts), all members in the dimension inherit the Accounts property. • You can create a metaoutline that does not have a dimension tagged as Measures (Accounts). • Hyperion Essbase calculates the dimension tagged as Time and the dimension tagged as Accounts before it calculates other dimensions in the database. For more information on calculation, see the <i>Hyperion Essbase OLAP Server Database Administrator's Guide</i>. • To calculate members of the dimension tagged as Accounts on the second pass through the Hyperion Essbase outline, see “Specifying Two-Pass Calculation on Dimensions and Members” on page 6-14.

Table 6-1: Dimension Types (Continued)

Type	Description
time	<p data-bbox="581 464 1317 552">Describes how often you collect and update data. The dimension tagged as Time enables several accounts dimension functions, such as first time balance and last time balance.</p> <p data-bbox="581 569 1203 625">Follow these rules when tagging a dimension as Time in a metaoutline:</p> <ul data-bbox="581 642 1317 1312" style="list-style-type: none"> <li data-bbox="581 642 1240 669">• You can tag only one dimension in a metaoutline as Time. <li data-bbox="581 680 1317 737">• When you tag a dimension as Time, all members in the dimension inherit the time property. <li data-bbox="581 747 1317 869">• You can create multiple hierarchies inside the dimension tagged as Time; for example, you can specify a Year, Quarter, Month hierarchy and a Year, Season hierarchy in the same time dimension. <li data-bbox="581 886 1265 942">• You can create a metaoutline that does not have a dimension tagged as Time. <li data-bbox="581 959 1292 1016">• You can add time members to a dimension that is not tagged as Time. <li data-bbox="581 1033 1317 1178">• Hyperion Essbase calculates the dimension tagged as Time and the dimension tagged as Accounts before it calculates other dimensions in the database. For more information on calculation, see the <i>Hyperion Essbase OLAP Server Database Administrator's Guide</i>. <li data-bbox="581 1194 1317 1312">• To calculate members of the dimension tagged as Time on the second pass through the Hyperion Essbase outline, see “Specifying Two-Pass Calculation on Dimensions and Members” on page 6-14.


Table 6-1: Dimension Types (Continued)

Type	Description
attribute	<p>Describes characteristics, or attributes, of dimensions and members. Follow these rules when using an attribute dimension in a metaoutline:</p> <ul style="list-style-type: none"> • You can build attribute dimensions in the metaoutline based on attribute-enabled columns in the OLAP model. • Each attribute dimension is associated with a base dimension. This is done automatically when the dimension is created. • You can create a metaoutline that has no attribute dimensions. • To add attribute dimensions and members, see Chapter 3, “Adding Dimensions, Member Levels, and Measures to a Metaoutline,” and Appendix A, “Creating a Sample Metaoutline Containing Attribute Dimensions.”

Setting a Dimension Type

You can determine whether a dimension is tagged as standard, accounts, or time by setting its dimension type.

- To set a dimension type:
 1. In the metaoutline, select the dimension for which you want to set the dimension type.
 2. Right-click the dimension and, from the pop-up menu, select Properties to open the **Dimension Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button, , from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-2: Setting a Dimension Type

4. From the **Dimension Type** group, select the type of dimension.
For more information about dimension types, see Table 6-1 and the *Hyperion Essbase OLAP Server Database Administrator's Guide*.
5. Click **OK**.

Specifying Dimension Storage Properties

You can determine whether a dimension is stored as sparse or dense.

- Use *sparse storage* for a dimension in which a low percentage of available data positions is filled. For example, the sample application TBC Metaoutline includes the Product and Market dimensions. Product represents product units and Market represents geographical regions in which products are sold. Because not every product is sold in every market, Product and Market are sparse dimensions.
- Use *dense storage* for a dimension in which a high percentage of available data positions are filled. For example, in the sample application TBC Metaoutline, time data exists for almost all products in all markets, so Year is a dense dimension.

For more information about dimension storage properties, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Note: In a metaoutline, you must tag at least one dimension as Dense and one dimension as Sparse.

➤ To set a dimension storage property:

1. In the metaoutline, select a dimension; for example, Measures.
2. Right-click the dimension and, from the pop-up menu, select Properties to open the **Dimension Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-3: Setting Dense or Sparse Dimensions

4. From the **Dimension Storage** group, select **Dense** or **Sparse**.
5. Click **OK**.

Specifying Data Storage Properties for Dimensions and Members

You can determine how and when Hyperion Essbase stores data values for a dimension and for a member. For example, you can tell Hyperion Essbase to calculate the value for a member only when a user requests it and then to discard the value. Table 6-2 identifies and describes each storage property.

Table 6-2: Data Storage Properties

Property	Description
Store data	Stores the data value with the member. This property is the default.
Dynamic Calc And Store	Calculates the data value when a user requests it and then stores the data value.
Dynamic Calc	Calculates the data value when a user requests it and then discards the data value.
Never share	Does not allow members to be shared implicitly.
Label only	Creates a member that is used only for navigation. A label only member contains no data value.

➤ To set a data storage property:

1. In the metaoutline, select a dimension or a member level; for example, Measures.
2. Right-click the dimension or the member level and, from the pop-up menu, select Properties to open the **Dimension Properties** or **Measure Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-4: Setting Storage Properties

4. From the **Data Storage** group, select the kind of storage to use.

Note: Do not set a dimension or member as shared in the **Dimension Properties** or **Member Properties** dialog box, unless you are certain that *all* members of the relevant dimension are shared members.

For more information about data storage properties, see Table 6-2 and the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

5. Click **OK**.

Setting Aliases for Dimensions and Members

You can assign an *alias* (alternate name) to a dimension and to a member. The use of aliases improves the readability of Hyperion Essbase outlines and reports. For example, members in the Hyperion Essbase outline created by the Product dimension of TBC Metaoutline are identified both by product codes, such as 100-20, and by descriptive aliases, such as Kool Cola.

Because a member level in a metaoutline corresponds to a column of member names in the related OLAP model, an alias in a metaoutline must correspond to a column of alias names in the related OLAP model. TBC Metaoutline, for example, tells OLAP Integration Server to retrieve member names from the FAMILYID column and member aliases from the SKUNAME column.

Table 6-3 shows sample names and aliases in the TBC Metaoutline.

Table 6-3: Sample Member Names and Aliases

FAMILYID	SKUNAME
100-10	Cola
100-20	Diet Cola
100-30	Caffeine Free Cola

You must observe the following rules when creating an alias:

- You cannot perform transformations on members to which you assign aliases. Use OLAP Model to perform needed transformations. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.
- If the column in the relational data source that contains the alias names has any value with more than 80 characters (the maximum allowed in Hyperion Essbase naming rules), you can use the SUBSTRING function in OLAP Model to extract the 80 or fewer characters that you want to use in the alias name. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.


- You must create the alias column in the external data source in advance. See the *Hyperion Integration Server Desktop OLAP Model User's Guide* and the documentation for the external data source.
 - If you want to assign aliases to recursive members, the alias column must be in the same physical table as the recursive members. For more information on recursive members, see “Creating Recursive Dimensions” on page 3-17.
- To assign an alias to a dimension or member:
1. In the metaoutline, select the dimension or member level.
 2. Right-click the dimension or member level and, from the pop-up menu, select Properties to open the **Dimension Properties** or **Member Properties** dialog box.
- You can also open the dialog box by clicking the **View Properties** button,  , from the toolbar or by selecting Properties from the Edit menu.
3. Select the **Member Info** tab.



Figure 6-5: Assigning Aliases

4. Perform one of the following tasks:
 - If you are assigning an alias to a metaoutline member level, from the **Alias** drop-down list, select the column that contains the aliases.
 - If you are assigning an alias to a dimension or a user-defined member, in the **Alias** text box, enter the alias name.

When you assign an alias to a dimension or a user-defined member, the **Alias** drop-down list box changes to a text box.

See “About Naming” on page 3-8.

Note: Do not set a dimension or member as shared in the **Dimension Properties** or **Member Properties** dialog box unless you are certain that *all* members of the relevant dimension are shared members.

5. Click **OK**.

Specifying Two-Pass Calculation on Dimensions and Members

By default, Hyperion Essbase calculates outlines from the bottom up—calculating first the values for the children and second the value for the parent. Sometimes, however, the value of a child depends upon the value of the parent or the value of another member in the Hyperion Essbase outline. To obtain the correct values for dependent dimensions and members, Hyperion Essbase must calculate the outline and then recalculate the dimensions and members that are dependent on the calculated values of other dimensions and members. Dimensions and members that are calculated on the second pass through the Hyperion Essbase outline are called *two-pass calculations*.

For example, to calculate the ratio between Sales and Margin, Hyperion Essbase needs to calculate Margin (a parent member based on its children, including Sales). To ensure that the ratio calculation is based on a newly calculated Margin figure,

tag the Margin% ratio member as a two-pass calculation. Hyperion Essbase calculates the database and then recalculates the Margin% member. The second calculation produces the correct result.

Note: Although two-pass calculation is a property that you can give to any member, it works only on members of Accounts dimensions, Dynamic Calc members, and Dynamic Calc and Store members. If you assign two-pass calculation to other members, Hyperion Essbase ignores it.

- To tag a dimension or member to use two-pass calculation:
 1. In the metaoutline, select the dimension or member level.
 2. Right-click the dimension or member level and, from the pop-up menu, select Properties to open the **Dimension Properties** or **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-6: Specifying Two-Pass Calculation

4. Select Two Pass Calculation.

For more information about two-pass calculation, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

5. Click OK.

Specifying Member Consolidation Properties

Member consolidation properties determine how children roll up into their parents. By default, new members are given the addition (+) operator, meaning that Hyperion Essbase adds the members. For example, Hyperion Essbase adds January, February, and March figures and stores the result in their parent, Qtr1. Table 6-4 describes each consolidation operator.

Table 6-4: Member Consolidation Properties

Operator	Description
+	Adds the member to the result of calculations performed on other members. This operator is the default operator.
-	Multiplies the member by -1 and then adds the result to the result of calculations performed on other members.
*	Multiplies the member by the result of calculations performed on other members.
/	Divides the member into the result of calculations performed on other members.
%	Divides the member into the result of previous calculations performed on other members and multiplies the result by 100 to yield a percentage value.
~	Does not use the member in the consolidation to its parent.
From Database Column	Uses the consolidation value (+, -, *, /, %, ~) of the selected column of the data source. This option is not selectable for measure columns in the metaoutline.

- To set the consolidation operator of a member:
 1. In the metaoutline, select a member level; for example, PRODUCT_DESC.
 2. Right-click the member level and, from the pop-up menu, select Properties to open the **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-7: Choosing Member Consolidation

4. From the **Consolidation** group, select the appropriate consolidation operator.
For more information about member consolidation, see Table 6-4 and the *Hyperion Essbase OLAP Server Database Administrator's Guide*.
5. Click **OK**.

Setting Members as Dynamic Time Series

You can calculate period-to-date values dynamically in a Hyperion Essbase database by tagging specific members as Dynamic Time Series. For example, you can enable dynamic calculation of Hyperion Essbase quarter-to-date values by setting the Quarter member of the Year dimension of the related metaoutline as Dynamic Time Series. For more information about Dynamic Time Series members, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Note: To tag values in a metaoutline as Dynamic Time Series, you must set the date-part of the column that contains the time data of the related OLAP Model. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

Note: Dynamic Time Series members can not be at level 0 (leaf level) of the metaoutline.

► To tag a member as Dynamic Time Series:

1. In the metaoutline, select the member level; for example, Month.
2. Right-click the member level and, from the pop-up menu, select Properties to open the **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Member Info** tab.



Figure 6-8: Setting a Member as Dynamic Time Series

4. Select **Dynamic Time**.
5. Click **OK**.

Specifying Account Properties

Only the dimension tagged as accounts and the members of the dimension tagged as accounts can access the built-in accounting functionality provided by Hyperion Essbase. This topic contains the following subtopics that describe accounts dimension properties:

- “Specifying Time Balance Properties” on page 6-20
- “Specifying Variance Reporting Properties” on page 6-23

Specifying Time Balance Properties

By default, Hyperion Essbase bases the calculation of a parent of the Year dimension on the consolidation properties and formulas of the children of the parent. For example, the Qtr1 member is, by default, the sum of the children January, February, and March.

If, instead of a sum, you need a beginning, ending, or average value for a time period, you can select the appropriate time balance property for the appropriate member of the Accounts dimension. When you set a time balance property on a member of the Accounts dimension, the property affects how Hyperion Essbase calculates the accounts member across the Year dimension.

For example, if you want Opening Inventory for Qtr1 to reflect the beginning inventory value for Qtr1, set a time balance property of first on the accounts member Opening Inventory. When Opening Inventory is calculated across the Year dimension, Qtr1 becomes equal to Jan (the first child in the Qtr1 hierarchy).

Table 6-5 summarizes time balance properties.

Table 6-5: Time Balance Properties

Property	Description
None	This property is the default value. When you set the time balance property as None, Hyperion Essbase rolls up parents in the time dimension in the usual way—the value of a parent is based on the formulas and consolidation properties of children of the parent.
First	Set the time balance as First when you want the parent value to represent the value of the first member in the branch (often at the beginning of a time period).
Last	Set the time balance as Last when you want the parent value to represent the value of the last member in the branch (often at the end of a time period).
Average	Set the time balance as Average when you want the parent value to represent the average of the children values.

If you set the time balance as First, Last, or Average, you must set the skip property to tell Hyperion Essbase what to do when it encounters missing values or values of 0. Table 6-6 describes each skip property.

Table 6-6: Skip Properties

Property	Description
None	Does not skip data when calculating the parent value. This property is the default value. However, if Hyperion Essbase encounters #MISSING data when calculating an average, it does not divide by the total number of members. It divides by the number of members with data values. Therefore, setting the skip property to None or #MISSING does not affect average (but does affect first and last).
Missing	Skips #MISSING data when calculating the parent value.
Zeros	Skips data that equals zero when calculating the parent value.
Missing and zeros	Skips both #MISSING data and data that equals zero when calculating the parent value.

- To set a time balance property for a member:
1. In the metaoutline, select the member level; for example, OPENINGINVENTORY.
 2. Right-click the member level and, from the pop-up menu, select Properties to open the **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Account Info** tab.

If the options in the **Account Info** tab are not selectable, the dimension that the member belongs to is not tagged as Accounts. To tag a dimension as Accounts, see “Setting a Dimension Type” on page 6-7.

The **Account Info** tab is not selectable for non-account type dimensions and members.



Figure 6-9: Setting Time Balance Properties

- 4.** From the **Time Balance** group, select the property that you want to use; for example, **First**.
- 5.** If you set the time balance as First, Last, or Average, set the appropriate **Skip** property.
- 6.** Click **OK**.

Specifying Variance Reporting Properties

Variance reporting properties determine how Hyperion Essbase calculates the difference between actual and budget data for a member whose formula includes an @VAR or an @VARPER function. Any member that represents an expense to the company requires an expense property. For more information about functions in formulas, see “Specifying Formulas” on page 6-27.

Table 6-7 describes the two variance reporting properties: expense and non-expense.

Table 6-7: Variance Reporting Properties

Property	Description
Expense	For expense items, actual expenses should be lower than budgeted expenses. When actual expenses are greater than budgeted expenses, variance is negative. The @VAR function calculates BUDGET - ACTUAL. For example, if budgeted expenses are \$100 and actual expenses are \$110, the variance is -10.
Non-expense	For non-expense items, such as sales, actual should be higher than budget. When actual is less than budget, variance is negative. The @VAR function calculates ACTUAL - BUDGET. For example, if budgeted sales are \$100 and actual sales are \$110, the variance is 10. By default, members are non-expense.

- To set the variance reporting property of a member:
 1. In the metaoutline, select the member level; for example, MARKETING.
 2. Right-click the member level and, from the pop-up menu, select Properties to open the **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Account Info** tab.

If the options in the **Account Info** tab are not selectable, the dimension that the member belongs to is not tagged as Accounts. To tag a dimension as Accounts, see “Setting a Dimension Type” on page 6-7.

The **Account Info** tab is not selectable for non-account type dimensions and members.



Figure 6-10: Tagging Members as Expense or Non-Expense

4. From the **Variance Reporting** group, select **Non Expense** or **Expense**.

For more information about variance reporting, see Table 6-7 and the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

5. Click **OK**.

Specifying UDAs

You can create your own UDAs (user-defined attributes) for members. A *UDA* is a word or phrase about a member. For example, you might create a UDA called Big Market to identify any market that sells more than a certain amount. Use UDAs in calculation scripts and report scripts. For a comparison of UDAs and attributes, see “Comparison of UDAs and Attributes” on page 3-25. For more information, see the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.

Understanding Rules for UDAs

When creating UDAs, keep the following guidelines in mind:

- You can define multiple UDAs on a single member.
- You cannot define the same UDA twice on the same member.
- You can define the same UDA on different members.
- A UDA can use the same name as a member, alias, level, or generation. When you name UDAs, follow the same naming rules as for members. See “About Naming” on page 3-8.
- If you are using DB2 OLAP server, you can specify an anchor dimension by setting a UDA of RELANCHOR on the appropriate dimension.
- You cannot create a UDA on Hyperion Essbase shared members.
- A UDA applies to only a specified member. Descendants and ancestors of the member do not automatically receive the UDA.

Defining a UDA

► To define a UDA:

1. In the metaoutline, select the member level for which to define the UDA; for example, REGION.
2. Right-click the member level and, from the pop-up menu, select Properties to open the **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **UDAs** tab.




Figure 6-11: Defining UDAs

4. If you want to define the UDA, select **Value** and in the **Value** text box, enter the UDA.

5. If you want OLAP Integration Server to retrieve the UDA from a column in the relational data source, select **From Database Column** and, from the drop-down list, select the desired column.

If the UDA in the relational data source has more than 80 characters, you can use the SUBSTRING function in OLAP Model to extract the characters (a maximum of 80) that you want to use in the UDA. See the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

6. Click **Add to List** to add the UDA to the **Existing UDAs** list.
The **Existing UDAs** box displays all UDAs for the selected member.
7. If you want to delete a UDA, in the **Existing UDAs** list, select the UDA, and click the **Delete** button, .
8. If you want to add another UDA to the member, repeat steps 4 through 6.
9. Click **OK**.

Specifying Formulas

You can associate formulas with dimensions, user-defined members, and measures. A *formula* determines how Hyperion Essbase calculates data in a Hyperion Essbase database. For more information about formulas, see the *Hyperion Essbase OLAP Server Database Administrator's Guide* and the functions section of the *Hyperion Essbase Quick Technical Reference*.

- To add a formula to a dimension or member:
 1. In the metaoutline, select the dimension or member level to which to add the formula; for example, PROFIT%.
 2. Right-click the member level and, from the pop-up menu, select Properties to open the **Dimension Properties** or **Member Properties** dialog box.

You can also open the dialog box by clicking the **View Properties** button,



, from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Formula** tab.



Figure 6-12: Typing Formulas

4. In the edit field, enter the formula. Be sure to end the formula with a semicolon (;); for example, **PROFIT % SALES;**

Note: The % sign in Hyperion Essbase formulas performs percentage operations.

5. Click **OK**.

OLAP Integration Server does not validate the formula. Errors in the formula are not displayed in the log file until OLAP Integration Server calculates the Hyperion Essbase database. See the *Hyperion Integration Server System Administrator's Guide* for more information on the OLAP Integration Server log file. For more information about formula options and built-in functions, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Viewing Hyperion Essbase Properties

You can customize the view of the OLAP Metaoutline main window to display Hyperion Essbase properties in either columnar or condensed format. Columnar format places Hyperion Essbase properties under column headings, and condensed format places properties in parentheses and to the right of the dimension or member name. Default Hyperion Essbase settings are not displayed.

- To display Hyperion Essbase properties in columnar format:
 1. In the metaoutline, select a dimension, member level, or outline; for example TBC Metaoutline.
 2. Select View > Display Essbase Properties.
 3. If Display as Columns is not selected, select it.
The columnar format view is the default.
 4. To determine which properties to display, from the submenu, select any of the following items:
 - **Consolidation Objects:** Displays consolidation properties, such as +, -, and %.
 - **Formula Objects:** Displays the assigned formula.
 - **Dimension Type:** Displays the dimension type: time or accounts.
 - **Alias:** Displays aliases.
 - **Member Properties:** Displays member properties, such as storage type, two-pass calculation, dynamic time, time balance and variance reporting properties, and UDAs.

The metaoutline is displayed in columnar format and the properties that you selected are displayed.

Setting Hyperion Essbase Properties

5. If you want to display all Hyperion Essbase properties, select **All**.

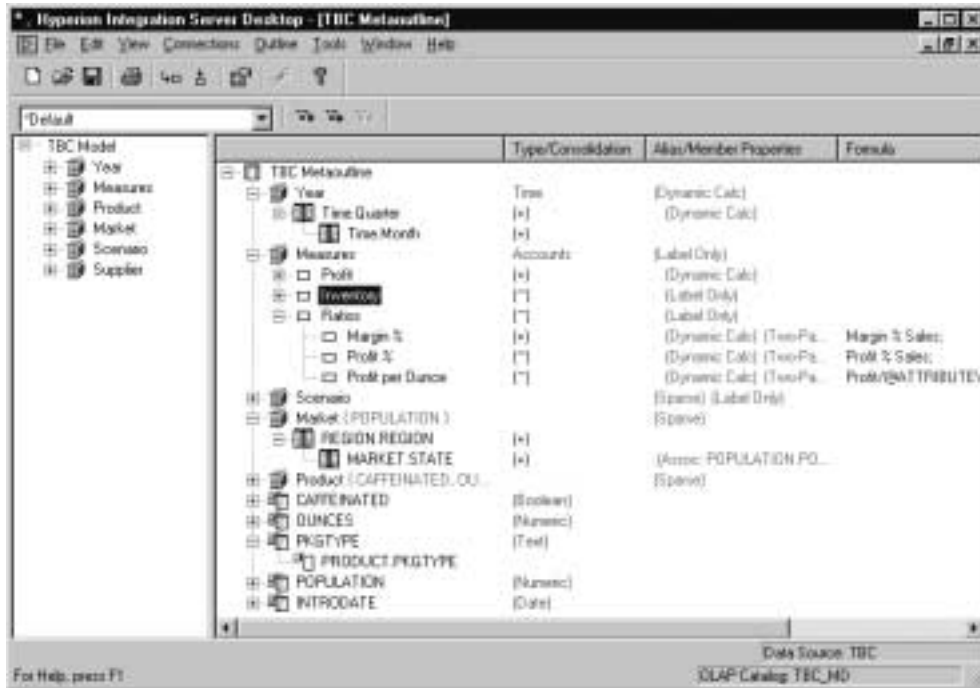


Figure 6-13: Viewing Hyperion Essbase Properties in Column Format

When the Display as Columns command is deselected, the columnar format view is replaced by the condensed format view, as shown in Figure 6-13. To return to the columnar format view, select Display as Columns.

- To display Hyperion Essbase properties in condensed format:
 1. Select a dimension, member level, or outline; for example TBC Metaoutline.
 2. Select View > Display Essbase Properties.
 3. If Display as Columns is selected, deselect it.

4. To determine which properties to display, from the submenu, select any of the following items:
 - **Consolidation Objects:** Displays consolidation properties, such as +, -, and %.
 - **Formula Objects:** Displays the assigned formula.
 - **Dimension Type:** Displays the dimension type: Time or Accounts.
 - **Alias:** Displays aliases.
 - **Member Properties:** Displays member properties, such as storage type, two-pass calculation, dynamic time, time balance and variance reporting properties, or UDAs.

The metaoutline is displayed in condensed format and the properties that you selected are displayed.

5. If you want to display all Hyperion Essbase properties, select **All**.

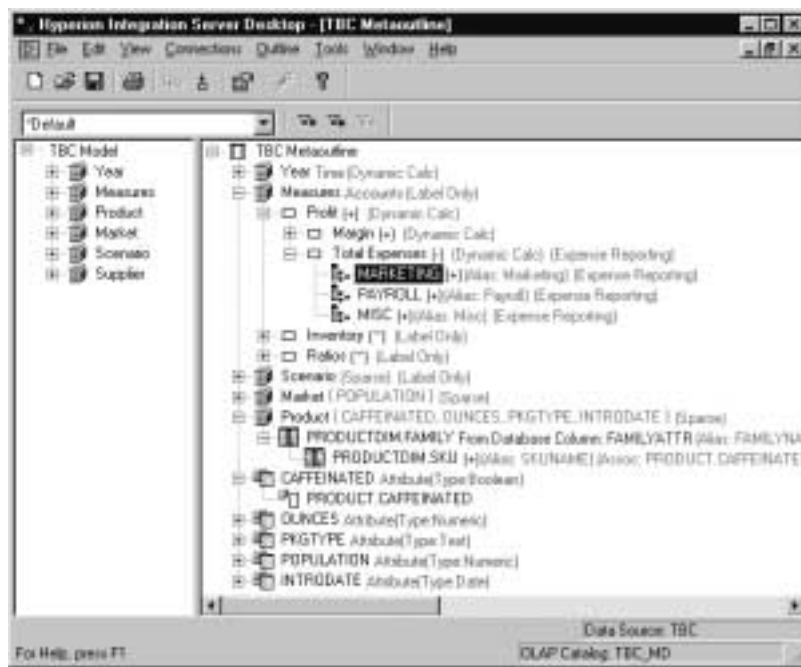


Figure 6-14: Viewing Hyperion Essbase Properties in Condensed Format

To return to the columnar format view, select Display as Columns.

Setting Hyperion Essbase Properties

Loading Dimensions and Members

This chapter describes how to use a metaoutline to add dimensions and members to a related Hyperion Essbase outline, a process also referred to as *member load*. First, the chapter describes everything that you need to do before starting a member load, such as connecting to the appropriate servers and setting query governors. Then, the chapter steps you through the member load process that you must follow to create or update the Hyperion Essbase outline.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 7-2
- “About Loading” on page 7-3
- “About Naming Applications and Databases” on page 7-4
- “Procedure 1: Preparing to Load Members” on page 7-4
- “Procedure 2: Starting a Member Load” on page 7-8
- “Procedure 3: Loading Members Immediately” on page 7-13
- “Procedure 4: Scheduling a Member Load” on page 7-17
- “Viewing Scheduled Member and Data Loads” on page 7-19
- “Deleting Scheduled Member and Data Loads” on page 7-22

If you are not sure how to view a Hyperion Essbase outline, see the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.

To load data into a Hyperion Essbase outline, see Chapter 8, “Loading Data.”

Understanding the Metaoutline Workflow

Figure 7-1 illustrates the workflow for creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following processes:

- Creating and working with a metaoutline
- Adding dimensions, member levels, and measures to a metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, member levels, and measures in a metaoutline
- Loading dimensions and members from the metaoutline into a Hyperion Essbase outline
- Loading the data specified in the metaoutline into a Hyperion Essbase database

This chapter focuses on loading dimensions and members from a metaoutline into a Hyperion Essbase outline.

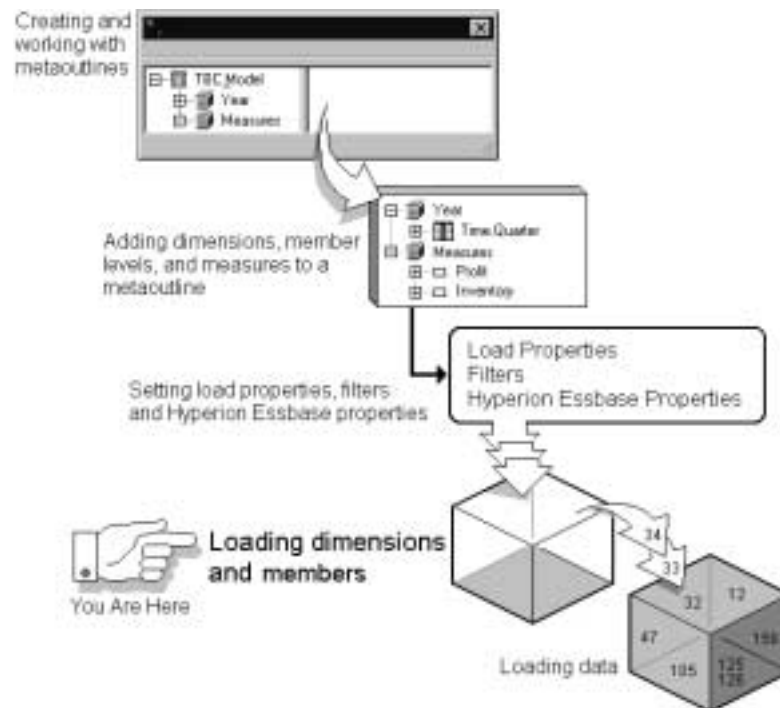


Figure 7-1: Metaoutline Workflow

About Loading

When you perform a load, OLAP Integration Server performs the following tasks:

- It retrieves information from OLAP Metadata Catalog and uses the information to generate SQL statements that specify how and what to retrieve from the external data source.
- It retrieves data from the external data source.
- It performs on the retrieved data the transformations specified in the OLAP model and the metaoutline.
- It loads the transformed data into the specified Hyperion Essbase application and database. If no Hyperion Essbase application and database exist for a member load, OLAP Integration Server creates an application and database.

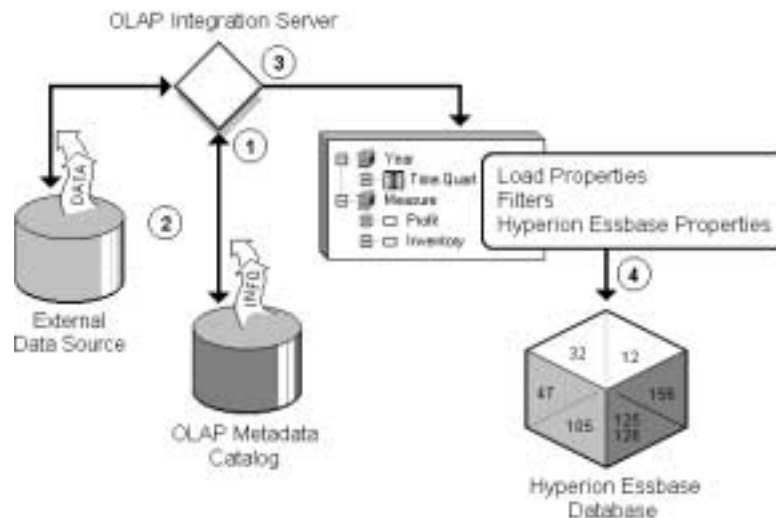


Figure 7-2: Loading Process

About Naming Applications and Databases

When naming applications and databases, follow these rules:

- Use no more than 8 characters.
- Do not use the following special characters anywhere in the name:

*	(asterisk)	<	(less than sign)
\	(backslash)	>	(greater than sign)
[]	(brackets)	.	(period)
:	(colon)	+	(plus sign)
,	(comma)	?	(question mark)
=	(equal sign)	"	(double quotation mark)
;	(semicolon)	'	(single quotation mark)
	(vertical bar)		

- Do not use spaces anywhere in the name.

Procedure 1: Preparing to Load Members

Before you start a member load, you must perform the following procedure:

1. Connect to the Hyperion Essbase server computer on which you want to create or change the Hyperion Essbase outline.

See the *Hyperion Integration Server Installation Guide* for information on starting Hyperion Integration Server Desktop.

2. If needed, set query governors to control the duration and the size of the queries that Hyperion Integration Server makes to the data source.

See “Using Query Governors” on page 7-5.

3. If needed, select a metaoutline filter to use for performing the load.


See “Viewing the Contents of a Metaoutline Filter” on page 5-12.

Using Query Governors

A *query governor* controls the duration and the size of the queries that OLAP Integration Server makes to the data source. A query governor prevents you from retrieving more data from the external data source than you want or need. If you set a query governor, OLAP Integration Server stops the query when it is taking too long or retrieving too much data.

For example, if you set the maximum retrievable rows to 200, OLAP Integration Server retrieves approximately 200 rows. If the query calls for more than 200 rows, OLAP Integration Server stops the query and displays an error message. Remember that viewing more than 10,000 rows is difficult for spreadsheet users.

CAUTION: Whether OLAP Integration Server loads the retrieved rows before stopping the query depends on the Isolation Level setting in the Hyperion Essbase database. For more information, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

- To set a query governor:
1. In the metaoutline, select the metaoutline name, and choose one of the following options to display the **OLAP Metaoutline Properties** dialog box:
 - Right-click the metaoutline name, and select Properties from the pop-up menu.
 - From the toolbar, click the **View Properties** button,  .

2. Select the **Governors** tab.



Figure 7-3: Setting a Query Governor

3. In the **Maximum time per query (in seconds)** text box, enter the maximum query time.
For example, to stop a query that takes longer than 100 seconds, type **100**.
4. In the **Maximum rows per query** text box, enter the maximum number of rows to retrieve.
For example, to stop a query that retrieves more than 200 rows, type **200**.

CAUTION: Whether or not Hyperion Integration Server loads the retrieved rows before stopping the query depends on the Isolation Level setting in the Hyperion Essbase database. For more information, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

5. If you are setting a query governor to view a sample outline or dimension, perform one of the following tasks:
 - Select **Stop at first level 0 member for each dimension** to see the structure of a Hyperion Essbase outline or the structure of a dimension without including all members. A level 0 member is a leaf level member; that is, a member with no hierarchy of members below it.
 - Select **Build entire branch** to view a dimension as it is displayed in the Hyperion Essbase outline. If the number that you enter for the duration or the size of the query is smaller than the entire branch, OLAP Integration Server stops when it reaches the duration or size limit.

OLAP Integration Server truncates sample outlines if they contain more than 32,767 members.

Note: If there is a shared member in the dimension, set the query governor to **Build entire branch**.
6. Click **OK**.

Procedure 2: Starting a Member Load

When you have performed the appropriate preparation steps, you are ready to start the member load.

➤ To load members:

1. Make sure that you perform the tasks described in “Procedure 1: Preparing to Load Members” on page 7-4. From the menu bar, select Outline > Member Load.

The **Essbase Application and Database** dialog box is displayed.

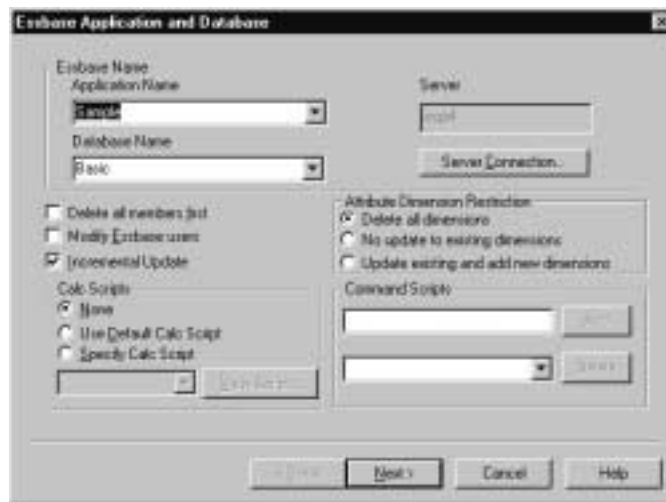


Figure 7-4: Choosing the Hyperion Essbase Application and Database

2. Complete one of the following actions:

- If you are creating a new Hyperion Essbase application and database, in the **Application Name** and **Database Name** text boxes, enter the names of the application and database.

See “About Naming Applications and Databases” on page 7-4.

Hyperion Essbase application and database names must be unique and must consist of no more than eight characters. It is recommended that database and application names start with a capital letter.

- If you are using an existing Hyperion Essbase application or database, select the names that you want from the appropriate lists.

3. If you want to delete all dimensions and members in an existing Hyperion Essbase outline, select **Delete all members first**.

When you delete all members, OLAP Integration Server removes all members from the existing Hyperion Essbase outline and then uses the member levels of the metaoutline to recreate the database. Because deleting all members is slower than creating or updating a Hyperion Essbase outline without deleting all members, delete all members first only if you have a specific reason to do so. You should delete all members if, for example, you know that some members have been removed from the OLAP model and you want to build a smaller Hyperion Essbase outline that contains the smaller set of members.

4. If you want to determine which Hyperion Essbase users can access the Hyperion Essbase application and database into which you are loading members, complete the following steps:

- a. Select **Modify Essbase users**.

When you select **Modify Essbase users**, the **Essbase Users** dialog box is displayed. The list boxes show all available and selected Hyperion Essbase users:





Figure 7-5: Selecting Hyperion Essbase Users

OLAP Integration Server retrieves the list of available users from the locations described in Table 7-1:

Table 7-1: Retrieval Locations of Specified Hyperion Integration Server Users

Type of Application or Database	Retrieval Location
New application	The Hyperion Essbase server. All of the users can create new applications and databases on the Hyperion Essbase server; for example, <code>sys</code> .
Existing application	The Hyperion Essbase application.
New database	The Hyperion Essbase application.
Existing database	The Hyperion Essbase database.

- b. In the **Available** list box, select a user to whom you want to give access.
- c. Click the  button to move the selected user to the **Selected** list box.
- d. When the names of all users who should access the Hyperion Essbase application and database are in the **Selected** list box, click **Next**.

Tip: If you want to give access to all available users, you need only to click the  button.

- 5. If you want to select specific dimensions not to update, select **Incremental Update**.

By default, OLAP Integration Server updates all dimensions and members. Updating an existing Hyperion Essbase outline incrementally is faster than updating all dimensions and members.

- 6. If you have selected **Incremental Update**, and the metaoutline contains attribute dimensions, select one of the options under **Attribute Dimension Restriction**:
 - a. **Delete all dimensions** deletes all attribute dimensions after the update.
 - b. **No update to existing dimensions** does not change attribute dimensions while updating other areas.
 - c. **Update existing and add new dimensions** updates all changes to attributes.

7. Click **Next**.

The **Incremental Update** dialog box is displayed:



Figure 7-6: Updating Hyperion Essbase Outlines Incrementally

- a. To prevent OLAP Integration Server from updating a dimension or member, select the dimension or member and click **Don't Update**.
A red **X** appears on the dimension that will not be updated.
 - b. When you have marked all dimensions and members that OLAP Integration Server should not update, click **Next**.
8. Proceed to "Procedure 3: Loading Members Immediately" on page 7-13 or to "Procedure 4: Scheduling a Member Load" on page 7-17.



You can use the `LOADMEMBER` command in OLAP Command Interface to load members. For more information about the `LOADMEMBER` command, see the *Hyperion Integration Server System Administrator's Guide*.

9. Select the calculation script to use to calculate the Hyperion Essbase database from the choices described in Table 7-2.

Table 7-2: Selecting a Calculation Script

Selection	Result
None	OLAP Integration Server does not run a calculation script on the Hyperion Essbase database.
Default	OLAP Integration Server runs the default calculation script that is specified for the Hyperion Essbase database. If you have not specified a calculation script to use as the default, Hyperion Essbase uses standard calculation order to calculate the database.
Any other calculation script	OLAP Integration Server runs the selected calculation script on the Hyperion Essbase database.

The calculation script that you choose is run after OLAP Integration Server loads data into the Hyperion Essbase database. The calculation script determines how Hyperion Essbase calculates the data values in the database.

For more information on calculation and on calculation scripts, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Procedure 3: Loading Members Immediately

You can schedule a member load to start immediately or to start at a specific date and time. If you choose to start a member load immediately, the Member Load dialog box is displayed. The dialog box provides you with dynamic status information about the member load and specific messages for any errors or warnings that may occur. If you encounter an error during the load, you can stop the member load. Hyperion Integration Server executes member loads on the computer that is running OLAP Integration Server.

Loading Dimensions and Members

► To load members immediately:

1. Make sure that you perform the tasks described in “Procedure 2: Starting a Member Load” on page 7-8.

After you complete the appropriate dialog boxes, as directed in “Procedure 2: Starting a Member Load” on page 7-8, and click **Next**, the **Schedule Essbase Load** dialog box is displayed:



Figure 7-7: Loading Members Immediately

2. If you want to save the script to a different name or change the name of the script, click **Save Scripts**.

The **Save Script As** dialog box is displayed:



Figure 7-8: Saving a Script

- a. Enter the new script name.
- b. Click **OK** to return to the **Schedule Essbase Load** dialog box.

3. Select **Now**.
4. Click **Finish**.

The **Member Load** dialog box is displayed:



Figure 7-9: Displaying Member Load Information

5. If you encounter an error and want to stop the member load, click **Stop Load**. (The **Stop Load** button is displayed during the load.)

To start the member load again, return to Step 1 of this procedure.

If any dimension or member is not loaded into the Hyperion Essbase outline, a reject text file for the rejected dimension or member is automatically created. The name of a reject text file consists of the name of the relevant dimension followed by the leaf node number of the dimension or member that did not load.

Loading Dimensions and Members

For example, Figure 7-10 illustrates that if the three member levels of the Product dimension do not load, two .txt files are created: PRODUCTDIM_0.txt (PRODUCTDIM.SKU), and PRODUCTDIM_1.txt (PRODUCTDIM.SKUNAME).

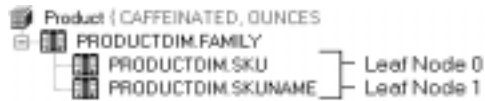


Figure 7-10: Leaf Member Example

For each reject file, a corresponding rules file is created. The rules files created for the rejected member levels of Figure 7-10 are PRODUCTDIM_0.rul (PRODUCTDIM.SKU), and PRODUCTDIM_1.rul (PRODUCTDIM.SKUNAME). Both .txt and .rul files are located in the ISHOME\loadinfo default directory. To place the files in a directory other than the default directory, set the ISLOADINFO environment variable to point to a different directory.

For more information on rules files, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

If a member does not load successfully, see “Problems with Loading Members or Data” on page 10-6.

For information on starting an unscheduled data load, see “Procedure 3: Loading Data Immediately” on page 8-8.

Procedure 4: Scheduling a Member Load

You can schedule a member load to start at a specific date and time. You can also schedule a load to be run regularly; for example, once a week. Hyperion Integration Server executes member loads on the computer that is running OLAP Integration Server.

➤ To schedule a member load:

1. Make sure that you perform the tasks described in “Procedure 2: Starting a Member Load” on page 7-8.

After you complete the appropriate dialog boxes as directed in “Procedure 2: Starting a Member Load” on page 7-8, the **Schedule Essbase Load** dialog box is displayed:



Figure 7-11: Scheduling the Member Load

2. Select **Schedule**.
3. If Hyperion Integration Server is on a computer running UNIX, proceed to step 5.
4. If Hyperion Integration Server is on a computer running Windows NT, start the Schedule service as described in “Starting the Windows NT Schedule Service Automatically” on page 7-18 and then proceed to step 5.

5. Enter the time to run the member load:
 - a. In the **Date** text box, enter the date. Use the mm/dd/yyyy format; for example, 01/17/2000.
 - b. In the **Time** text box, enter the time. Use the hh:mm:ss format; for example, 23:59:00 is 11:59 p.m. To schedule a load to run at midnight, enter 00:00:00, as midnight is considered the first hour of a day, not the last hour of a day.

Note that Hyperion Integration Server uses a 24-hour clock.

- c. From the **Frequency** drop-down list, select how frequently to perform the member load. You can schedule a member load to be performed one time, every day, every week, or every month.

You cannot schedule a member load more than once on a single day. To perform a member load more than once a day, you must implement each load individually. See “Procedure 3: Loading Members Immediately” on page 7-13.

6. Click **Finish**.

OLAP Integration Server loads the members into the Hyperion Essbase outline as scheduled.

Note: If a member does not load successfully, see “Problems with Loading Members or Data” on page 10-6.


To view a scheduled member load, see “Viewing Scheduled Member and Data Loads” on page 7-19. To delete a scheduled member load, see “Deleting Scheduled Member and Data Loads” on page 7-22.

Starting the Windows NT Schedule Service Automatically

OLAP Integration Server schedules loads by using the Windows NT Schedule service. It is recommended that you set the Schedule service to start automatically. See the Windows NT documentation. For more information, refer to the *Hyperion Integration Server Installation Guide*.

Starting the Windows NT Schedule Service Manually

You may, on occasion, need to start the Windows NT Schedule service manually.

- To start the Windows NT Schedule service manually:
 1. From the Windows NT taskbar, select Settings > Control Panels.
 2. Double-click the **Services** button, , to open the **Services** dialog box.

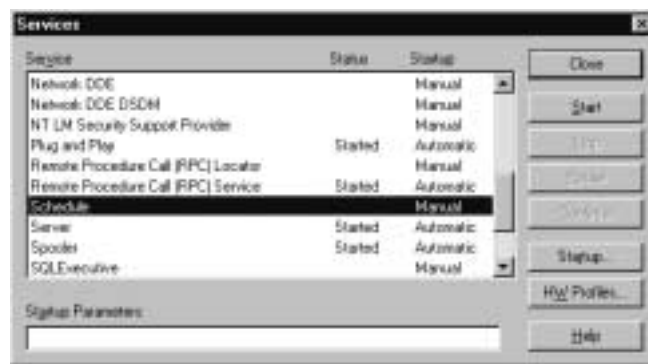


Figure 7-12: Starting the Windows NT Schedule Service

3. Scroll down to the **Schedule** service and select it.
4. Click **Start**.
5. Click **Close**.

Viewing Scheduled Member and Data Loads

You can view member and data loads after scheduling them. For information on performing data loads, see Chapter 8, “Loading Data.”

Before you can view scheduled loads, start Hyperion Integration Server Desktop and OLAP Server. See the *Hyperion Integration Server Installation Guide* for information on starting Hyperion Integration Server Desktop.

If Hyperion Integration Server is on a computer running Windows NT, you must start the Schedule service as described in “Starting the Windows NT Schedule Service Automatically” on page 7-18.

- To view all scheduled member and data loads:
 1. From the OLAP Metaoutline main window menu bar, select Tools > Scheduler.

The **Scheduled Jobs** dialog box is displayed.



Figure 7-13: Viewing a List of Scheduled Loads

The dialog box lists all member and data loads that are currently scheduled. The dialog box contains the following columns:

- **Job Id** displays the identification number for the load.
- **Essbase Target** displays the name of the Hyperion Essbase application and database into which OLAP Integration Server loads members or data.
- **M** stands for member load. A Y (yes) in the **M** column identifies a member load. **D** stands for data load. A Y (yes) in the **D** column identifies a data load. A Y (yes) in each column indicates a combined member and data load.
- **Next Run** displays the date of the next scheduled load.
- **Frequency** displays how frequently OLAP Integration Server performs the load.

Note: To update the list of scheduled loads, click **Refresh**.

2. To view more information about a particular load, in the **Job Id** column, select the job number of the load and then click **View**.

OLAP Integration Server opens the **View existing job** window. The window lists all properties for the selected load.



Figure 7-14: Viewing a Description of the Selected Load

3. When you finish viewing the load properties, click **OK**.
4. In the **Scheduled Jobs** dialog box, click **Close**.

Deleting Scheduled Member and Data Loads

You can delete member and data loads after you schedule them. For information on performing data loads, see Chapter 8, “Loading Data.”

Before you can delete scheduled loads, you must start Hyperion Integration Server. See the *Hyperion Integration Server Installation Guide* for information on starting Hyperion Integration Server Desktop.

If Hyperion Integration Server is on a computer running Windows NT, you must start the Schedule service as described in “Starting the Windows NT Schedule Service Automatically” on page 7-18.

► To delete a scheduled member or data load:

1. From the OLAP Metaoutline main window menu bar, select Tools > Scheduler.

OLAP Integration Server opens the **Scheduled Jobs** dialog box. The dialog box lists all member and data loads that are currently scheduled:



Figure 7-15: Viewing a List of Scheduled Loads

Note: To update the list of scheduled loads, click **Refresh**.

2. In the **Job Id** column, select a job number.
3. Click **Delete**.

OLAP Integration Server prompts you to confirm your direction to delete the scheduled load.

4. Click **Yes**.

Loading data into a Hyperion Essbase database is a process also referred to as *data load*. This chapter describes how to prepare for a data load, how to select a calculation script to execute after the data load finishes, and how to schedule the data load to start immediately or at a specific time or date.

This chapter contains the following topics:

- “Understanding the Metaoutline Workflow” on page 8-1
- “Procedure 1: Preparing to Load Data” on page 8-3
- “Procedure 2: Starting a Data Load” on page 8-4
- “Procedure 3: Loading Data Immediately” on page 8-8
- “Procedure 4: Scheduling a Data Load” on page 8-12

You can view the new and updated data in Hyperion Essbase Spreadsheet Add-in to verify that the data load was successful. If you need instruction, see the *Hyperion Essbase Spreadsheet Add-in User’s Guide*.

Understanding the Metaoutline Workflow

Figure 8-1 illustrates the workflow for creating a metaoutline and using the metaoutline to create and load members and data into a Hyperion Essbase database. This workflow includes the following processes:

- Creating and managing a metaoutline
- Adding dimensions, member levels, and measures to the metaoutline
- Setting load properties, filters, and Hyperion Essbase properties on dimensions, members, and measures in the metaoutline

Loading Data

- Loading dimensions and members from the metaoutline into the related Hyperion Essbase outline
- Loading the data specified in the metaoutline into the related Hyperion Essbase database

This chapter focuses on loading data into a Hyperion Essbase database from a metaoutline.

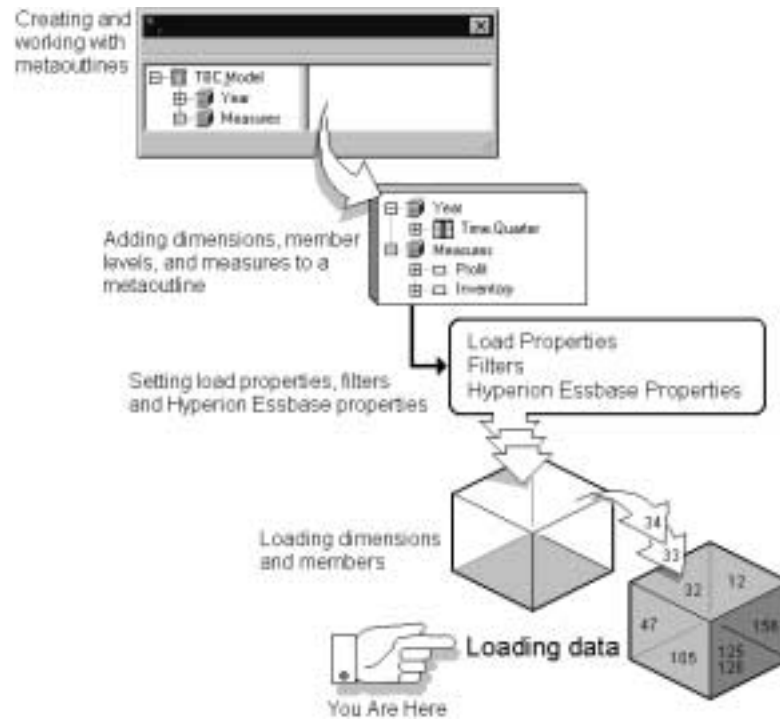


Figure 8-1: Metaoutline Workflow

Procedure 1: Preparing to Load Data

Before you start a data load, you must perform the following tasks:

1. Open the metaoutline.
See the *Hyperion Integration Server Installation Guide* for information on starting Hyperion Integration Server Desktop.
2. Connect to the Hyperion Essbase server computer on which you want to load the data.
See the *Hyperion Integration Server Installation Guide* for information on connecting to Hyperion Essbase servers.
3. If you want to control the duration and the size of the queries that OLAP Integration Server makes to the data source, set query governors.
See “Using Query Governors” on page 7-5.
4. If you want to filter data, select a metaoutline filter to use for performing the load.
See “Viewing the Contents of a Metaoutline Filter” on page 5-12.
5. If the Hyperion Essbase outline does not already exist, perform a member load.
See Chapter 7, “Loading Dimensions and Members.”

Note: If the data in the fact table aggregates in more than one way, such as sales (which are additive) and percentage discount values (which are not additive), you must load the different types of data separately into the Hyperion Essbase outline. To load the different types of data separately, create two metaoutlines from one OLAP model and load the data of each metaoutline separately into the Hyperion Essbase outline.

Now you are ready to start the data load. For more information about the steps that Hyperion Integration Server follows when performing a load, see “About Loading” on page 7-3.

Procedure 2: Starting a Data Load

- To start a data load:
 1. Make sure that you perform the tasks described in “Procedure 1: Preparing to Load Data” on page 8-3.
 2. From the menu bar, select Outline > Data Load.

The **Essbase Application and Database** dialog box is displayed.

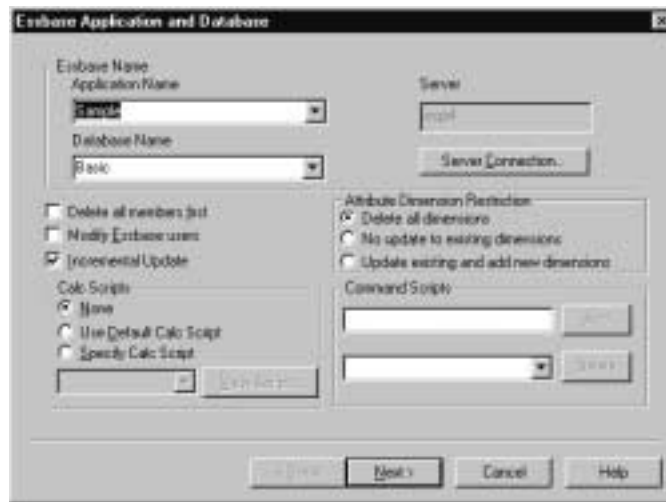


Figure 8-2: Choosing the Hyperion Essbase Application and Database

3. From the list of applications, select the Hyperion Essbase application that contains the database into which you want to load data.
4. From the list of Hyperion Essbase databases, select the Hyperion Essbase database into which you want to load data.

5. Select **Modify Essbase users** to determine which Hyperion Essbase users can access the Hyperion Essbase application and database into which you are loading data.

The **Essbase Users** dialog box is displayed:





Figure 8-3: Selecting Hyperion Essbase Users

OLAP Integration Server retrieves the list of available users from the locations described in Table 8-1.

Table 8-1: Retrieval Locations of Specified Hyperion Integration Server Users

Type of Application or Database	Retrieval Location
New application	The Hyperion Essbase server. All users can create new applications and databases on the Hyperion Essbase server; for example, <i>sys</i> .
Existing application	The Hyperion Essbase application.
New database	The Hyperion Essbase application.
Existing database	The Hyperion Essbase database.

6. To allow a user to access the Hyperion Essbase application and database, complete the following steps:
 - a. In the **Available** list box, select the user.
 - b. Click the  button to move the user to the **Selected** list box. To move all users, click the  button.
 - c. When the names of all users who should access the Hyperion Essbase application and database are in the **Selected** list box, click **Next**.
7. If you want to select specific members not to update, in the **Essbase Application and Database** dialog box, select **Incremental Update**.

By default, OLAP Integration Server updates all dimensions and members. Updating an existing Hyperion Essbase outline incrementally is faster than updating all dimensions and members.
8. If you have selected **Incremental Update**, and the metaoutline contains attribute dimensions, select one of the options under **Attribute Dimension Restriction**:
 - a. **Delete all dimensions** deletes all attribute dimensions after the update.
 - b. **No update to existing dimensions** does not change attribute dimensions while updating other areas.
 - c. **Update existing and add new dimensions** updates all changes to attributes.

9. Click **Next**.

The **Incremental Update** dialog box is displayed:



Figure 8-4: Updating a Hyperion Essbase Outline Incrementally

- a. To prevent OLAP Integration Server from updating a member, select the member and click **Don't Update**.

When a member is selected for **Don't Update**, the **Don't Update** button toggles to an **Update** button, as shown in Figure 8-4. With this configuration of the button, you can change the status of the member back to **Update**, if you choose.

Note: You must update at least one member from each dimension when performing a data load. If you choose a dimension and click **Don't Update**, you will get an error message. If more than one hierarchy exists in the Hyperion Essbase outline, you can perform an incremental update.

- b. After you mark all members that OLAP Integration Server should not update, click **Next**.



You can use the `LOADDATA` command in OLAP Command Interface to load data. For more information about this command, see the *Hyperion Integration Server System Administrator's Guide*.

10. Select the calculation script to use to calculate the Hyperion Essbase database from the choices described in Table 8-2.

Table 8-2: Selecting a Calculation Script

Selection	Result
None	OLAP Integration Server does not run a calculation script on the Hyperion Essbase database.
Default	OLAP Integration Server runs the default calculation script that is specified for the Hyperion Essbase database. If you have not specified a calculation script to use as the default, Hyperion Essbase uses standard calculation order to calculate the database.
Any other calculation script	OLAP Integration Server runs the selected calculation script on the Hyperion Essbase database.

The calculation script that you choose is run after OLAP Integration Server loads data into the Hyperion Essbase database. The calculation script determines how Hyperion Essbase calculates the data values in the database.

For more information on calculation and on calculation scripts, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

11. Click **Next** and proceed to “Procedure 3: Loading Data Immediately” on page 8-8.

Procedure 3: Loading Data Immediately

You can schedule a data load to start immediately or to start at a specific date and time. If you choose to start a data load immediately, the Data Load dialog box is displayed. The dialog box provides you with dynamic status information about the data load and specific messages for any errors or warnings that may occur. If you encounter an error during the load, you can stop the data load. Hyperion Integration Server executes data loads on the computer that is running OLAP Integration Server.

- To load data immediately:
 1. Make sure that you perform the tasks described in “Procedure 2: Starting a Data Load” on page 8-4.
 After you complete the appropriate dialog boxes (as directed in “Procedure 2: Starting a Data Load” on page 8-4) and click **Next**, the **Schedule Essbase Load** dialog box is displayed.



Figure 8-5: Loading Data Immediately

2. If you want to save the script to a different name or change the name of the script, click **Save Scripts**.

The **Save Script As** dialog box is displayed:



Figure 8-6: Saving a Script

- a. Enter the new script name.
- b. Click **OK** to return to the **Schedule Essbase Load** dialog box.

Loading Data

3. In the **Schedule Essbase Load** dialog box, select **Now**.
4. Click **Finish**.

The **Data Load** dialog box is displayed:

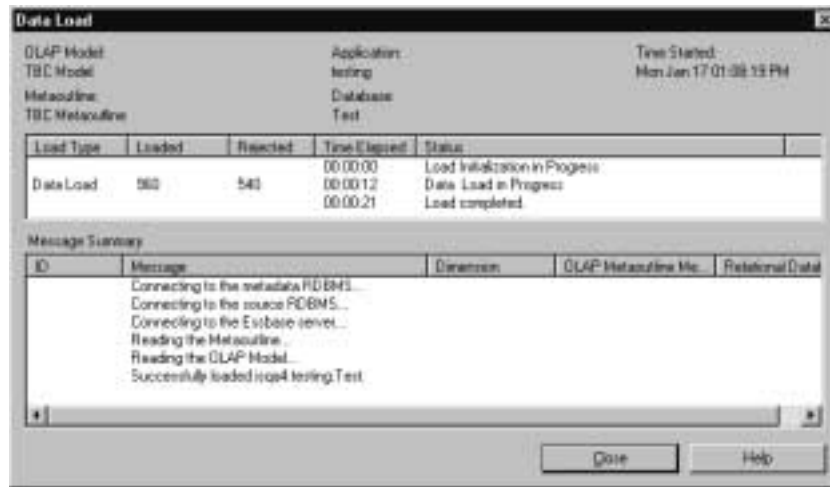


Figure 8-7: Displaying Data Load Information

5. If you encounter an error and want to stop the data load, click **Stop Load**. (The **Stop Load** button is displayed during the load.)

If you want to restart the data load, return to step 1 of this procedure.

If you load only data into the Hyperion Essbase outline and the load fails, rejected dimensions and members are listed in a `dataload.txt` file.

If you load both members and data into the Hyperion Essbase outline and the load fails, a reject text file for each rejected dimension and member is created. The name of each file consists of the name of the relevant dimension followed by the leaf node number of the dimension or member that did not load.

For example, Figure 8-8 illustrates that if the two member levels of the Product dimension do not load, two .txt files are created: PRODUCTDIM_0.txt (PRODUCTDIM.SKU), and PRODUCTDIM_1.txt (PRODUCTDIM.SKUNAME).



Figure 8-8: Leaf Node Example

For each rejected file, a corresponding rules file is created. The rules files created for the rejected member levels in Figure 8-8 are PRODUCTDIM_0.rul (PRODUCTDIM.SKU), and PRODUCTDIM_1.rul (PRODUCTDIM.SKUNAME).

Both .txt and .rul files are located in the ISHOME\loadinfo default directory. To place the files in a directory other than the default directory, set the ISLOADINFO environment variable to point to a different directory.

- If the data does not load successfully, see “Problems with Loading Members or Data” on page 10-6.
- For information on starting an unscheduled member load, see “Procedure 3: Loading Members Immediately” on page 7-13.

Procedure 4: Scheduling a Data Load

You can schedule a data load to start immediately or to start at a specific date and time. Hyperion Integration Server executes scheduled data loads on the computer that is running OLAP Integration Server.

- To schedule a data load to run at a specific time and date:
 1. Make sure that you perform the tasks described in “Procedure 2: Starting a Data Load” on page 8-4.

The **Schedule Essbase Load** dialog box is displayed:



Figure 8-9: Scheduling a Data Load

2. Select **Schedule**.
3. If Hyperion Integration Server is on a computer running UNIX, proceed to Step 5.
4. If Hyperion Integration Server is on a computer running Windows NT, start the Schedule service as described in “Starting the Windows NT Schedule Service Automatically” on page 7-18.

5. Enter the time to run the data load:
 - a. In the **Date** text box, enter the date. Use the mm/dd/yyyy format; for example, 01/17/2000.
 - b. In the **Time** text box, enter the time. Use the hh:mm:ss format; for example, 23:59:00 is 11:59 p.m. To schedule a load to run at midnight, enter 00:00:00, as midnight is considered the first hour of a day, not the last hour of a day. Note that Hyperion Integration Server uses a 24-hour clock.
 - c. From the **Frequency** drop-down list box, select how frequently to perform the data load. You can schedule data loads to be performed one time, every day, every week, or every month.

To perform a data load more than once a day, you must implement each load individually.

6. Click **Finish**.

OLAP Integration Server loads the data into the Hyperion Essbase database as scheduled.

Note: If the data does not load successfully, see “Failure of a Member Load or Data Load” on page 10-4.

To delete a scheduled data load, see “Deleting Scheduled Member and Data Loads” on page 7-22. To view a scheduled data load, see “Viewing Scheduled Member and Data Loads” on page 7-19.

Loading Data

Creating Drill-Through Reports

This chapter describes how to create Drill-Through reports that users can access using Hyperion Essbase Spreadsheet Add-in. This chapter contains the following topics:

- “About Drill-Through” on page 9-1
- “Adding a Report” on page 9-2
- “Specifying the Contents of a Report” on page 9-5
- “Specifying the Format of a Report” on page 9-16
- “Specifying Spreadsheet User Actions” on page 9-21
- “Working with Reports” on page 9-26

About Drill-Through

When you create a Hyperion Essbase database, you do not use all of the data of the associated relational data source. Instead, you choose the data that spreadsheet users are most likely to want and summarize it in a way that is useful to them. For example, you might summarize the amount of root beer sold in each state. The summaries enable spreadsheet users to compare sales across states and regions. They can answer questions such as “Which state sold the most root beer in January?” and “Which state sold the most root beer in July?”

Creating Drill-Through Reports

Some spreadsheet users may need more detailed information to create a strategy to increase root beer sales. They need to answer the following types of questions:

- What are the ten lowest performing stores in California?
- What are the addresses of the stores?
- What are the names of the store managers?

This information is not stored in the Hyperion Essbase database, but it is stored in the relational data source.

To view the relevant data, spreadsheet users must *drill through* to the relational data source and view the relational data in the context of the data in the Hyperion Essbase database.

Hyperion Integration Server enables you to define Drill-Through reports. A Drill-Through report is based on an *intersection level* (member combination) that spreadsheet users can double-click to start the drill-through process. Then spreadsheet users view or customize pre-defined Drill-Through reports that retrieve the relevant detail columns from the relational data source. Because OLAP Integration Server captures the metadata necessary to create the Hyperion Essbase outline, it returns the Drill-Through report in the context of the data that spreadsheet users are viewing.

Data for attribute dimensions and members appear in the Hyperion Essbase Spreadsheet Add-in, but attribute data does not appear in Drill-Through reports.

For more information on the Hyperion Essbase Spreadsheet Add-in and Drill-Through reports, see the *Hyperion Essbase Spreadsheet Add-in User's Guide* and the Hyperion Essbase Spreadsheet Add-in Online Help.

Adding a Report

You can create a new Drill-Through report and then add intersection levels and columns, or you can base a Drill-Through report on an existing report and vary specific intersection levels and columns. You can create multiple reports for a single intersection, and spreadsheet users can select which report to view.

This topic contains the following subtopics that describe Drill-Through report creation:

- “Creating a New Report” on page 9-3
- “Copying an Existing Report” on page 9-5

For more information on intersection levels and columns, see “About Intersection Levels” on page 9-6.

Creating a New Report

The Drill-Through report that you create is what spreadsheet users see when they drill from a cell in Hyperion Essbase Spreadsheet Add-in through to the relational data source.

- To create a new report:
 1. From the menu toolbar, select Edit > Drill-Through Reports to display the **Drill-Through Reports** dialog box. For example, the dialog box shown in Figure 9-1 lists the three sample reports included in the sample application.



Figure 9-1: Drill-Through Reports Dialog Box

Creating Drill-Through Reports

2. Click **Add** to add a new report.

A dialog box opens, prompting you to confirm that you want to add a new report.

3. Click **Yes**.

A new report is displayed in the **Drill-Through Reports** dialog box. By default, the new report is named Report4.



Figure 9-2: New Drill-Through Report

4. If you want to rename the report, see “Renaming a Report in the Drill-Through Reports Dialog Box” on page 9-26.
5. If you want to add intersection levels and columns to the report, see “Adding Intersection Levels” on page 9-11 and “Adding Drill-Through Columns” on page 9-13.

Copying an Existing Report

If you want to base a Drill-Through report on an existing report and vary a few intersection levels and columns, you can save time by copying the existing report.

- To copy an existing report:
 1. From the menu toolbar, select Edit > Drill-Through Reports.
 2. In the **Drill-Through Reports** dialog box, select the Drill-Through report that you want to copy.
 3. Click **Add**.

A dialog box opens, prompting you to confirm that you want to copy the report.
 4. Click **Yes**.
 5. If you want to rename the report, see “Renaming a Report in the Drill-Through Reports Dialog Box” on page 9-26.
 6. If you want to add or change intersection levels and columns in the report, see “Adding Intersection Levels” on page 9-11 and “Adding Drill-Through Columns” on page 9-13.

Specifying the Contents of a Report

You must define the contents of a Drill-Through report. The defining process includes adding intersection levels and columns and setting query governors to control the amount of data retrieved from the relational data source. This topic contains the following subtopics:

- “About Intersection Levels” on page 9-6
- “Following the Rules for Intersection Levels” on page 9-7
- “Determining Where to Set an Intersection Level” on page 9-7
- “Reviewing Sample Drill-Through Reports” on page 9-8
- “Adding Intersection Levels” on page 9-11
- “Adding Drill-Through Columns” on page 9-13
- “Setting the Size of a Query” on page 9-15

About Intersection Levels

A Drill-Through report is made up of intersection levels and columns. An *intersection level* defines the point at which users drill through; that is, the value that they double-click in Hyperion Essbase Spreadsheet Add-in to view the report. For example, in the Product Supplier report in the sample application, Time.Month and PRODUCTDIM.SKU are the intersection levels.



Figure 9-3: Drill-Through Levels in the Product Supplier Report

When a spreadsheet user double-clicks a data cell that corresponds to a Drill-Through intersection level, OLAP Integration Server retrieves from the relational data source the detail data that is specified in the default Drill-Through report. For example, to view a report defined on the intersection levels Time.Month and PRODUCTDIM.SKU, a spreadsheet user can select any of various intersections, such as the intersection of January, Profit, West, 100-20, or the intersection of February, Profit, California, 200-10.

An intersection level affects all members of the specified dimension at the specified level. For example, if you set an intersection level on REGION.REGION in the MARKET dimension, you are setting the intersection level at level 1. (Levels are numbered from the leaf node, level 0, toward the root.) All members of the MARKET dimension at level 1 and lower can access the Drill-Through report. Thus, both the members under REGION.REGION *and* the members under

COLORADO, level 0, can access the report. In a dimension that has more than one level 1 member, all level 1 members can access the report if any level 1 member is specified as the intersection level.

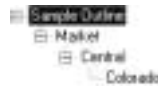


Figure 9-4: Intersection Levels

Following the Rules for Intersection Levels

Use the following rules when setting intersection levels:

- In a single Drill-Through report, you can add only one intersection level for each dimension. For example, in the Market dimension in TBC Metaoutline, you can define an intersection level for REGION or for STATE but not for both.
- You can select only one measure for each report. To create multiple reports, each with a different measure, create the first report, make a copy of the first report, and change the measure of the copied report.

To make a report applicable to all measures, select the dimension tagged as Accounts.

- Intersection levels are inherited. A child of a dimension or member inherits the Drill-Through level defined for its parent. Thus, to define a different intersection level for the child of the dimension or member, you must create a new report.

Determining Where to Set an Intersection Level

When you define an intersection level in a metaoutline, you determine the Hyperion Essbase cells from which a spreadsheet user can drill through.

- **Dimension:** If you set an intersection level for a metaoutline dimension, spreadsheet users can drill through from any intersection that contains the related Hyperion Essbase dimension. For example, if you set an intersection level at Product, spreadsheet users can drill through from any level in the Product dimension. Thus, spreadsheet users can double-click any data value that involves a member of the Product dimension, such as Colas or 100-10.

To make a report applicable to all measures, select the dimension tagged as Measures.

- **Member:** If you set an intersection level for a metaoutline member level, spreadsheet users can drill through from any intersection that contains a Hyperion Essbase member that is based on the designated metaoutline member level. For example, if you set an intersection level at PRODUCTDIM.SKU, spreadsheet users can double-click any data value that involves a member based on PRODUCTDIM.SKU, such as 100-10 or 100-20.
- **Neither a dimension nor a member:** If you do not set an intersection level for a dimension or for any member of the dimension, spreadsheet users can drill through from any member of the dimension (assuming that you have defined at Drill-Through report on at least one intersection level for the Hyperion Essbase database).

Reviewing Sample Drill-Through Reports

The sample application contains three Drill-Through reports that illustrate how the setting of various intersection levels affects the reports that spreadsheet users see.

The sample metaoutline contains the following Drill-Through reports:

- An Expenses by Product report with intersection levels of MONTH and STATE. To view this report, Hyperion Essbase Spreadsheet Add-in users drill down to any month under any quarter and any state under any region. The worksheet excerpt in Figure 9-5 shows a spreadsheet at the highest level from which spreadsheet users can view the Expenses by Product report.

At this level of the Hyperion Essbase Spreadsheet Add-in, users are prompted to view the Drill-Through report. Users can view the report only from the MONTH level in the Year dimension, and from the STATE level in the Market dimension, because those are the lowest levels in their respective dimensions. In the other dimensions, users can view the report from any level.



		Measures	Scenario	Product
Colorado	Jan	585		
Colorado	Feb	622		
Colorado	Mar	596		

Figure 9-5: Worksheet for Expenses by Product Report

The highest level of the Expenses by Product report is shown in Figure 9-6.

	A	B	C	D	E	F
1	STATE	SKUNAME	TRANSDATE	PAYROLL	MARKETING	MSC
2	Colorado	Caffeine Free Cola	2000-01-04	7.5	15	0
3	Colorado	Caffeine Free Cola	2000-01-09	2.5	5	0
4	Colorado	Caffeine Free Cola	2000-01-24	11	24	1
5	Colorado	Cola	2000-01-03	12	27	0
6	Colorado	Cola	2000-01-06	3.4	6.8	0
7	Colorado	Cola	2000-01-21	1.7	3.4	0
8	Colorado	Cola	2000-01-28	4.9	9.8	0
9	Colorado	Dark Cream	2000-01-01	11.02	21.66	0.38
10	Colorado	Dark Cream	2000-01-02	0.2	0.4	0
11	Colorado	Dark Cream	2000-01-04	1.6	3.2	0
12	Colorado	Dark Cream	2000-01-08	17.6	35.2	0
13	Colorado	Dark Cream	2000-01-18	0.6	1.2	0

Figure 9-6: Expenses by Product Drill-Through Report

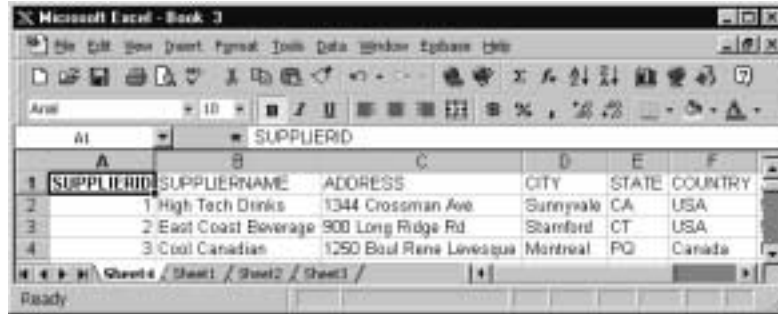
- A Product Supplier report on the intersection level of MONTH and PRODUCTDIM.SKU. The highest level of the spreadsheet from which the Product Supplier report can be viewed is shown in Figure 9-7. Users can view the report only from the MONTH level in the Year dimension, and from PRODUCTDIM.SKU in the Product dimension, because those are the lowest levels in their respective dimensions. In the other dimensions, users can view the report from any level.

		Measures	Scenario	Market
100-10	Jan	1710		
100-10	Feb	1666		
100-10	Mar	1720		

Figure 9-7: Worksheet for Product Supplier Report

Creating Drill-Through Reports

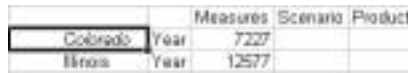
The highest level of the Product Supplier report is shown in Figure 9-8.



SUPPLIERID	SUPPLIERNAME	ADDRESS	CITY	STATE	COUNTRY
1	High Tech Drinks	1344 Crossman Ave	Sunnyvale	CA	USA
2	East Coast Beverage	900 Long Ridge Rd	Stamford	CT	USA
3	Cool Canadian	1250 Boul Rene Levesque	Montreal	QC	Canada

Figure 9-8: Product Supplier Drill-Through Report

- A Regional Director report with only one intersection level, MARKET.STATE. To view this report, spreadsheet users drill down to any state under any region. The worksheet excerpt in Figure 9-9 shows a spreadsheet at the highest level from which spreadsheet users can view the Regional Director report. Users can view the report from any level in the other dimensions, as long as they have drilled down to STATE in the MARKET dimension.



	Measures	Scenario	Product
Colorado	Year	7227	
Illinois	Year	12577	

Figure 9-9: Worksheet for Regional Director Report

The highest level of the Regional Director report is shown in Figure 9-10.



REGION DIRECTOR
Central
Cindy Traveller

Figure 9-10: Regional Director Drill-Down Report

Adding Intersection Levels

An *intersection level* defines the point at which users drill through; that is, the value that they double-click in Hyperion Essbase Spreadsheet Add-in to view a report. For more information, see “About Intersection Levels” on page 9-6, “Determining Where to Set an Intersection Level” on page 9-7, and “Reviewing Sample Drill-Through Reports” on page 9-8.

➤ To add an intersection level to a Drill-Through report:

1. Create a report.

See “Adding a Report” on page 9-2.

2. In the **Drill-Through Reports** dialog box, select **OLAP Intersection Levels**, and click **Add**.

The **Add New Intersection Levels** dialog box is displayed. The left list box contains all dimensions and member levels defined in the metaoutline. Each member level in the metaoutline represents a set of members at the same level in the Hyperion Essbase database. The right list box contains all intersection levels defined in the Drill-Through report.

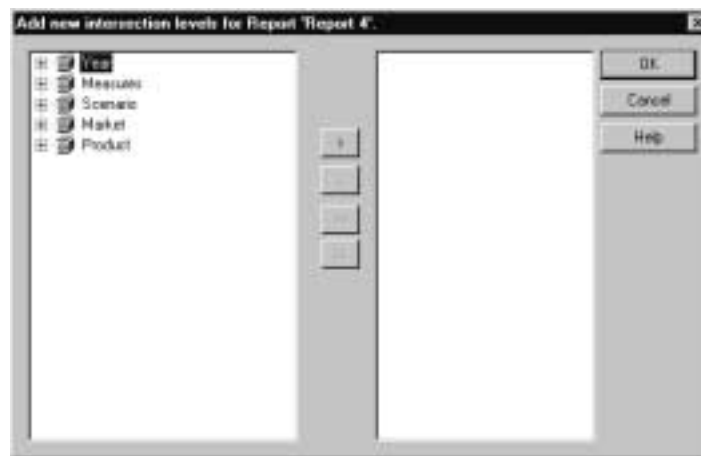


Figure 9-11: Adding Intersection Levels

Creating Drill-Through Reports

3. Click all signs to expand dimensions and see member levels.
4. Select the member level to add, and click to move the member level to the right list box.

To move all member levels to the right list box, click .

To remove a member level from the right list box, click .

To remove all member levels from the right list box, click .

For example, Figure 9-12 shows a set of intersection levels defined for REPORT 4.

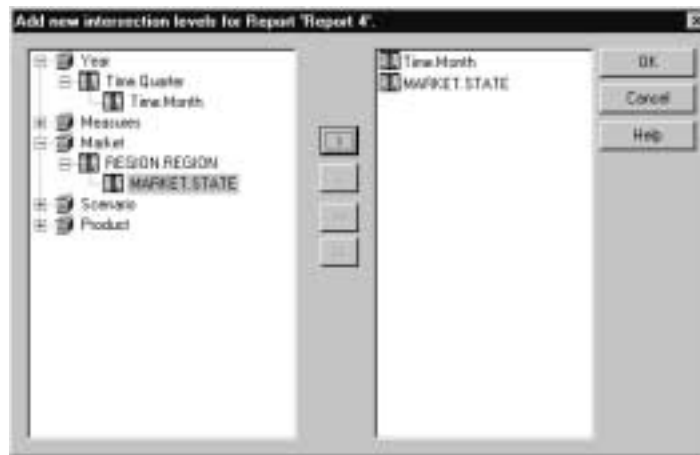


Figure 9-12: Intersection Levels

5. Click **OK** to return to the **Drill-Through Reports** dialog box.

Adding Drill-Through Columns

A *Drill-Through column* is the relational database column that OLAP Integration Server retrieves when a spreadsheet user double-clicks an intersection level. Columns contain detail information that is not available in the Hyperion Essbase database; for example, a list of store managers.

You can use in the Drill-Through report any column of the OLAP model that is not marked as Hidden, including columns marked as Drill-Through.

► To add columns to a Drill-Through report:

1. Create a report.

See “Adding a Report” on page 9-2.

2. Add the intersection levels.

See “Adding Intersection Levels” on page 9-11.

3. In the **Drill-Through Reports** dialog box, select **Drill-Through Report Columns**, and click **Add**.

The **Add New Report Columns** dialog box is displayed. The left list box contains all tables and columns that are not marked as Hidden in the OLAP model. The right list box contains all columns defined in the Drill-Through report.

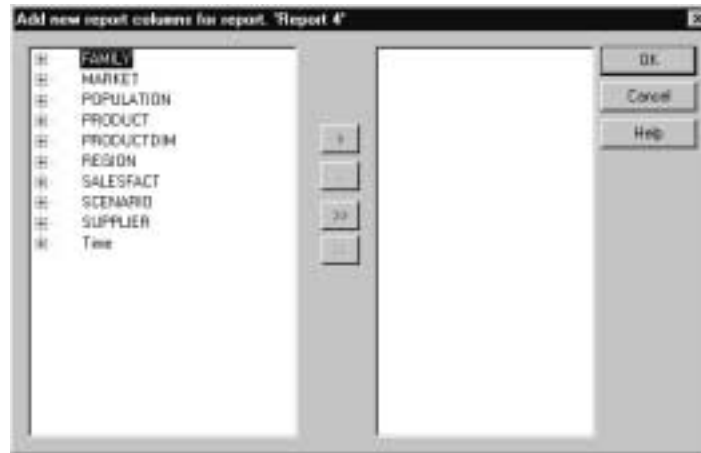







Figure 9-13: Adding Drill-Through Columns

Creating Drill-Through Reports

4. Click the  signs to expand the tables.
5. Select the column to add and click  to move the column to the right list box.

To add all columns to the right list box, click .

To remove a column from the right list box, click .

To remove all columns from the right list box, click .

For example, Figure 9-14 contains the set of columns for a TBC Drill-Through report. These columns provide spreadsheet users a view of details on population size, product, and package type.

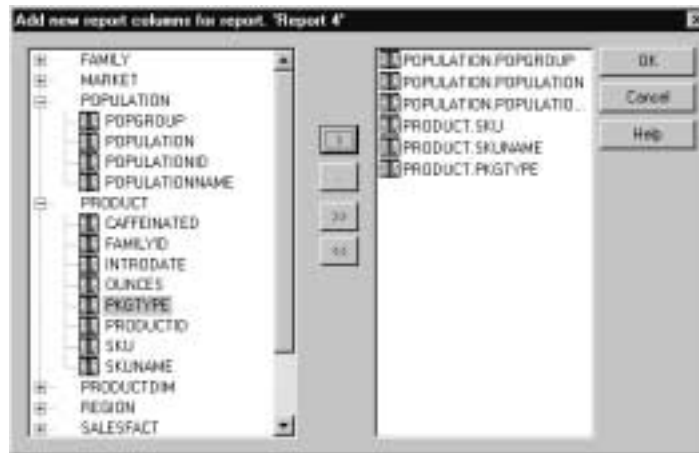


Figure 9-14: Sample Drill-Through Report Columns

6. Click **OK** to return to the **Drill-Through Reports** dialog box.

Setting the Size of a Query

A query governor controls the duration and the size of the queries that OLAP Integration Server makes to the data source to retrieve columns. A query governor prevents you from retrieving more data from the external data source than you want or need. For example, a column in the relational data source may contain 10,000 rows, and you may not want to retrieve all rows. More than 10,000 rows is difficult for spreadsheet users to view.

If you set a query governor, OLAP Integration Server stops the query if it is taking too long or retrieving too much data. For example, if you set 200 as the maximum number of rows to be retrieved, OLAP Integration Server retrieves approximately 200 rows. If the query tries to retrieve more than 200 rows, OLAP Integration Server stops the query and displays an error message.

Note: If a query stops because it reaches the time or row limit specified in the query governor, the spreadsheet user views incomplete data.

- To set a query governor:
 1. In the **Drill-Through Reports** dialog box, select a report name.
 2. Click **Edit** to open the **Report properties** dialog box.
 3. In the **Report properties** dialog box, select the **Governors** tab.

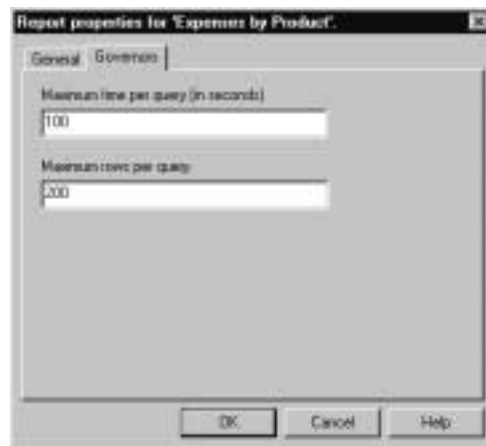


Figure 9-15: Setting a Query Governor

4. In the **Maximum time per query (in seconds)** text box, enter the maximum query time.
For example, to stop a query that takes longer than 100 seconds, type **100**.
5. In the **Maximum rows per query** text box, enter the maximum number of rows to retrieve.
For example, to stop a query that retrieves more than 200 rows, type **200**.
6. Click **OK**.

Specifying the Format of a Report

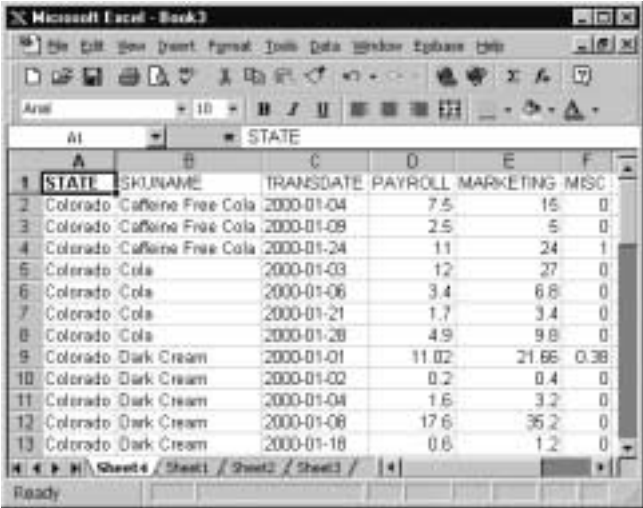
You can define the format of a report by specifying the order of columns and rows. This topic contains the following subtopics:

- “Specifying Column Ordering” on page 9-17
- “Specifying Row Ordering” on page 9-19

You can enable spreadsheet users to specify the order of columns and rows in a Drill-Through report. See “Customizing a Report” on page 9-22.

Specifying Column Ordering

Columns are displayed in a report from left to right in the order in which they are listed from top to bottom in the list box of the Drill-Through Reports dialog box. Arrange the columns in the list box in the order in which you want them to be displayed in the Drill-Through report; for example, in Figure 9-16, STATE is the first column, SKUNAME is the second, and TRANSDATE is the third.



The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F
1	STATE	SKUNAME	TRANSDATE	PAYROLL	MARKETING	MSC
2	Colorado	Caffeine Free Cola	2000-01-04	7.5	15	0
3	Colorado	Caffeine Free Cola	2000-01-09	2.5	5	0
4	Colorado	Caffeine Free Cola	2000-01-24	11	24	1
5	Colorado	Cola	2000-01-03	12	27	0
6	Colorado	Cola	2000-01-06	3.4	6.8	0
7	Colorado	Cola	2000-01-21	1.7	3.4	0
8	Colorado	Cola	2000-01-28	4.9	9.8	0
9	Colorado	Dark Cream	2000-01-01	11.02	21.66	0.38
10	Colorado	Dark Cream	2000-01-02	0.2	0.4	0
11	Colorado	Dark Cream	2000-01-04	1.6	3.2	0
12	Colorado	Dark Cream	2000-01-08	17.6	35.2	0
13	Colorado	Dark Cream	2000-01-18	0.6	1.2	0

Figure 9-16: Column Order in TBC Drill-Through Report

Creating Drill-Through Reports

- To move a column:
 1. From the menu bar, select Edit > Drill-Through Reports to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, click the sign adjacent to the Drill-Through report that you want to work with.
 3. From the list of the columns in the report, select a column; for example, SALESFACT.TRANSDATE.

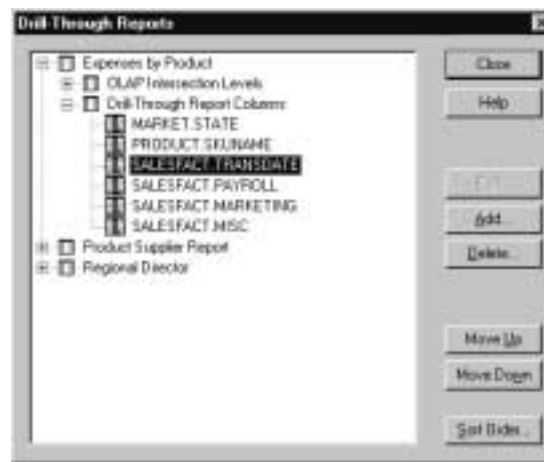


Figure 9-17: Positioning Columns

4. Click **Move Up** or **Move Down** to position the column.
5. When the columns are in the correct order, click **Close**.

Specifying Row Ordering

You can determine the order in which Hyperion Essbase Spreadsheet Add-in displays the rows it retrieves; for example, you can sort the contents of the PRODUCTDIM.SKUNAME column in descending order (from highest to lowest value). This sort presents the products in reverse alphabetical order, from Vanilla Cream to Old Fashioned to Caffeine Free Cola and produces the following report:

	A	B	C	D	E	F
1	STATE	SKUNAME	TRANSDATE	PAYROLL	MARKETING	MISC
2	Colorado	Vanilla Cream	2000-01-02	26.19	18.43	0.97
3	Colorado	Vanilla Cream	2000-01-09	20	10	0
4	Colorado	Vanilla Cream	2000-01-25	0.81	0.57	0.03
5	Colorado	Strawberry	2000-01-02	0.5	0.5	0
6	Colorado	Strawberry	2000-01-03	1.44	1.68	0
7	Colorado	Strawberry	2000-01-06	3.6	4.2	0
8	Colorado	Strawberry	2000-01-08	5.1	5.1	0
9	Colorado	Strawberry	2000-01-12	4.4	4.4	0
10	Colorado	Strawberry	2000-01-19	5.04	5.88	0
11	Colorado	Strawberry	2000-01-20	1.92	2.24	0
12	Colorado	Orange	2000-01-08	10.6	5.3	0
13	Colorado	Orange	2000-01-10	28.51	21.99	0
14	Colorado	Orange	2000-01-14	1.89	1.71	0
15	Colorado	Old Fashioned	2000-01-03	5.7	5.7	0
16	Colorado	Old Fashioned	2000-01-11	4.07	5.55	0
17	Colorado	Old Fashioned	2000-01-16	5.28	7.2	0
18	Colorado	Old Fashioned	2000-01-17	4.3	4.3	0
19	Colorado	Old Fashioned	2000-01-23	1.65	2.25	0

Figure 9-18: Row Sort Order in TBC Drill-Through Report

If you do not determine the sort order, spreadsheet users view data in the order determined by the relational data source. SQL Server, for example, sorts members in ascending order by default.

Creating Drill-Through Reports

- To determine the order of rows:
 1. From the menu bar, select Edit > Drill-Through Reports to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, select the Drill-Through report to sort.
 3. Click **Sort Order** to open the **Set Sort Order** dialog box.

The **Set Sort Order** dialog box displays in the left list box the columns included in the report. The right list box displays the columns for which you want to set the row sort order.

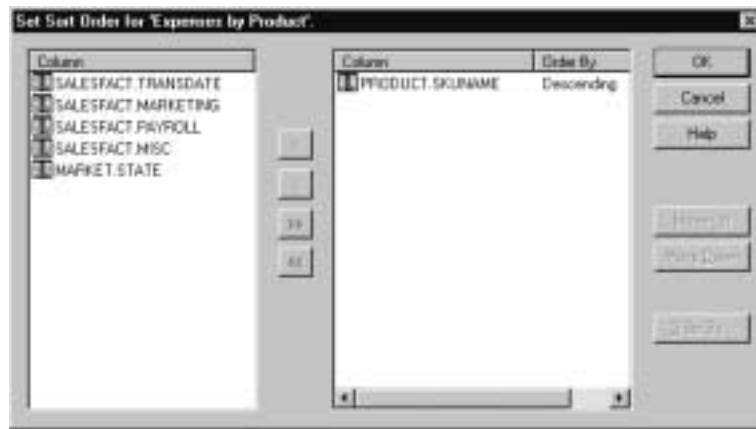





Figure 9-19: Setting Sort Order

4. Select the column for which you want to set the sort order and click  to add the column to the right list box.

To sort all columns, click  to add all columns to the right list box.

To remove a column from the right list box, click .

To not sort any columns, click  to remove all columns from the right list box.

5. Select a column in the right list box, and click **Order By**.

The **Order By** dialog box is displayed.



Figure 9-20: Order By Dialog Box

6. Select the sort order: **Ascending** (A to Z, 0 to 9) or **Descending** (Z to A, 9 to 0).
7. Click **OK** to return to the **Set Sort Order** dialog box.
8. If you want to set the sort order for another column, repeat steps 4, 5, 6, and 7.
9. Click **OK** to return to the **Drill-Through Reports** dialog box.

Specifying Spreadsheet User Actions

You can specify the actions that spreadsheet users can perform after they open a Drill-Through report. You can determine whether spreadsheet users can customize the report, whether they must use a specific server running OLAP Integration Server, and whether they must use a database username and password to log on to the relational data source. This topic contains the following subtopics:

- “Customizing a Report” on page 9-22
- “Specifying the OLAP Integration Server Name” on page 9-23
- “Specifying a Database Username and Password” on page 9-24

Customizing a Report

You can enable spreadsheet users to customize a Drill-Through report that they see in their worksheet. For example, they can choose to exclude certain columns or to change row and column ordering. If you do not enable spreadsheet users to customize a report, they see all columns that you specify and use the row and column ordering that you specify.

- To enable spreadsheet users to customize a report:
 1. In the **Drill-Through Reports** dialog box, select the report that you want to enable spreadsheet users to customize.
 2. Click **Edit** to open the **Report properties** dialog box.

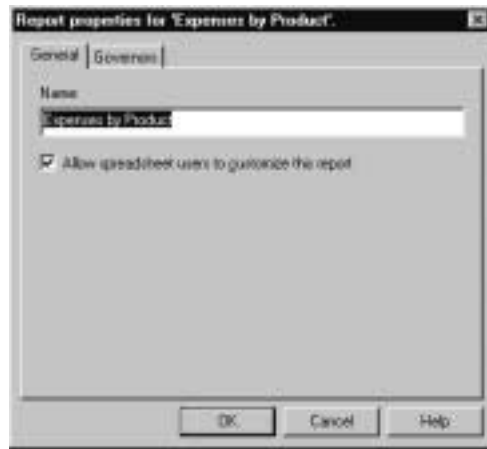


Figure 9-21: Enabling Spreadsheet Users to Customize a Report

3. Select **Allow spreadsheet users to customize this report**.
To prevent spreadsheet users from changing the report, deselect **Allow spreadsheet users to customize this report**.
4. Click **OK**.

Specifying the OLAP Integration Server Name

You can determine whether spreadsheet users must enter the name of the server running OLAP Integration Server to which they want to connect. If spreadsheet users do not enter the OLAP Integration Server name, Hyperion Essbase Spreadsheet Add-in connects to the OLAP Integration Server specified in the Drill-Through report. If you have many users, connecting them to different OLAP Integration Servers enables you to balance the load across multiple OLAP Integration Servers and thereby increase performance.

If you use multiple servers, each server must use the Open Database Connectivity (ODBC) data source name for the external data source and for the OLAP Metadata Catalog that are specified in the OLAP model.

For example, suppose that you create the OLAP model on Integration Server1 and specify the following naming conventions: use OLAP_TBC as the ODBC data source name for the external data source and OLAP_MD as the ODBC data source name for OLAP Metadata Catalog. On Integration Server2, you must also use OLAP_TBC as the ODBC data source name for the external data source and OLAP_MD as the ODBC data source name for OLAP Metadata Catalog.

- To always prompt spreadsheet users for the name of the server running OLAP Integration Server:
 1. In the OLAP Metaoutline main window, select the metaoutline name; for example, TBC Metaoutline.
 2. Right-click the metaoutline name and, from the pop-up menu, select Properties.

The **OLAP Metaoutline Properties** dialog box is displayed.

3. Select the **Drill-Through Reports** tab.



Figure 9-22: Prompting for the Name of the Server Running Hyperion Integration Server

4. Select **Always prompt for OLAP Integration server name**.

To have spreadsheet users connect to the default OLAP Integration Server, deselect **Always prompt for OLAP Integration server name**.

5. Click **OK**.

Specifying a Database Username and Password

You can determine whether spreadsheet users must enter a username and password to log on to the data source. This requirement can provide a layer of security by determining who can access the relational data source.

If you do not choose to prompt for the database username and password, spreadsheet users who connect to the relational database use the username and password of the user who created the Drill-Through report. The user who created the report may have higher access privileges than are appropriate for spreadsheet users.

- To require spreadsheet users to specify a username and a password:
 1. In the OLAP Metaoutline main window, select the metaoutline name; for example, TBC Metaoutline.
 2. Right-click the metaoutline name and, from the pop-up menu, select Properties.

The **OLAP Metaoutline Properties** dialog box is displayed.

3. Select the **Drill-Through Reports** tab.



Figure 9-23: Prompting for a Username and Password

4. Select **Always prompt for database username and password**.

To enable spreadsheet users to use the database username and password of the user who created the Drill-Through report, deselect **Always prompt for database username and password**.


5. Click **OK**.

Working with Reports

You can rename reports, delete reports, and delete parts of a report. This topic contains the following subtopics:

- “Renaming a Report in the Drill-Through Reports Dialog Box” on page 9-26
- “Renaming a Report in the Report Properties Dialog Box” on page 9-27
- “Deleting an Individual Report, an Intersection Level, or a Drill-Through Column” on page 9-28
- “Deleting All Intersection Levels or Drill-Through Columns in a Report” on page 9-28

Renaming a Report in the Drill-Through Reports Dialog Box

- To rename a report in the **Drill-Through Reports** dialog box:
 1. From the menu bar, select Edit > Drill-Through Reports to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, click the report name to put the dialog box in edit mode, .
The screenshot shows a text input field with a list icon on the left and the text 'Report 1' inside. The field is highlighted with a thick border, indicating it is in edit mode.
 3. Type the new report name.
See “About Naming” on page 3-8.
 4. Press the **Enter** key, or select another object in the **Drill-Through Reports** dialog box. Repeat steps 2, 3, and 4.

Renaming a Report in the Report Properties Dialog Box

- To rename a report in the **General** tab of the **Report properties** dialog box:
 1. From the menu bar, select **Edit > Drill-Through Reports** to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, select the report for which you want to change the name.
 3. Click **Edit** to open the **Report properties** dialog box.

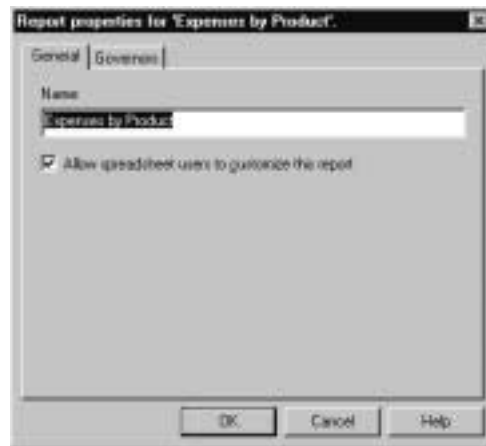


Figure 9-24: Renaming a Report

4. In the **Name** text box, type the new name.
See “About Naming” on page 3-8.
5. Click **OK**.

Deleting an Individual Report, an Intersection Level, or a Drill-Through Column

- To delete a report, an intersection level, or a Drill-Through column:
 1. From the menu bar, select Edit > Drill-Through Reports to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, select the report, intersection level, or column to delete.
 3. Click **Delete**.

A dialog box opens, prompting you to confirm that you want to delete the report, intersection level, or column.
 4. Click **Yes**.

Deleting All Intersection Levels or Drill-Through Columns in a Report

- To delete all intersection levels or Drill-Through columns:
 1. From the menu bar, select Edit > Drill-Through Reports to open the **Drill-Through Reports** dialog box.
 2. In the **Drill-Through Reports** dialog box, select the phrase **OLAP Intersection Levels** or the phrase **Report Drill-Through Columns**.
 3. Click **Delete**.

A dialog box opens, prompting you to confirm that you want to delete all intersection levels or columns.
 4. Click **Yes**.

This chapter suggests solutions for problems that you may discover while working with Hyperion Integration Server Desktop. It contains the following topics:

- “Problems with Creating, Validating, Saving, or Deleting Metaoutlines” on page 10-1
- “Problems with Transformations” on page 10-2
- “Problems with Filters” on page 10-3
- “Failure of a Member Load or Data Load” on page 10-4
- “Problems with Loading Members or Data” on page 10-6

Problems with Creating, Validating, Saving, or Deleting Metaoutlines

If you experience a problem while creating, validating, saving, or deleting a metaoutline, consider the following troubleshooting scenarios:

- If no OLAP model is listed in the Existing tab of the Welcome dialog box, you cannot create a metaoutline. Make sure that you are connected to the correct OLAP Metadata Catalog. See the *Hyperion Integration Server Installation Guide* for information on connecting to OLAP Metadata Catalog.
- If you experience a problem with validating a metaoutline, check to be sure that you are following the validation rules. (See “Validating Metaoutlines Manually” on page 2-19.) If you are following the conditions for validating a metaoutline, view each invalid dimension separately. Be sure that the structure of each dimension looks appropriate. (See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.)

Troubleshooting

- You cannot save or delete a metaoutline if other users are reading it; you receive a message that the metaoutline is write locked. You can perform any of the following tasks:
 - Save or delete the metaoutline later.
 - Use File > Save As to save the metaoutline to a different name. See “Saving a Metaoutline to a Different Name” on page 2-27.
 - Delete the locks. See the *Hyperion Integration Server System Administrator’s Guide* for information on working with locks.
- Are users unable to drill through? If you delete a metaoutline that was used to build a Hyperion Essbase outline, you can no longer drill through on the Hyperion Essbase outline. See Chapter 9, “Creating Drill-Through Reports.”

Problems with Transformations

If member names, dimension names, or data are not being transformed correctly, consider the following troubleshooting scenarios:

- If you set a member to use two-pass calculation and the calculation is not working, remember that, although two-pass calculation is a property that you can give to any member, two-pass calculation works only on members of the Accounts dimension, Dynamic Calc members, and Dynamic Calc and Store members. If you tag any other type of member as Two-pass Calculation, Hyperion Essbase ignores the tag. See the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.
- If you cannot tag a member as Dynamic Time Series, check to be sure that you have set the date-part in the OLAP model. For information on defining transformations for columns, see the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.
- If a formula is working incorrectly, check the syntax carefully. OLAP Integration Server does not validate formula syntax. Errors in a formula do not display until OLAP Integration Server calculates the Hyperion Essbase database. For more information about formula options and built-in functions, see the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.

Problems with Filters

If member names, dimension names, or data are not being filtered correctly, consider the following troubleshooting scenarios:

- Are you connected to OLAP Integration Server? See the *Hyperion Integration Server Installation Guide* for information on starting Hyperion Integration Server Desktop.
- Is the correct metaoutline filter selected for both the member load and the data load? You must use the same metaoutline filter for the member load and the data load.
- If the filter includes extra conditions, make sure that the filter did not inherit conditions from the *Default filter. If the filter did inherit conditions, delete the inherited conditions. See “Creating a Member Filter Interactively” on page 5-14.
- If you experience a syntax error, use the Verify button in the Member Load Filtering or Measure Properties dialog box to check the syntax. View a sample of the dimension on which you defined the filter and check the generated SQL. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.
- Is the generated SQL correct? Use the Sample command to check the generated SQL to make sure that the filter does what you expect (see “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50) or check the SQL statements in the Hyperion Integration Server log file (see the *Hyperion Essbase OLAP Server Database Administrator’s Guide*).

Many filter errors do not show up until you perform a member load or a data load. See also “Problems with Loading Members or Data” on page 10-6.

Failure of a Member Load or Data Load

If a member load or data load is failing, consider the following troubleshooting scenarios:

- Is the metaoutline valid? See “Validating Metaoutlines Manually” on page 2-19.
- Are the external data source, OLAP Metadata Catalog, OLAP Integration Server, and Hyperion Essbase all running correctly? See the *Hyperion Integration Server Installation Guide*.
- Can you connect to the external data source, OLAP Metadata Catalog, OLAP Integration Server, and the Hyperion Essbase server without using Hyperion Integration Server Desktop? See the *Hyperion Integration Server Installation Guide*.
- Are you experiencing a problem viewing a sample outline? Check to be sure that you set the query governors appropriately, especially if you have shared members in the sample outline.
- Is the Hyperion Essbase outline into which you are loading members or data in use? See the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.
- Did you attempt to perform a transformation on members to which you assigned aliases? You cannot perform transformations on members to which you assign aliases. Use OLAP Model to perform needed transformations. For information on defining transformations, see the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.
- Did you receive a time-out error? See the *Hyperion Integration Server System Administrator’s Guide*.
- Does the load run for a long time and then stop? Set query governors to control the duration and size of the queries that OLAP Integration Server makes to the external data source. See “Using Query Governors” on page 7-5.
- If you perform a member load and the Hyperion Essbase outline contains no members, check to see if you used an invalid member name, such as #MISSING. See “About Naming” on page 3-8.

- If a load stops after loading some members or some data, check the reject text file that is automatically created. The reject text file lists all dimensions and members that were not loaded into the Hyperion Essbase outline. See “Procedure 3: Loading Data Immediately” on page 8-8. You can also check the Isolation Level setting in the Hyperion Essbase database. See the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.
- Is a scheduled load not displayed in the Scheduled Jobs dialog box? Make sure that you started the Schedule service on Windows NT. See “Starting the Windows NT Schedule Service Automatically” on page 7-18.
- Do you receive an operating system error when you try to schedule a load? Make sure that `at.exe` is correctly identified in the operating system path. (This program schedules programs to run at a certain time.) If `at.exe` is not correctly identified, correct the problem, restart OLAP Integration Server, and reschedule the load.
- Do some error messages contain error code numbers but no text? If so, check the `message.txt` file in the `ISHOME\esslib\bin` directory. The file contains a column of error code numbers and a column containing the text of error messages.
- View the OLAP Integration Server log file for information about why a load failed. See the *Hyperion Integration Server System Administrator’s Guide*.
- View the Hyperion Essbase error log file for information about why a load failed. See the *Hyperion Essbase OLAP Server Database Administrator’s Guide*.
- View the ODBC log of the data source that is experiencing a problem. If you do not know how to view the ODBC log, see the ODBC documentation.
- Did you try to schedule a member load or data load after someone deleted the `Batch` directory? You must have a `Batch` directory in the `is` directory to schedule loads.

See also “Problems with Loading Members or Data” on page 10-6.

If you experience a problem with the external data source, OLAP Metadata Catalog, OLAP Integration Server, or Hyperion Essbase, contact the appropriate database administrator. After the problem is fixed, restart the load.

Problems with Loading Members or Data

If you have problems loading members or data, check the following troubleshooting scenarios:

- If you perform a member load and the Hyperion Essbase outline contains no members, check to see if you used an invalid member name, such as #MISSING. See “About Naming” on page 3-8.
- Did you perform the load correctly, but some members or data are missing? Check the following troubleshooting scenarios:
 - Did you set the build level incorrectly? See “Setting the Member and Data Load Build Level” on page 4-25.
 - Is the filter filtering out more than you want it to? View the dimensions that contain the incorrect member levels. See “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-50.
 - Did you specify that OLAP Integration Server should create duplicate members as shared? See “Handling Duplicate Member Names” on page 4-6.
 - Did you give an attribute Boolean property or calculation property the same name as a member. See “Setting the Calculations Name Format” on page 3-28 and “Specifying Attribute Boolean Names and Date Formats” on page 3-31.
 - Is any name in the data source longer than 80 characters? Alias names and UDAs must be fewer than 80 characters. You can use the SUBSTRING function in the OLAP model to extract the 80 or fewer characters that you want to use in a member name. For information on defining transformations for string columns as substrings, see the *Hyperion Integration Server Desktop OLAP Model User’s Guide*.
- Did you get a shared member error during loading and are some shared members not loaded? Make sure that all load properties and filters defined on real members are also defined on shared members. See Chapter 4, “Setting Load Properties,” Chapter 5, “Working with Filters,” and Chapter 6, “Setting Hyperion Essbase Properties.”
- Did you perform the load correctly, but there is no data in the Hyperion Essbase database? Check the filters. If a filter filters out all data, a value of #MISSING is displayed in the Hyperion Essbase database. See “Creating a Member Filter Interactively” on page 5-14.

- Did you remember to calculate the data before viewing it in Hyperion Essbase Spreadsheet Add-in? See the *Hyperion Essbase OLAP Server Database Administrator's Guide*.
- Does the load run for a long time and stop? Set query governors to control the duration and the size of the queries that OLAP Integration Server makes to the external data source. See “Using Query Governors” on page 7-5.
- If the load stops after loading some members or data, check the reject text file that is automatically created. The reject text file lists all dimensions and members that were not loaded into the Hyperion Essbase outline. See “Procedure 3: Loading Data Immediately” on page 8-8. You can also check the Isolation Level setting in the Hyperion Essbase database. See the *Hyperion Essbase OLAP Server Database Administrator's Guide*.
- View the Hyperion Integration Server log file for information about the load. For information on the log file, see the *Hyperion Integration Server System Administrator's Guide*.
- View the Hyperion Essbase error log file for information about the load. For information on the log file, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

See also “Failure of a Member Load or Data Load” on page 10-4.

Report any problem to the appropriate database administrator (DBA)—the DBA for the external data source, the DBA for OLAP Metadata Catalog, the DBA for OLAP Integration Server, or the DBA for Hyperion Essbase. After the problem is fixed, restart the load.

Troubleshooting

Creating a Sample Metaoutline Containing Attribute Dimensions

This appendix provides procedures for creating a metaoutline containing attribute dimensions. The sample login information, sample screens, OLAP Model name, metaoutline name, and relational data source are all based on the Hyperion Integration Server sample application.

The Hyperion Integration Server sample application is based on a fictitious company named The Beverage Company (TBC) and contains the following components:

- A relational data source (TBC)
- An OLAP Metadata Catalog (TBC_MD)
- A sample OLAP model (TBC Model)
- A sample metaoutline (TBC Metaoutline)

For information about installing the sample application, see the *Hyperion Integration Server Installation Guide*. For information on creating an OLAP model containing attribute-enabled columns, see the *Hyperion Integration Server Desktop OLAP Model User's Guide*.

Refer to the following topics in this appendix for information about creating a sample metaoutline containing attribute dimensions:

- “Starting up and Logging On” on page A-2
- “Opening the Standard User Interface to Create a New Metaoutline” on page A-3
- “Working in the OLAP Metaoutline Main Window” on page A-5

Creating a Sample Metaoutline Containing Attribute Dimensions

To create the sample metaoutline described in the following procedures, you must have the sample application installed. If you do not have the sample application installed, you can use these procedures as a guide for creating a metaoutline with your own relational data source, OLAP Metadata Catalog, and completed OLAP model.

Starting up and Logging On

To begin the metaoutline creation process, you must first start OLAP Integration Server, and then start and log on to Hyperion Integration Server Desktop. For details on starting up and logging on, see “Starting Hyperion Integration Server Desktop” on page 2-3. You also must have completed an OLAP model on which to base the metaoutline.

After you have logged on, Hyperion Integration Server Desktop displays the Welcome dialog box, as shown in Figure A-1.



Figure A-1: Hyperion Integration Server Desktop Welcome Dialog Box

Opening the Standard User Interface to Create a New Metaoutline

To create a new metaoutline containing attribute dimensions, you must use the standard user interface, rather than the OLAP Metaoutline Assistant.

- To open the OLAP Metaoutline standard user interface:
 1. In the **Welcome** dialog box, select **New**.
 2. Select the **OLAP Metaoutline** icon.
 3. From the drop-down list at the bottom of the dialog box, select the OLAP model on which to base the metaoutline (TBC Model in the sample application), and click **Open**.

The **Data Source** dialog box opens, as shown in Figure A-2.



Figure A-2: Data Source Dialog Box

4. From the **Data Source** drop-down list, select the relational data source to use for creating the metaoutline (TBC for the sample application).
5. From the **User Name** drop-down list, select the user name for creating the metaoutline (tbc for the sample application).

Creating a Sample Metaoutline Containing Attribute Dimensions

6. In the **Password** text box, type the name of your password (**password** for the sample application, all lowercase letters) and click **OK**.

After you have entered the appropriate relational data source information and have connected to the OLAP Metadata Catalog, the OLAP Metaoutline main window opens, as shown in Figure A-3.

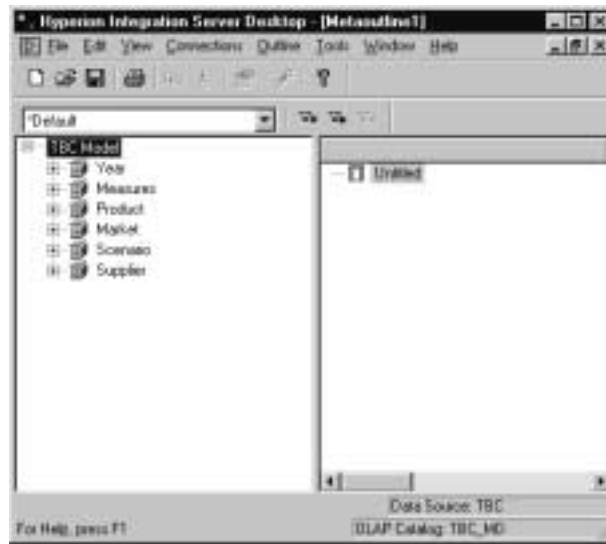


Figure A-3: OLAP Model Dimensions in OLAP Metaoutline Main Window

Working in the OLAP Metaoutline Main Window

This topic contains the following subtopics that describe how to work in the OLAP Metaoutline main window to create a metaoutline containing attribute dimensions:

- “Viewing OLAP Model Dimensions and Attribute-enabled Columns” on page A-5
- “OLAP Model Dimensions with Attribute-Enabled Columns” on page A-6
- “Associating Level 0 Attribute Members with Members of their Base Dimension” on page A-10
- “Setting a Numeric Range for an Attribute Member” on page A-11
- “Using Attribute Calculations” on page A-14
- “Using Attribute Boolean Properties” on page A-16
- “Creating the Measures Dimension” on page A-21
- “Assigning Aliases to Shared Members” on page A-26
- “Saving the Metaoutline” on page A-26

Viewing OLAP Model Dimensions and Attribute-enabled Columns

The left frame of the window displays information about the OLAP model on which you are basing the metaoutline (TBC Model in the sample application). The right frame is blank and is the area where you work to create a metaoutline.

- To view dimensions and attribute-enabled columns:
1. In the OLAP Metaoutline main window, expand the Year, Measures, Product, Market, Scenario, and Supplier dimensions in the left frame by clicking the plus symbol (+) in front of each dimension name.
 2. Expand the Product table under the Product dimension by clicking the plus symbol (+) in front of the name.

As shown in Figure A-4, the attribute-enabled columns are shown indented below Attributes_PRODUCT because these columns from the PRODUCT relational source table were attribute-enabled in the OLAP model.

Creating a Sample Metaoutline Containing Attribute Dimensions

- Expand the attribute-enabled columns for the Market dimension by clicking the plus symbol (+) in front of Attributes_POPULATION.

As shown in Figure A-4, the attribute-enabled columns are shown indented below Attributes_POPULATION because these columns from the POPULATION relational source table were attribute-enabled in the OLAP model.

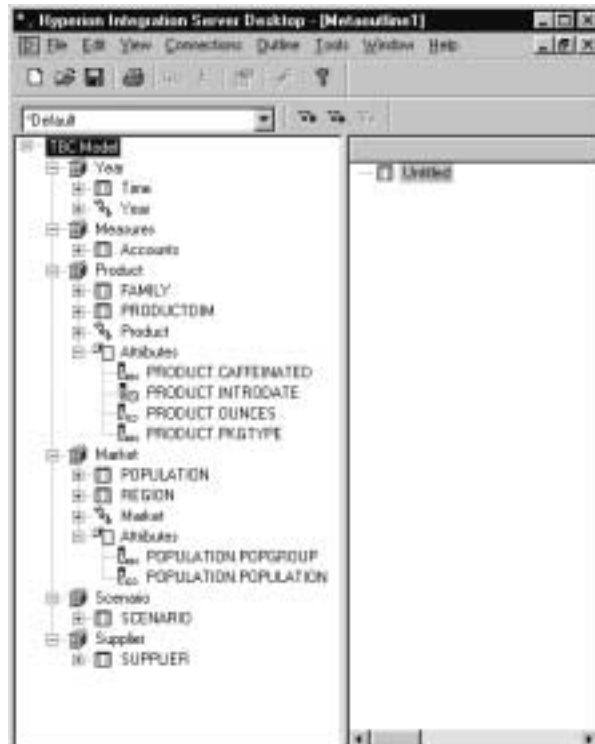


Figure A-4: OLAP Model Dimensions with Attribute-Enabled Columns

Creating Hierarchies, Dimensions, and Attribute Dimensions

➤ To create hierarchies, dimensions and attribute dimensions to build a metaoutline:

1. Drag the Year hierarchy from the left frame to the **Untitled** icon in the right frame and release the mouse button.

Note that dragging an OLAP model hierarchy to the right frame creates a hierarchy in the metaoutline.

2. Drag the Measures dimension from the left frame to the **Untitled** icon in the right frame and release the mouse button.
3. Drag the Product dimension from the left frame to the **Untitled** icon in the right frame and release the mouse button.

The next step is to create the Market dimension in the metaoutline. Creating the Market dimension can be accomplished by using any one of the following methods:

- Drag the Market dimension to the **Untitled** icon in the right frame and release the mouse button.

This action creates a dimension but no hierarchy.

- Drag the Market hierarchy to the **Untitled** icon in the right frame and release the mouse button.

This action creates both a dimension and a hierarchy.

- Drag an attribute-enabled column associated with the Market dimension to the **Untitled** icon in the right frame and release the mouse button.

This action creates both a Market dimension and an attribute dimension and also associates the attribute dimension with its base dimension, Market.

For the sample metaoutline that you are creating, use the third method. Note that the OLAP model has two attribute-enabled columns: POPGROUP and POPULATION.

4. Drag the attribute-enabled POPGROUP column to the **Untitled** icon in the right frame and release the mouse button.

Note that, because the POPGROUP column has a data type of string in the relational data source (identified by the ABC icon), the available data types displayed in the drop-down list are Text and Boolean, with Text displayed as the default data type selection.

OLAP model attribute data types are based on the data type of the relational data source columns. Available attribute data types in the associated metaoutline are based on the Level 0 attribute member type. Data types can be revised in the metaoutline based on the guidelines shown in Table A-1.

Table A-1: Allowable Data Type Changes for Attribute-enabled Columns

Column Data Type/ Essbase Attribute Type	Text	Numeric	Boolean	Date
String	X		X	
Numeric	X	X	X	
Datetime	X			X

5. Drag the attribute-enabled POPULATION column to the POPGROUP column in the right frame and release the mouse button.

Note that because the POPULATION column has a data type of numeric in the relational data source (identified by the 123 icon), the available attribute dimension types in the drop-down list change from Text and Boolean to Text, Numeric, and Boolean, with Numeric displayed as the default selection.

When the available data types displayed in the list box change because an attribute-enabled column with a different Level 0 attribute member data type is added, a warning message is displayed to advise you of the change:



Figure A-5: Attribute Dimension Data Type Change Warning Message

You can accept the change or click **Cancel** to retain the original data type.

Note: When columns of different data types are dragged on top of each other, the common data type (usually the text data type) becomes the default value.

6. Drag the Market hierarchy to the Market dimension in the right frame and release the mouse button.
7. Drag the SCENARIO column under the SCENARIO table to the **Untitled** icon in the right frame and release the mouse button.
8. Drag the attribute-enabled column CAFFEINATED to the **Untitled** icon in the right frame and release the mouse button.

Note that because CAFFEINATED has a data type of string, the default data type assigned in the metaoutline is Text. In the Hyperion Essbase database, CAFFEINATED is a Boolean attribute dimension. Therefore, the attribute dimension type of the CAFFEINATED dimension needs to be changed from Text to Boolean.

To change the data type of the CAFFEINATED dimension, complete these two steps:

- a. Right-click the CAFFEINATED attribute dimension in the right frame and select Properties from the pop-up menu.
 - b. In the **Attribute Dimension Properties** dialog box, select Boolean from the attribute dimension type drop-down list and click **OK**.
9. Drag the attribute-enabled column OUNCES to the **Untitled** icon in the right frame and release the mouse button.

Because OUNCES has a data type of numeric, it is assigned the default data type of Numeric in the metaoutline.

10. Drag the attribute-enabled column PKGTYPE to the **Untitled** icon in the right frame and release the mouse button.

Because PKGTYPE has a data type of string, it is assigned the default data type of Text in the metaoutline.

11. Drag the attribute-enabled column INTRODATE to the **Untitled** icon in the right frame and release the mouse button.

Because INTRODATE has a data type of datetime, it is assigned the default data type of Date in the metaoutline.

Associating Level 0 Attribute Members with Members of their Base Dimension

Now that you have created all the necessary Level 0 attribute members for the various attribute dimensions in the sample metaoutline, these attributes must be associated with members of their base dimension. You can make this base-dimension association in either of two ways:

- By member level
- By metaoutline member

- To associate Level 0 attribute members with base dimension members:
1. Right-click the POPULATION member under the POPULATION attribute dimension and select Properties from the pop-up menu to display the **General** tab of the **Attribute Member Properties** dialog box:



Figure A-6: Attribute Member Properties Dialog Box, General Tab

2. In the **Association** option, select the **By Level** option and select 0 from the drop-down list, and then click **OK**.
3. Right-click the CAFFEINATED member under the CAFFEINATED attribute dimension and select Properties from the pop-up menu.
4. In the **Association** option, select the **By Level** option and select 0 from the drop-down list, and then click **OK**.
5. Right-click the OUNCES member under the OUNCES attribute dimension and select Properties from the pop-up menu.
6. In the **Association** option, select the **By Level** option and select 0 from the drop-down list, and then click **OK**.
7. Right-click the PKGTYPE member under the PKGTYPE attribute dimension and select Properties from the pop-up menu.
8. Select the **Association by Level** option and from the drop-down list under **By Level**, select 0 and click **OK**.
9. Right-click the INTRODATE member under the INTRODATE attribute dimension and select Properties from the pop-up menu.
10. In the **Association** option, select the **By Level** option and select 0 from the drop-down list, and then click **OK**.

Setting a Numeric Range for an Attribute Member

In the sample application supplied with the TBC Metaoutline, the POPULATION.POPULATION size range is divided into population groups of SMALL, MEDIUM and LARGE in the Essbase Outline, with population size intervals of 3,000,000.

- To set the numeric range to intervals of 3,000,000.
 1. Right-click the metaoutline name at the top of the right frame. From the pop-up menu, choose **Attribute Member Names**.

Creating a Sample Metaoutline Containing Attribute Dimensions

2. Select the **Numeric Boundaries** tab of the **Attribute Member Names** dialog box, then select **Tops of ranges** and click **OK**.

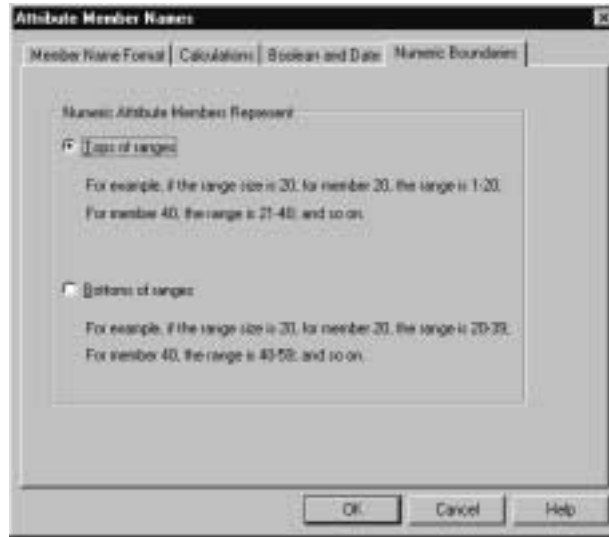


Figure A-7: Attribute Member Names Dialog Box, Numeric Boundaries Tab

3. Right-click the POPULATION.POPULATION attribute member and choose Properties from the pop-up menu.
4. Select the **Numeric Range** tab in the **Attribute Properties** dialog box.
5. Enter **3000000** in the **Start** text box.
6. Select Below from the drop down list in the **Size** text box.
This sets the lowest range to be populations below 3,000,000.
7. Click the **ADD** button on the lower right side of the dialog box.
A new line for selections becomes available.
8. In the new line, enter **3000001** in the **Start** text box and enter **3000000** in the **Size** text box.
9. Click **ADD**.
A new line for selections becomes available.



10. Enter **2000000** in the **Start** text box and select **Above** from the drop down list in the **Size** text box.

This sets all populations above 20,000,000 into a group “above 20,000,000.”

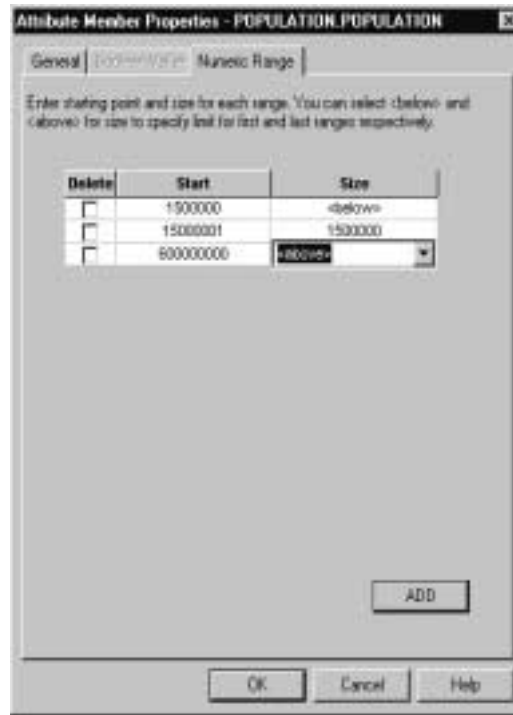


Figure A-8: Attribute Member Properties Dialog Box, Numeric Range Tab

You have set a population grouping of “below 3,000,000.” You also have set a series of groupings of size 3,000,000 between 3,000,001 and 20,000,000. Finally, you have set a single group for all populations above 20,000,000. The results of these settings are visible in the reports and spreadsheets produced with this metaoutline.

Using Attribute Calculations

Attribute members have five predefined calculations; the results of these calculations will appear in spreadsheets and in reports. For example, in the sample application, predefined calculations in the attribute dimension can yield the average yearly sales of 12 ounce bottles of Vanilla Creme Soda. Both Ounces and Package type (bottles) are attribute dimensions, and Average is one of the five types of calculations.

Attribute calculations have the following properties:

- System-defined. When you create an attribute dimension in a metaoutline, the resulting reports and spreadsheets have all five calculations available for all attribute members.
 - Dynamic Calc. All Attribute calculations are calculated when a user requests the calculation, and then the calculation is discarded. You cannot store the calculated data in a database.
 - Displayed in Reports. Attribute calculation results are not displayed in the metaoutline. The calculation results are available only in spreadsheets and reports.
- To define new names for the attribute calculations:
1. Right-click the name of the metaoutline that appears in the right frame of the metaoutline screen.
 2. From the pop-up list, select **Attribute Member Names**.
 3. In the **Attribute Member Names** dialog box, select the **Calculations** tab.

4. Type the new calculation name in the appropriate text box.



Figure A-9: Attribute Member Names Dialog Box, Calculations Tab

The five attribute calculations are described below:

- **Sum** calculates a sum, or total, of the values for a member with an attribute or combination of attributes. Sum supports multiplication and two-pass calculations which may be defined in the member dimension related to that attribute. In most cases, however, attribute Sum is a simple addition of values. If an attribute dimension is specified, but no calculation is specified, Sum is the default dimension that will appear in the spreadsheet.
- **Count** calculates the number of members with the specified attribute or combination of attributes. Count includes only those members for which a data block exists.
- **Average** calculates a mathematical mean, or average, of the non-missing values for a specified attribute or combination of attributes (Sum divided by Count).
- **Min** calculates the minimum data value for a specified attribute or combination of attributes.
- **Max** calculates the maximum data value for a specified attribute or combination of attributes.

Creating a Sample Metaoutline Containing Attribute Dimensions

For further information about using attribute calculations, see the *Hyperion Essbase OLAP Server Database Administrator's Guide*.

Using Attribute Boolean Properties

Attribute Boolean properties allow the user to sort on simple Boolean values, such as Caffeinated True or False.

In the section on “OLAP Model Dimensions with Attribute-Enabled Columns” on page A-6, we changed the data type for Caffeinated from “text” in the model to “Boolean” in the metaoutline. The Caffeinated property is displayed in the metaoutline with a Boolean datatype as shown in Figure A-10.

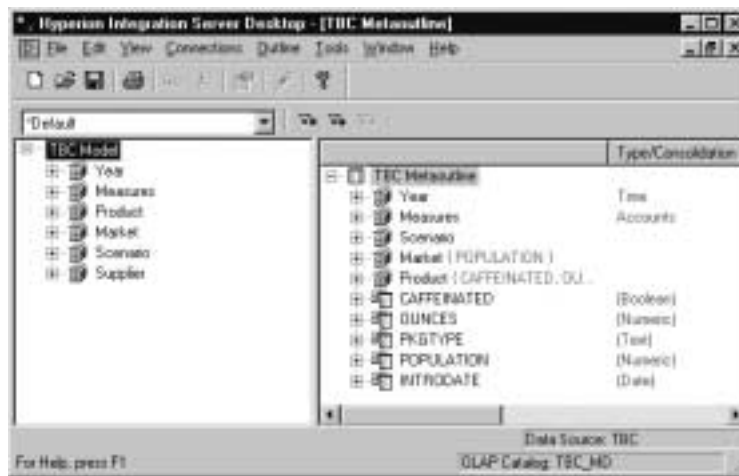
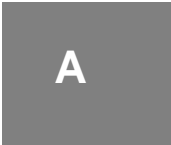


Figure A-10: Metaoutline with Caffeinated as Boolean datatype



The Caffeinated results can be chosen to appear in spreadsheets and reports. Figure A-11 shows Caffeinated True results in the sample Hyperion Essbase Spreadsheet Add-In.

	A	B	C	D	E
1					Measures
2	Actual	Year	True	100-10	22777
3	Actual	Year	True	100-20	5708
4	Actual	Year	True	100	28485
5	Actual	Year	True	200-10	7201
6	Actual	Year	True	200-20	12025
7	Actual	Year	True	200	19226
8	Actual	Year	True	300-10	12195
9	Actual	Year	True	300-20	2511
10	Actual	Year	True	300-30	11093
11	Actual	Year	True	300	25799
12	Actual	Year	True	400	0
13	Actual	Year	True	100-20	5708
14	Actual	Year	True	200-10	7201
15	Actual	Year	True	200-20	12025
16	Actual	Year	True	300-10	12195
17	Actual	Year	True	300-20	2511
18	Actual	Year	True	300-30	11093
19	Actual	Year	True	Diet	28826
20	Actual	Year	True	100-10	22777
21	Actual	Year	True	100-20	5708
22	Actual	Year	True	200-10	7201
23	Actual	Year	True	200-20	12025

Figure A-11: Spreadsheet Results for Caffeinated True

You can also use the **Boolean** tab of the **Attribute Properties** dialog box to express other properties as “True” within a given metaoutline. For example, you can set the PKGTYP so that True equals Bottle. You can also set more elaborate criteria for True, which can be convenient for interpreting complex spreadsheets.

Creating a Sample Metaoutline Containing Attribute Dimensions

- To change the PKGTYPE datatype to Boolean:
 1. Right-click on the PKGTYPE attribute dimension and select Properties from the pop-up menu.
 2. In the **Attribute Dimension Properties** dialog box, select Boolean from the **Attribute Dimension Type** drop-down list.

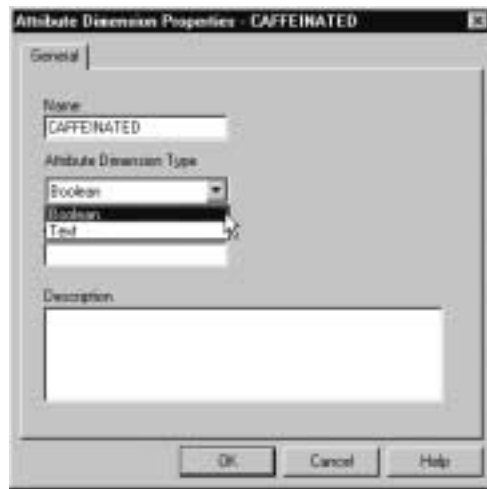


Figure A-12: Attribute Dimension Properties Dialog Box

Now that you have changed the data type to Boolean, you can use the Boolean value tab to set requirements for the reports.

- To set additional Boolean reporting criteria:
 1. Right-click on the PRODUCT.PKGTYPE attribute member and select Properties from the pop-up menu.
 2. In the **Attribute Member Properties** dialog box, select the **Boolean Value** tab.
 3. In the **Apply Values As** option, select **TRUE**.
 4. In the **Operator** drop down list, select = (equal to).
 5. In the **Value** drop-down list, choose Bottle.

6. Click **OK**.



Figure A-13: Attribute Member Properties Dialog Box, Boolean Value Tab

Reports now show Bottle as a Boolean attribute equal to **True**.

Creating a Sample Metaoutline Containing Attribute Dimensions

More elaborate choices can be shown as **True** by selecting more criteria. For example, you can click **Add**, and then select **Or** in the **And/Or** drop-down list, = in the **Operator** list, and NULL in the **Value** drop-down list.

These choices set the True value to apply to both bottles and null value, as shown in Figure A-14, which illustrates the flexibility available in the **Attribute Member Properties** dialog box.

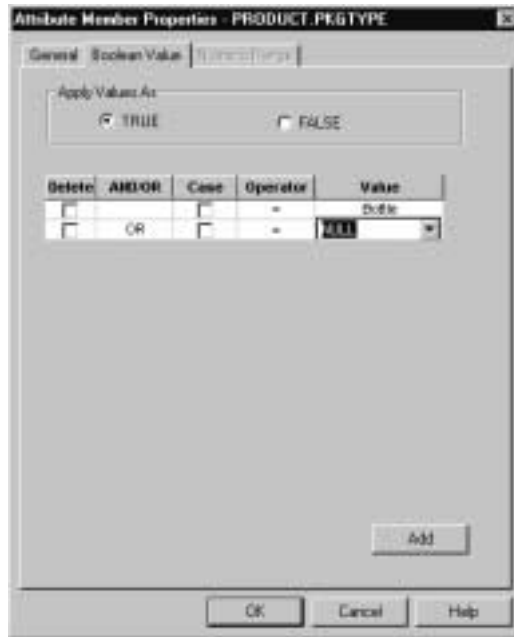




Figure A-14: Attribute Member Properties Dialog Box, Boolean Value Tab with Two Criteria Chosen

Creating the Measures Dimension

The Measures dimension in the sample metaoutline you are creating consists of a combination of columns from the SALESFACT table in the OLAP model, as well as user-defined members.

- To create the Measures dimension in the metaoutline:
1. Select the Measures dimension in the right frame, go to the toolbar and click the **Add member as a child** button, , and type the new member name, **Profit**.
 2. Select the Profit member, go to the toolbar and click the **Add member as a child** button, and type the new member name, **Margin**.
 3. Select the Margin member, go to the toolbar, click the **Add member as a sibling** button, , and type the new member name, **Total Expenses**.
 4. Drag the SALES column under the Accounts table to the Margin member in the right frame and release the mouse button.
 5. Drag the COGS column under the Accounts table to the Margin member in the right frame and release the mouse button.
 6. Drag the MARKETING column under the Accounts table to the Total Expenses member in the right frame and release the mouse button.
 7. Drag the PAYROLL column under the Accounts table to the Total Expenses member in the right frame and release the mouse button.
 8. Drag the MISC column under the Accounts table to the Total Expenses member in the right frame and release the mouse button.

Setting Consolidation and Calculation Methods

Next, you must define the consolidation and calculation methods to be used in determining the following OLAP reporting variances:

- Margin
- Profit

Defining Consolidation and Calculation Methods for Determining the Margin

Margin is equal to SALES - COGS (Cost of Goods Sold). To set the formula for determining the Margin in Hyperion Essbase reports based on the sample metaoutline, you must define the subtraction consolidation method and specify expense variance reporting.

- To set the consolidation and calculation methods for determining the Margin:
 1. Right-click COGS in the right frame, and select Properties from the pop-up menu to display the **Member Properties** dialog box.
 2. Select the **Member Info** tab, as shown in Figure A-15.



Figure A-15: Member Properties Dialog Box, Member Info Tab

3. Under **Consolidation**, select the **- (subtraction)** option and click **OK**.

4. In the **Member Properties** dialog box, select the **Account Info** tab, as shown in Figure A-15.



Figure A-16: Member Properties Dialog Box, Account Info Tab

5. Under **Variance Reporting**, select the **Expense** option and click **OK**.
 The Expense Variance Reporting feature can be used on any expense item member and tells Hyperion Essbase to calculate a favorable or unfavorable variance.

Defining Consolidation and Calculation Methods for Determining the Profit

Next, you must set the consolidation and calculation methods for determining Profit (Margin - Total Expenses) in Hyperion Essbase reports.

- To set the consolidation and calculation methods for determining Profit:
 1. Right-click Total Expenses in the right frame, and select Properties from the pop-up menu.
 2. Select the **Member Info** tab.

Creating a Sample Metaoutline Containing Attribute Dimensions

3. Under **Consolidation**, select the **Subtraction** option, and then, under **Data Storage**, select the **Dynamic Calc** option and click **OK**.
 4. Select the **Account Info** tab and under **Variance Reporting**, select the **Expense** option and click **OK**.
- To complete consolidation and calculation method settings for variance reporting, make the following selections:
1. Right-click **MARKETING** in the right frame, select **Properties** from the pop-up menu, and then select the **Account Info** tab.
 2. Under **Variance Reporting**, select the **Expense** option and click **OK**.
 3. Right-click **PAYROLL** in the right frame, select **Properties** from the pop-up menu, and then select the **Account Info** tab.
 4. Under **Variance Reporting**, select the **Expense** option and click **OK**.
 5. Right-click **MISC** in the right frame, select **Properties** from the pop-up menu, and then select the **Account Info** tab.
 6. Under **Variance Reporting**, select the **Expense** option and click **OK**.
 7. Right-click **Margin** in the right frame, select **Properties** from the pop-up menu, and then select the **Member Info** tab.
 8. Under **Data Storage**, select the **Dynamic Calc** option and click **OK**.
 9. Right-click **Profit** in the right frame, select **Properties** from the pop-up menu, and then select the **Member Info** tab.
 10. Under **Data Storage**, select the **Dynamic Calc** option and click **OK**.
 11. Right-click **Measures** in the right frame, select **Properties** from the pop-up menu, and then select the **Member Info** tab.
 12. Under **Data Storage**, select the **Label Only** option and click **OK**.

Adding Variances to the Scenario Dimension

- To add Variances to the Scenario dimension:
 1. Select the Scenario member in the right frame, click the **Add member as a sibling** button, and type **Variance** as the **New member name**.
 2. Select the Variance member, click the **Add member as a sibling** button, and type **Variance %** as the **New member name**.
 3. Right-click Variance in the right frame, select Properties from the pop-up menu, and then select the **Member Info** tab.
 4. Click the **Two Pass Calculation** checkbox, and under **Data Storage**, select the **Dynamic Calc** option, and click **OK**.
 5. Select the **Formula** tab, type **@VAR(Actual,Budget)** as the formula, and click **OK**.
 6. Right-click Variance % in the right frame, select Properties from the pop-up menu, and then select the **Member Info** tab.
 7. Click the **Two Pass Calculation** checkbox, and under **Data Storage**, select the **Dynamic Calc** option, and click **OK**.
 8. Select the **Formula** tab, type **@VARPER(Actual,Budget)** as the formula, and click **OK**.

Selecting the Build Option

The Product dimension contains shared members (alternate hierarchies). An outline that contains shared members can result in erroneous reporting unless the correct Hyperion Essbase outline build option is selected.

- To select the correct outline build option:
 1. Right-click the Product dimension in the right frame, select Properties from the pop-up menu, and then select the **Outline build** tab.
 2. Under **Duplicate Member Names**, select the **Add as shared members** option and click **OK**.

Assigning Aliases to Shared Members

Both the FAMILY and SKU members of the Product dimension in the sample application have an alias that you assign to them.

► To assign aliases to shared members in the Product dimension:

1. Right-click FAMILY in the right frame, select Properties from the pop-up menu, and then select the **Member Info** tab.
2. Select the FAMILYNAME column from the **alias** drop-down list and click **OK**.

The FAMILY member also has a database column that can be associated as a consolidation attribute.

3. Under **Consolidation**, select **From Database Column**, and then select FAMILYATTR from the drop-down list and click **OK**.
4. Next, right-click SKU in the right frame, select Properties from the pop-up menu, and select the **Member Info** tab.
5. Select the SKUNAME column from the **alias** drop-down list and click **OK**.

Saving the Metaoutline

You have now created a sample metaoutline containing attribute dimensions.

► To save the metaoutline:

1. Select File > Save.
2. In the **Save New Metaoutline** dialog box, type a name for the metaoutline and click **OK**.

To load members and data from the sample metaoutline into Hyperion Essbase, see Chapter 7, “Loading Dimensions and Members” and Chapter 8, “Loading Data.”

#MISSING. *See* missing data (#MISSING).

accounts dimension. A dimension type that makes accounting intelligence available. You can tag only one dimension as accounts; you do not need an accounts dimension.

Add Joins mode. In Hyperion Integration Server, a state in which you can draw lines to define joins between objects in the OLAP model.

agent log file. A record of actions performed by the agent (server).

aggregate. *See* consolidate.

aggregation level. *See* consolidation level.

alias. An alternative name for a dimension, member, or description.

alias column. In Hyperion Integration Server, a column in the data source that contains the aliases for a member level in the metaoutline.

alternate name. *See* alias.

ancestor. A branch member that has members below it. For example, in a dimension that includes years, quarters, and months, the members Qtr2 and 2001 are ancestors of the member April.

application. In Hyperion Essbase, a container for one or more Hyperion Essbase databases and files, such as calc scripts and data load rules, that are related to the databases.

ARBORPATH. An environment variable that specifies the Hyperion Essbase root directory.

Architect. *See* Hyperion Integration Server Desktop.

attribute. A classification of a member in a dimension. You can specify an attribute to select and group members with a specified attribute and to perform calculations and application-specific functions. For example, a Product dimension can have several attributes, such as Size and Flavor. A specific member of the Product dimension can have the Size attribute 8 and the Flavor attribute Cola.

base dimension. A standard dimension that is associated with one or more attribute dimensions. To classify a member of a base dimension, you associate it with a member of one or more attribute dimensions that describes the classification, such as a specific flavor. For example, assuming products have flavors, the Product dimension is the base dimension for the Flavors attribute dimension.

block. The primary storage unit within Hyperion Essbase. A data block is a multidimensional array representing the cells of all dense dimensions.

branch. A member of a hierarchy that may or may not contain leaf members.

Builder. *See* Hyperion Integration Server Desktop.

calc script. *See* calculation script.

calculation script. A text file containing a set of instructions telling Hyperion Essbase how to calculate a database.

Catalog. *See* OLAP Metadata Catalog.

cell. A unit of data representing the intersection of dimensions in a multidimensional database; the intersection of a row and a column in a worksheet.

child. A member that has a parent above it in the database outline. A child may have peers (siblings) that exist at the same layer of the database outline.

column. In relational databases, a vertical part of a table. Also known as a field. A column contains all the values for a specific type of information. Every column has a name and a particular data type. For example, a column may contain the name or employee number for each employee.

Command Interface. *See* OLAP Command Interface.

concatenation. An operation that joins two characters or strings in the order specified, forming one string whose length is equal to the sum of the lengths of the two characters or strings. For example, the strings “New York” and “Library,” when concatenated, become “New York Library.”

condition. In relational databases, a data extraction criterion. For example, you can apply a condition to extract only the data that begins with the letter A.

consolidate. The process of gathering data from dependent entities and aggregating the data to parent entities. After you enter or load data into child entities, you perform a consolidation to aggregate the data through the organization. As data consolidates, intercompany processing, conversion methods, equity adjustments, and minority ownerships perform calculations on the data.

consolidation level. The top of an aggregation hierarchy or any branch or sub-branch below the top, including the input (leaf) portion of the hierarchy.

data block. *See* block.

data cleansing. The process of making inconsistent data consistent. Examples of inconsistent data are data in which some values are incorrect or are not of the correct data type.

data file. A file containing data blocks; Hyperion Essbase generates the data file during a data load and stores it on disk.

data load. The process of populating a Hyperion Essbase database with data. Loading data establishes actual values for the cells defined by the structural outline of the database.

data load rules. A set of criteria or rules that Hyperion Essbase uses to determine how to load data from a text-based file or a relational data set into a Hyperion Essbase database.

data point. *See* cell.

data source. External data, such as a text file, spreadsheet file, or SQL database that will be loaded into a Hyperion Essbase database.

data type. Defines the kind of data that a column can contain. For example, columns with the Numeric data type contain numbers.

data value. *See* cell.

database. A collection of related information. Each unit (record) of the database is typically organized in a fixed format to make it easy to retrieve selected portions of the data on demand. Each record is made up of one or more data fields, and each data field can hold one piece of data (known as a value).

database administrator. An individual who administers database servers, such as Hyperion Essbase, and who may also design, maintain, and create databases.

database outline. *See* outline.

DateTime transformation. A set of instructions that defines how to change or reformat a relational database DateTime data type to your choice of date format.

DBMS. *See* relational database management system (RDBMS, DBMS).

denormalization. The process of adding redundancy to data in a database, typically by joining tables to form more complete sets of data in the individual tables. This process is performed for the purpose of increasing data retrieval performance. *Contrast with* normalization.

dense dimension. A dimension likely to contain data for every combination of dimension members. For example, a time dimension is typically a dense dimension because it contains all combinations of all members. *Contrast with* sparse dimension.

descendant. Any member below a parent in the database outline. For example, in a dimension that includes years, quarters, and months, the members Qtr2 and April are descendants of the member Year.

detail member. *See* leaf member.

dimension. A data category that is used to organize business data for retrieval and preservation of values. Each dimension usually contains a hierarchy of related members grouped within it. For example, a Year dimension often includes members for each time period, such as quarters and months. Other common business dimensions may be measures, natural accounts, products, and markets.

dimension branch. A collection of dimension tables organized in a hierarchical structure, with one of the dimension tables joined directly to the fact table. A dimension branch defines a single, potential Hyperion Essbase dimension in a Hyperion Integration Server metaoutline.

dimension build rules. In Hyperion Essbase, a set of operations similar to data load rules. Instead of loading data, the dimension build rules modify the outline based on data in the external data source file.

dimension table. A container in the OLAP model for one or more relational tables that define a potential Hyperion Essbase dimension. When one dimension table joins to another dimension table, forming a dimension branch, the dimension is composed of the columns of all the dimension tables in the dimension branch.

dimension type. A property in the OLAP model and in the metaoutline that defines which tables form the Hyperion Essbase accounts, time, and standard (default) dimensions.

duplicate member. The second occurrence of a member name in a data source. Users can determine whether Hyperion Integration Server ignores duplicate members or adds them as shared members. *See also* shared member.

Dynamic Calc And Store members. A member that the Hyperion Essbase server calculates only upon the first retrieval of the value. The server then stores the calculated value in the database. Subsequent retrievals of a Dynamic Calc And Store member do not require calculating.

Dynamic Calc members. A member that the Hyperion Essbase server calculates only at retrieval time. The server discards calculated values after the retrieval request is complete.

dynamic calculation. In Hyperion Essbase, a calculation that occurs only when you retrieve data on a member that has been tagged as Dynamic Calc or Dynamic Calc And Store. The values for that member are calculated at retrieval time instead of being precalculated during batch calculation.

Dynamic Time Series. A process that is used to perform dynamic period-to-date reporting.

Dynamic Time Series members. Predefined members that are used to perform Dynamic Time Series reporting.

Essbase database. A repository of data within Hyperion Essbase that contains a multidimensional data storage array. Each database consists of a defined storage structure (a database outline), data, security definitions, and other associated files, such as calculation scripts or data load rules. *See also* application.

Essbase OLAP server. A database server that locates and accesses multidimensional data.

Essbase outline. *See* outline.

fact table. In Hyperion Integration Server, a container for one or more relational tables that define the data values for each dimension intersection in the OLAP model. For example, if the OLAP model contains Products, Region, and Year dimensions, the fact table might include data values for the number of units of Product A sold in New York in January.

field. (1) In Hyperion Essbase, a value or item in a data source file that is retrieved from a Hyperion Essbase database. (2) In relational databases, a space allocated for a particular item of information. Fields are the smallest units of information that you can access. Most fields have certain characteristics associated with them. For example, some fields are numeric, whereas others are textual. Every field has a name.

file delimiter. One or more characters, such as a comma (,), separating fields in a data source.

filter. (1) In Hyperion Essbase, a method for controlling access to database cells. A filter is the most detailed level of security, allowing you to define the varying access levels that users can have to data. (2) In Hyperion Integration Server and in relational databases, a method for controlling which data you retrieve from a relational database. For example, you can choose to retrieve products only in the Product A product family. A filter contains one or more conditions.

foreign key. In relational databases, a column whose data values correspond to the values of a key column in another relational table. *See also* key column *and* primary key.

formula. In Hyperion Essbase, a combination of operators and functions as well as dimension names, member names, and numeric constants. Formulas are used to calculate relationships between members of a Hyperion Essbase database. @VAR(Actual, Budget) is an example of a formula.

generation. A layer in a hierarchical tree structure that defines member relationships in a Hyperion Essbase database. Hyperion Essbase orders generations incrementally from the dimension (generation 1) down to the leaf members.

generation name. A unique name that describes a generation.

hierarchy. A set of multidimensional relationships in an outline, often created in a tree formation. For example, parents, children, and generations represent a hierarchy.

Hyperion Essbase kernel. A layer of the Hyperion Essbase server that provides the foundation for a variety of functions, including data loading, calculations, spreadsheet lock&send, partitioning, and restructuring. The Hyperion Essbase kernel reads, caches, and writes data; manages transactions; and enforces transaction semantics to ensure data consistency and data integrity.

Hyperion Integration Server Desktop. The client component of the Hyperion Integration Server product family. This tool is used to create OLAP models and metaoutlines, and to load data into Hyperion Essbase databases.

index. (1) In Hyperion Essbase, a method of retrieving data based on sparse dimensions. Also refers to the index files, collectively. (2) In relational databases, pointers that are logically arranged by the values of a key. Indexes optimize access to relational data.

index cache. In Hyperion Essbase, a buffer in memory that holds index pages.

index entry. In Hyperion Essbase, a pointer to an intersection of sparse dimensions. Each index entry points to a block on a disk and locates a particular cell within the block by means of an offset.

index file. In Hyperion Essbase, a file used to store data retrieval information. It resides on a disk and contains index pages.

index page. In Hyperion Essbase, a subdivision of an index file containing entries that point to data blocks.

input block. A type of data block that contains at least one loaded data value.

input data. Any data that is loaded from a data source and is not generated by calculation.

Integration Server. *See* OLAP Integration Server.

integrity constraint. In relational databases, a rule stating that each row must contain an entry for each required key column.

intersection level. In the Hyperion Essbase Spreadsheet Add-in, a Hyperion Essbase member combination that defines a specific value. For example, the member combination Actual, Root Beer, Sales, Jan, East represents the actual January sales value for root beer in the Eastern region.

join. In relational databases, a relationship between two tables based on matching key column values.

join columns. In Hyperion Integration Server, two relational table columns that are joined from one table to another.

key column. In relational databases, a column or columns that form a unique identifier for each row. For example, EMPLOYEE_ID might be a key column.

leaf member. A member that has no children.

left frame. (1) In the Hyperion Integration Server Desktop OLAP Metaoutline main window, the area on the left that enables you to view a list of dimensions previously defined in the OLAP model. (2) In the OLAP Model main window, the area on the left that displays a list of the tables and views available in your source relational database.

level. (1) In Hyperion Essbase, a layer of a hierarchical tree structure that defines database member relationships. Hyperion Essbase numbers levels incrementally from the leaf member (level 0) up to the dimension member. (2) In Hyperion Integration Server, the distance of a dimension table from the fact table in the OLAP model. For example, if a table is level 2, it is separated from the fact table by two links.

level 0 block. A data block that is created for sparse member combinations when all the members of the sparse combination are level 0 members.

level 0 member. *See* leaf member.

level name. A unique name that describes a level.

load member. In Hyperion Integration Server, a member in a user-defined dimension into which data is loaded. Only user-defined dimensions require load members. For all non-user-defined dimensions, Hyperion Integration Server is capable of loading members and data into the Hyperion Essbase database.

load properties. In Hyperion Integration Server, a set of rules that determine what actions the product performs on member level names and data as they are loaded.

logical column. In Hyperion Integration Server, a column created by manipulating the data in one or more physical columns. *See also* column. *Contrast with* physical column.

logical table. In relational databases, a table created by manipulating columns from one or more physical tables. The logical table is only a view of the data; the columns remain stored in the original tables and are not physically duplicated in the logical table. *See also* view. *Contrast with* physical table.

mathematical operator. A symbol that defines how data is calculated. A mathematical operator can be any of the standard mathematical or Boolean operators, for example, +, -, *, /, and %. Mathematical operators are used in formulas and outlines.

MDDB. *See* multidimensional database (MDDB).

measures. Data values that a user wants to track, such as Unit_Price and Discount. By default, measures values map to the accounts dimension in the OLAP model, which maps to the measure dimension in the OLAP metaoutline, which in turn maps to the accounts dimension in the Hyperion Essbase outline.

measures dimension. In Hyperion Integration Server, a dimension that, by default, maps to the accounts dimension in the Hyperion Essbase outline.

member. A discrete component within a dimension. For example, a time dimension might include members such as Jan, Feb, and Qtr1.

member combination. In Hyperion Essbase, a list of member names used to specify a set of data at the intersection of two or more dimensions. A member combination is specified by using the cross-dimensional operator -> (a hyphen followed by a right-angle bracket). For example, the actual sales data for the month of January in Sample Basic is Sales->Jan->Actual.

member level. A hierarchical level of detail within a dimension. For example, in a dimension that defines geographic areas by nation, which are then subdivided into provinces, the nation and province categories each represent a member level. A member level corresponds to a level in a Hyperion Essbase outline. The measures dimension contains actual members that are also member levels.

member load. The process of adding new dimensions or members (without data) to a Hyperion Essbase outline. *Contrast with* data load.

metadata. The components of a database outline that describe and hold data. Examples of metadata are dimension names, member names, properties, time periods, and security.

metaoutline. In Hyperion Integration Server, a template containing the structure and rules for creating a Hyperion Essbase outline from an OLAP model.

missing data (#MISSING). A marker indicating that data in the labeled location does not exist, contains no meaningful value, or was never entered or loaded. For example, missing data exists when an account contains data for a previous or a future period but not for the current period.

Move mode. In Hyperion Integration Server, a state in which you can pick up, move, and drop objects in the OLAP Model main window.

multidimensional. Describes a method of referencing data through three or more dimensions. An individual data value is the intersection of one member from each dimension.

multidimensional database (MDDB). A type of database that enables referencing data through three or more dimensions. An individual record is the intersection of a point for a set of dimensions. *Contrast with* relational database.

normalization. The process of grouping and removing redundancy from data so that each entity is in its appropriate place in the database and only in its appropriate place. *Contrast with* denormalization.

numeric transformation. In Hyperion Integration Server, a set of instructions that define how to change or reformat a relational database numeric field type. For example, you may choose to divide numeric data by 100.

ODBC. *See* Open Database Connectivity (ODBC).

ODBC data source. Location of the data that you are accessing and the information necessary to access the data using ODBC. For example, an ODBC data source that connects to an SQL server might require a server name, a database name, a user name, and a password. *See also* Open Database Connectivity (ODBC).

OLAP. *See* online analytical processing (OLAP).

OLAP Architect. *See* Hyperion Integration Server Desktop.

OLAP Builder. *See* Hyperion Integration Server Desktop.

OLAP Catalog. *See* OLAP Metadata Catalog.

OLAP Command Interface. In Hyperion Integration Server, a command-line tool that you can use to perform common operations on the Hyperion Essbase outline and the data in the Hyperion Essbase database. For example, you can use the LOADDATA command to load data.

OLAP Integration Server. The server component of the Hyperion Integration Server product family. OLAP Integration Server uses the information stored in OLAP Metadata Catalog to extract the dimension names and member names needed to build a Hyperion Essbase outline from the data source. When the Hyperion Essbase outline is complete, OLAP Integration Server extracts data from the data source, performs the operations specified in the metaoutline, and loads the data into the Hyperion Essbase database.

OLAP Metadata Catalog. In Hyperion Integration Server, a relational database containing metadata describing the nature, source, location, and type of data that you pull from the relational data source. OLAP Integration Server accesses OLAP Metadata Catalog to generate the SQL statements and the information required to generate a Hyperion Essbase database outline.

OLAP model. In Hyperion Integration Server, a logical model (star schema) that you create from tables and columns in a relational database. You can then use the OLAP model to generate the structure of a multidimensional database.

OLTP. *See* online transaction processing (OLTP).

online analytical processing (OLAP). A multidimensional, multiuser, client-server computing environment for users who need to analyze consolidated enterprise data in real time. OLAP systems feature drilling down, data pivoting, complex calculations, trend analyses, and modeling.

online transaction processing (OLTP). OLTP applications are commonly referred to as data capture, data entry, or data collection applications. OLTP applications enable an organization to capture the large amounts of data resulting from its daily activities but provide limited capability for reporting on the data.

Open Database Connectivity (ODBC). Standardized application programming interface (API) technology that allows applications to access multiple third-party databases.

outline. The structure that defines all elements of a database within Hyperion Essbase. For example, an outline contains definitions of dimensions, members, and formulas.

parent. A member that has an aggregated branch below it.

pass-through transformations. In Hyperion Integration Server, a feature that enables you to use functions specific to your relational database management system (RDBMS) to extract data values for columns. You can provide a statement that Hyperion Integration Server passes through to your RDBMS as a part of the SQL SELECT statement. You provide the statement as a property of a column and the pass-through feature returns a value for the column.

permission. A special privilege that must be assigned to users or groups to enable them to access or modify secure data. Permissions include Read, Read/Write and None.

physical column. A column that is stored in a relational database. *See also* column. *Contrast with* logical column.

physical table. A combination of rows and columns stored in a relational database. *Contrast with* logical table.

pointer. In relational databases, a data element that indicates the location of data in storage.

primary dimension table. A dimension table that joins directly to the fact table. Additional dimension tables may join to the primary dimension table to create a dimension branch.

primary key. In relational databases, a column (key) that uniquely identifies a row. For example, Employee_ID.

query governor. In Hyperion Integration Server, a parameter that you set to control the time and size of the queries made to the data source.

RDBMS. *See* relational database management system (RDBMS, DBMS).

record. A set of information in a data source. Records are composed of fields, each of which contains one item of information. A set of records constitutes a table. A single record constitutes a row in the table. For example, a table containing personnel information might contain records (rows) that have three fields: a NAME field, an ADDRESS field, and a PHONE_NUMBER field.

recursive table. A source relational table that contains information in one row that is a parent or child of information in another row. For example, in a relational source table containing the columns EMPLOYEE_ID, NAME, and MANAGER_ID, the columns EMPLOYEE_ID and MANAGER_ID are recursive because MANAGER_ID refers back to the EMPLOYEE_ID. Using Hyperion Integration Server, you can build a Hyperion Essbase outline hierarchy from a recursive source table.

relational database. A type of database that stores data in the form of related tables. A single database can be spread across several tables, and can be viewed in many different ways. *Contrast with* multidimensional database (MDDB).

relational database management system (RDBMS, DBMS). A system for accessing data in a relational database. This term is often used to describe systems for management of multiple relational databases or tables.

restructure. In Hyperion Essbase, an operation to regenerate or rebuild the database index and, in some cases, the data files.

right frame. In the Hyperion Integration Server Desktop OLAP Metaoutline main window, the area on the right, in which you build a metaoutline. In the OLAP Model main window, the area on the right, in which you build an OLAP model.

roll-up. *See* consolidate.

schema. In relational databases, a logical model that represents the data and the relationships between the data.

server. *See* Essbase OLAP server *and* OLAP Integration Server.

shared member. A member that shares storage space with another member of the same name. The shared member has a property that designates it as shared. The use of shared members prevents duplicate calculation of members that occur more than once in a Hyperion Essbase outline.

sibling. A child member within a dimension, having the same parent as another child member. For example, the members East and West are both children of the Markets dimension and siblings of each other.

sparse dimension. A dimension unlikely to contain data for all combinations of dimension members. For example, product and market dimensions are sparse if not all products are sold in all markets. *Contrast with dense dimension.*

SQL. *See* structured query language (SQL).

staging area. A database that you create to meet the needs of a specific application. A staging area is a snapshot or restructured version of one or more RDBMSs.

standard dimension. A dimension that is not an attribute dimension.

star schema. In Hyperion Integration Server, a logical model that represents your relational data in a form that mirrors OLAP. A star schema contains a fact table and one or more dimension tables.

storage manager. *See* Hyperion Essbase kernel.

string. A sequence of characters treated as a unit.

structured query language (SQL). A computer language used to access data in relational databases.

supervisor. A defined type of user who has full access to all applications, databases, related files, and security mechanisms for a server.

system administrator. A person who maintains the hardware, software, disk space distribution, and configurations for running software applications such as Hyperion Essbase.

table. In relational databases, a form of data storage in which data is stored in rows and columns.

time dimension. A dimension type that defines how often data is collected and updated, such as per fiscal or calendar periods. Only one dimension may be tagged as time; a time dimension is not required.

transformation rules. In Hyperion Integration Server, a set of instructions that define how to change or reformat the member names and data that you extract from the source relational database.

two-pass calculation. A Hyperion Essbase property that is used to recalculate members that are dependent on the calculated values of other members. Two-pass members are calculated during a second pass through the database outline.

UDA. Formerly called *user-defined attribute*. A UDA is a term associated with members of an outline to describe a particular characteristic of the members. Users can specify UDAs within calculation scripts and reports to return lists of members that have the specified UDA associated with them. UDAs can be applied to dense as well as sparse dimensions.

unary operator. A group of mathematical indicators (+, -, *, /, %) that define how consolidations take place on the database outline.

union. In relational databases, a type of join that combines the results of two SELECT statements. A union is often used to merge lists of values contained in two tables.

upper-level block. A type of data block that is created for sparse member combinations when at least one of the sparse members is a parent-level member.

user-defined attribute. *See* UDA.

user-defined dimension. Dimensions that you explicitly create in Hyperion Integration Server, rather than dimensions obtained or built from the relational data source.

user-defined member. Members that you explicitly create in Hyperion Integration Server, rather than obtaining and building them from the relational data source.

validation. (1) In Hyperion Essbase, a process of checking a rules file against the outline to make sure that the rules file is valid. (2) In Hyperion Integration Server, a process of checking the OLAP model and metaoutline.

view. In relational databases, a logical table created by combining columns from one or more tables.

Index

Symbols

- ! (exclamation points), in names in scripts and formulas, 3-10
- " (double quotation marks)
 - for terms in scripts and formulas, 3-10
 - in dimension and member level names, 3-8
 - in dimension and member names, 4-15, 4-18
 - in scripts and formulas, 3-10
- #MISSING
 - aggregate filters and, 5-7
 - in Hyperion Essbase databases, 10-6
 - properties, time balance and, 6-21
 - reserved word, 10-4
 - skipping during calculations, 6-21
 - zeros, skip properties, 6-21
- \$\$\$UNIVERSE\$\$\$
 - reserved word, 3-9
- % (percent signs)
 - consolidation property, 6-16
 - Hyperion Essbase formulas and, 6-28
 - in names in scripts and formulas, 3-10
- () (parentheses)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- * (asterisks)
 - consolidation property, 6-16
 - in names in scripts and formulas, 3-10
- + (plus signs)
 - consolidation property, 6-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
 - reversing, 4-29
- . (periods)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- / (slashes)
 - consolidation property, 6-16
 - in names in scripts and formulas, 3-10
- : (colons), in names in scripts and formulas, 3-10
- ; (semicolons), in names in scripts and formulas, 3-10
- < (less than signs)
 - filter operator, 5-15
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- <= (less than or equal to), filter operator, 5-15
- <> (not equal to) filter operator, 5-15
- = (equal signs)
 - filter operator, 5-15
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- > (greater than signs)
 - filter operator, 5-15
 - in names in scripts and formulas, 3-10
- >= (greater than or equal to), filter operator, 5-15
- @ (at signs)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- @VAR function, 6-23
- @VARPER function, 6-23
- [] (brackets), in names in scripts and formulas, 3-10
- \ (backslashes)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18

Index

- _ (underscores)
 - converting spaces to, 4-17
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- { } (braces)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- | (vertical bars)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- ~ (tilde), consolidation property, 6-16
- (dashes)
 - consolidation property, 6-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- (hyphens)
 - in names in scripts and formulas, 3-10, 4-18
- (minus signs), reversing, 4-29
- ' (single quotation marks)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- , (commas)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10

A

access

- See also* read access
- setting, 2-25
- specifying for OLAP models, 2-25

accessing

- data sources, 2-4
- help online, xiv

accounts dimensions

- dense dimension, 6-8
- dimension type, 6-4
- measure tag, 3-23
- properties, shared members and, 4-6
- two-pass calculation and, 6-15, 10-2

Add New Intersection Levels dialog box, 9-11

Add New Report Columns dialog box, 9-13

aggregate filters, problems with, 10-6

aggregation options, defined, 1-3

aliases

- assigning, 6-12, A-26
- attribute dimensions, setting, 3-36
- attribute members, setting, 3-38
- defined, 6-12
- dimensions, setting, 6-12
- members, setting, 6-12
- reports and, 6-12
- shared members and, 4-6, A-26
- transformations and, 10-4

alphabetizing members, 4-24

applications

- sample, xiii
- specifying, 7-9, 8-4

ascending sort, Drill-Through reports and, 9-19

assigning, aliases, 6-12

asterisks (*)

- consolidation property, 6-16
- in names in scripts and formulas, 3-10

at signs (@)

- in dimension and member level names, 3-9
- in dimension and member names, 4-15, 4-18
- in names in scripts and formulas, 3-10

at.exe, 10-5

Attribute Dimension Properties dialog box, 3-35, A-18

attribute dimensions

- creating, 3-26
- data type, 3-35
- in metaoutline, 3-4
- in sample application, A-1
- overview, 3-8
- setting aliases, 3-36

Attribute Member Names dialog box

- Boolean and Date tab, 3-31
- Calculations tab, 3-28, A-15
- Member Name Format tab, 3-27
- Numeric Boundaries tab, 3-33, A-12

Attribute Member Properties dialog box

- Boolean Value tab, 3-40, A-19, A-20
- General tab, 3-38, A-10
- Numeric Range tab, 3-42, A-13

- attribute members
 - associating with base dimension, 3-26, 3-37, A-10
 - Boolean names, 3-31
 - calculations, 3-28, A-14, A-15
 - compared with UDAs, 3-25
 - incremental load, 7-11, 8-6
 - leaf node and, 3-39
 - multiple numeric ranges, 3-42
 - naming, 3-27
 - setting aliases, 3-38
 - setting Boolean parameters, 3-39
 - setting numeric ranges, 3-42, A-11
 - several criteria in Boolean, 3-39
 - attribute-enabled columns, A-5, A-6
 - attributes
 - Boolean reporting criteria, A-18
 - Boolean type, A-8, A-16
 - case-sensitive criteria, 3-40
 - creating, 3-12
 - date type, A-8
 - filter example, 5-28
 - numeric type, A-8
 - troubleshooting, 10-6
 - automatic loads, scheduling, 7-17, 8-12
 - average, time balance property, 6-20
- B**
- backslashes (\)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - balance
 - beginning, 6-20
 - beginning and ending, 6-20
 - Batch directory, scheduled loads and, 10-5
 - batch processing, OLAP Command Interface, 1-9
 - beginning balance, 6-20
 - Boolean parameters
 - attributes, 3-39
 - braces ({ })
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
 - brackets ([]), in names in scripts and formulas, 3-10
- build level
 - problems with, 10-6
 - setting, 4-25
 - build option, A-25
- C**
- calc scripts
 - names with special characters, 3-10
 - setting default, 7-13, 8-8
 - UDAs and, 6-25
 - calculations
 - attributes, 3-28
 - methods, A-21, A-23
 - order of, 6-14
 - problems, 10-7
 - case
 - case-sensitive criteria in Boolean attributes, 3-40
 - case-sensitive names, 3-8
 - case-sensitive searches, 4-15
 - changing, 4-22
 - characters
 - forbidden in a name, 3-9, 4-15, 4-18, 7-4
 - maximum in dimension names, 3-8
 - maximum in member names, 3-8
 - quoting in scripts and formulas, 3-10
 - children
 - attaching a name to a parent, 4-20
 - shared members and, 4-6
 - chronological sorting, 4-16
 - closing balance, 6-20
 - collapsing, member branches, 2-12
 - colons (:), in names in scripts and formulas, 3-10
 - columns
 - adding, 9-13
 - capitalization, xiii
 - changing for measures, 4-30
 - creating dimensions from, 3-12
 - creating metaoutlines from, 3-12
 - deleting, 9-28
 - Drill-Through and, 9-2
 - example of, 9-17
 - ordering, 9-17
 - valid for Drill-Through reports, 9-13

Index

- command line
 - loading data from, 8-7
 - loading members from, 7-12
- commas (,)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- conditions
 - extra, problems, 10-3
 - measure filters and, 5-24
 - multiple, measure filters and, 5-25
 - multiple, member filters and, 5-16
 - selecting multiple, 5-16, 5-25
- connecting
 - problems with, 2-6
 - to data sources, 2-4
 - to OLAP Metadata Catalog, 2-4
- Connections menu
 - OLAP Metadata Catalog > Connect command, 2-28
 - OLAP Metadata Catalog > Delete Locks command, 2-29
- consolidation
 - defined, 1-3
 - drill down, 1-4
 - example, 1-3
 - members, 1-3
 - methods, A-21, A-23
 - properties, list of, 6-16
 - properties, shared members and, 4-6
- container filters, 5-4
- converting, spaces to underscores, 4-17
- copying
 - Drill-Through reports, 9-5
 - metaoutlines, 2-27
- creating
 - attribute dimensions, 2-10, 3-26, A-7
 - attributes from attribute-enabled columns, 3-12
 - calc scripts, 7-13, 8-8
 - dimensions, A-7
 - dimensions from columns, 3-12
 - hierarchies, 3-11, A-7
 - hierarchies from columns, 3-12
 - Hyperion Essbase outlines, 7-1
 - Hyperion Essbase outlines using filters, 5-8
 - Hyperion Essbase users, 7-10, 8-5

- measure dimensions, 3-23
- measure filters, 5-7
- measures, 3-11
- measures from columns, 3-12
- member filters, 5-6
- member levels, 3-11
- metaoutline filters, 5-10
- metaoutlines, A-3
- UDAs, 6-26
- user-defined dimensions and members, 3-18
- user-defined members, 3-18

curly braces. *See* braces

customer comparisons, 1-2

customizing, Drill-Through reports, 9-22

D

- daily, member and data load, 7-18
- dashes (-)
 - consolidation property, 6-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- data
 - checking, 3-51
 - drill down, 1-4
 - drill up, 1-4
 - loading. *See* data loading
 - missing, in sample databases, xiii
 - pivoting to change view, 1-5
 - storage properties, 6-9
- Data Load dialog box, 8-10
- data loading
 - attributes, 8-6
 - build level, setting, 4-25
 - calculation and, 10-7
 - daily, 7-17, 8-12
 - defined, 8-1
 - deleting scheduled loads, 7-19, 7-22
 - diagram, 7-3
 - failing, 10-4
 - Hyperion Essbase outlines, 7-1, 8-1
 - immediately, 8-8
 - incremental, 8-6
 - load member and, 3-21
 - members, new, 4-8

- metaoutline filters and, 5-5
- OLAP Command Interface and, 7-12, 8-7
- other sources, 3-7
- partial load, 10-5, 10-7
- prerequisites for, 8-3
- problems, 10-3, 10-6
- rules file, 7-16, 8-11
- scheduling, 8-12
- scheduling problems, 10-5
- starting, 8-4
- stopping, 8-10, 10-4
- text file, 7-16, 8-11
- viewing scheduled loads, 7-19, 7-22
- weekly, 7-17, 8-12
- Data Source dialog box, A-3
- data sources
 - accessing, 2-4
 - filtering data from, 5-1
 - Hyperion Essbase, specifying, 7-9, 8-4
 - names, OLAP Integration Server and, 9-23
 - OLTP databases, 1-5
 - sample, xiii
 - security for Drill-Through reports, 9-24
 - spreadsheet files, 1-5
 - SQL, 1-5
 - text files, 1-5
 - tuning, 3-51
 - username, Drill-Through reports and, 9-24
 - workflow for transferring, 1-9
- data type
 - attribute, 3-35
 - attribute dimension, 3-36
 - attributes and OLAP model, 3-36
- data values
 - See also* measures
 - defined, 3-6
 - extracting from a relational data source, 7-3
 - filtering, 5-7
 - manipulating, 4-28
 - multiple in measure filters, 5-25
 - reversing signs, 4-29
 - scaling, 4-28
 - selecting multiple in filters, 5-16
- databases. *See* data sources
- dataload.txt, 8-10, 10-5, 10-7
- date format, 3-31, 3-33, 7-18, 8-13
- decimal points, moving, 4-28
- default calc script, setting, 7-13, 8-8
- Delete an OLAP Model/Metaoutline dialog box, 2-28
- Delete Locks command, 2-29
- deleting
 - dimensions, 3-49
 - dimensions during loading, 7-9
 - measures, 3-49, 3-53
 - member levels, 3-49
 - member levels from hierarchies, 3-17
 - members during loading, 7-9
 - metaoutline filters, 5-13
 - metaoutlines, confirming, 2-29
 - metaoutlines, open by other users, 2-26, 2-29
 - metaoutlines, procedures, 2-28
 - problems with, 10-2
 - scheduled loads, 7-19, 7-22
- dense dimensions
 - accounts, 6-8
 - defined, 6-8
 - time, 6-8
- descending sort, Drill-Through reports and, 9-19
- descriptions, of metaoutlines, 2-25
- detail data
 - Drill-Through, 9-2
 - retrieving, 1-4
- dialog boxes
 - Add New Intersection Levels, 9-11
 - Add New Report Columns, 9-13
 - Attribute Dimension Properties, 3-35, A-18
 - Attribute Member Names, 3-28, 3-31, 3-33, A-12, A-15
 - Attribute Member Properties, 3-38, 3-40, 3-42, A-10, A-13, A-19, A-20
 - Data Load, 8-10
 - Data Source, A-3
 - Delete an OLAP Model/Metaoutline, 2-28
 - Dimension Properties, 4-7, 4-10, 6-7, 6-9, 6-11, 6-19, 6-26
 - Drill-Through Reports, 9-3
 - Edit Filter, 4-25, 5-14, 5-20, 5-21, 5-22, 5-29
 - Edit Hierarchy, 3-5
 - Edit Transformations, 4-14, 4-19, 4-22
 - Enter the new dimension's name, 3-19
 - Enter the new filter's name, 5-10
 - Essbase Application and Database, 7-8, 8-4

Index

- Essbase Users, 7-10, 8-5
- Incremental Update, 7-12
- Measure Properties, 3-48, 4-28, 4-29, 4-31, 5-23, 5-26
- Member Load, 7-15
- Member Properties, 3-46, 6-13, 6-15, 6-17, 6-19, 6-22, 6-24, 6-26, 6-28, A-22, A-23
- OLAP Metaoutline Properties, 3-22, 3-52, 7-6, 9-15
- Order By, 9-21
- Preview Filter Results, 3-51, 5-20, 5-23
- Preview Transformation Results, 3-51, 4-16
- Report Properties, 9-15, 9-27
- Sample Outline, 3-51
- Save New Metaoutline, 2-24
- Schedule Essbase Load, 7-14, 7-17, 8-9, 8-12
- Scheduled Jobs, 7-20, 7-22
- Select a Member, 3-22
- Select Values From, 5-16, 5-24, 5-30
- Services, 7-19
- Set Sort Order, 9-20
- View Existing Job, 7-21
- View Filter, 5-12, 5-31
- dimension building. *See* member loading
- dimension names, maximum length, 3-8
- Dimension Properties dialog box
 - Account Info tab, 6-19
 - Member Info tab, 6-7, 6-9, 6-11
 - Outline Build tab, 4-7, 4-10
 - UDA tab, 6-26
- dimension storage, specifying, 6-8
- dimension types
 - accounts, 6-4
 - standard, 6-4
 - time, 6-5
- dimensions
 - adding, 3-1
 - aliases, setting, 6-12
 - attribute, 3-4, A-1, A-7
 - attributes, 2-10, 3-8
 - creating, 3-12
 - defined, 1-3, 1-10, 3-4
 - deleting, 3-49
 - deleting during load, 7-9
 - dense, 6-8
 - Drill-Through reports, 9-7
 - examples, 1-3
 - extracting from a relational data source, 7-3
 - formulas on, 6-27
 - measures, creating, 3-23, A-21
 - members and, 1-3
 - minimum number in metaoutline, 2-20
 - naming rules, 3-8
 - ordering, 3-12
 - rejected, 7-15, 8-10, 10-5, 10-7
 - renaming, 3-44
 - sparse, 6-8
 - TBC Metaoutline and, 1-12
 - two-pass calculations, specifying, 6-14
 - types of, 6-3
 - use in consolidation, 1-3
 - user-defined, 3-7, 3-18
 - viewing sample, 3-50
- directories
 - default, 7-16, 8-11
 - default metaoutline, 2-3
 - is, 2-3
 - ISADMIN, 7-16, 8-11
- dividing, values while loading, 4-28
- documentation, xiii, xiv
- double quotation marks ("")
 - in dimension and member level names, 3-8
 - in dimension and member names, 4-15, 4-18
 - terms in scripts and formulas, 3-10
- Drill, 9-13
- drill down, 1-4
- drill up, 1-4
- Drill-Through columns
 - adding, 9-13
 - deleting, 9-28
 - deleting all, 9-28
 - example of, 9-17
 - ordering, 9-17
- Drill-Through reports
 - copying, 9-5
 - creating, 9-1, 9-3
 - customizing, 9-22
 - database username and, 9-24
 - deleting, 9-28
 - deleting metaoutlines and, 2-29, 10-2
 - load balancing, 9-23
 - missing, 10-2

- OLAP Integration Server name, 9-23
 - renaming, 9-26, 9-27
 - rows, sort order of, 9-19
 - sample, 9-8
 - security and, 9-24
 - valid columns for, 9-13
 - Drill-Through Reports dialog box, 9-3
 - Drill-Through, defined, 9-1
 - Drop leading/trailing spaces option, 4-17
 - dummy members, creating, 4-9
 - duplicate members
 - defined, 4-3
 - handling, 4-6
 - ignoring, 4-3
 - problems with, 10-6
 - Dynamic Calc
 - attribute calculations, 3-28
 - property, 6-10
 - Dynamic Calc and Store, property, 6-10
 - Dynamic Time Series
 - problems with, 10-2
 - setting, 6-18
- E**
- Edit Filter dialog box
 - Filters tab, 5-14, 5-20, 5-21, 5-22, 5-29
 - Ordering tab, 4-25
 - Edit Hierarchy dialog box, 3-5
 - Edit menu
 - Add Member as Child menu command, 3-19
 - Add Member as Sibling menu command, 3-20
 - Drill-Through Reports command, 9-3
 - Transformation command, 4-14
 - Edit Transformations dialog box, 4-14
 - Prefix/Suffix tab, 4-19
 - Replace tab, 4-14, 4-22
 - ending balance, 6-20
 - Enter the new dimension's name dialog box, 3-19
 - Enter the new filter's name dialog box, 5-10
 - equal signs (=)
 - filter operator, 5-15
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
 - Essbase Application and Database dialog box, 7-8, 8-4
 - Essbase outlines. *See* Hyperion Essbase outlines
 - Essbase Users dialog box, 7-10, 8-5
 - Essbase. *See* Hyperion Essbase
 - exclamation points (!), in names in scripts and formulas, 3-10
 - Exclusive Access
 - opening with, 2-5
 - saving with, 2-21
 - executable name
 - for Hyperion Integration Server Desktop, 2-3
 - expanding, member branches, 2-12
 - expense
 - property, 6-23
 - variance reporting property, 6-23
 - extra, dimensions and members, 3-7
- F**
- fields, adding prefixes or suffixes to, 4-18
 - File menu
 - Delete command, 2-28
 - New command, 2-4
 - Open command, 2-4
 - Print command, 2-27
 - Save As command, 2-23, 2-27
 - Validate command, 2-19
 - filtering
 - data sources, 5-1
 - data values, 5-7
 - members, 5-6
 - filters
 - See also* metaoutline filters
 - attribute, 5-28
 - conditions, creating multiple, 5-10
 - conditions, extra, 10-3
 - default, 5-4, 5-9, 5-12, 5-14
 - defined, 5-1
 - grouping, 5-4
 - Hyperion Essbase databases, creating multiple, 5-5
 - measure, example, 5-7
 - member, example, 5-3
 - metaoutline and data loading, 5-4
 - naming rules, 3-8

Index

- operators in, 5-15
- problems with, 10-6
- row, defined, 5-6
- shared members, problems and, 10-6
- SQL, 5-4, 5-19
- viewing current metaoutline, 5-8
- viewing in a metaoutline, 5-12
- flipping, plus and minus signs, 4-29
- forecasting, 1-4
- formulas
 - dimensions and, 6-27
 - names with special characters, 3-10
 - problems with, 10-2
 - setting, 6-27
 - shared members and, 4-6
 - user-defined members and, 6-27
- From Database Column, consolidation property, 6-16
- G**
 - generated SQL, viewing, 3-51
 - greater than or equal to (\geq), filter operator, 5-15
 - greater than signs ($>$)
 - filter operator, 5-15
 - in names in scripts and formulas, 3-10
- H**
 - Help menu
 - Help Topics command, xiv
 - hierarchies
 - consolidation and, 1-3
 - creating dimensions from, 3-12
 - creating, example, 3-15
 - creating, sample, A-7
 - defined, 1-11
 - deleting member levels from, 3-17, 3-49
 - Hyperion Essbase
 - applications, specifying, 7-9, 8-4
 - data views supported, 1-2
 - database, no data loaded, 10-6
 - databases, creating multiple from a single metaoutline, 5-5
 - databases, specifying, 7-9, 8-4
 - databases, workflow for creating, 1-9
 - dimensions, defined, 3-4
 - displaying properties in main window, 2-13
 - example outline, 3-6
 - members, compared to metaoutline member levels, 1-11
 - overview of, 1-3
 - properties, setting, 6-1
 - rejected dimensions or members, 7-15, 8-10, 10-5, 10-7
 - server, interaction with, 7-3
 - strategic planning, 1-4
 - users, creating, 7-10, 8-5
 - viewing properties, 6-29
 - viewing properties in columnar format, 2-14, 6-29, 6-30
 - viewing properties in condensed format, 6-29, 6-31
 - what if scenarios, 1-4
 - Hyperion Essbase OLAP Server, uses, 1-3
 - Hyperion Essbase outlines
 - building, xii, 7-1
 - creating smaller, 4-25
 - loading, 7-1, 8-1
 - problems writing to, 10-4
 - rejected dimensions or members, 7-15, 8-10, 10-5, 10-7
 - sample, problems with, 10-4
 - shared members, position of, 4-6
 - updating incrementally, 7-11, 8-6
 - viewing sample, 3-50
 - Hyperion Integration Server
 - components, 1-8
 - defined, 1-5
 - OLAP Command Interface, defined, 1-9
 - OLAP Model Assistant, defined, 1-7
 - product family, defined, 1-6
 - product family, diagram, 1-7
 - sample application, A-1
 - standard user interface, defined, 1-8
 - workflow, 1-6, 1-9
 - Hyperion Integration Server Desktop
 - defined, 1-1
 - executable name, 2-3
 - hyphens
 - in dimension and member names, 4-15, 4-18
 - hyphens (-)
 - in dimension and member level names, 3-9
 - in names in scripts and formulas, 3-10

I

- icons, 3-13
 - filter, 2-14
 - metaoutline startup, 2-3
 - transformation, 2-14
- ignoring
 - duplicate members, 4-3
 - new members, 4-9
- in, filter operator, 5-15
- incremental member load
 - attribute members, 7-11
 - attributes, 8-6
 - members and dimensions, 7-11
 - selecting, 8-6
- Incremental Update dialog box, 7-12
- intersection levels
 - adding, 9-6, 9-11
 - defined, 9-2
 - deleting, 9-28
 - deleting all, 9-28
 - rules for, 9-7
 - setting, 9-7
- intersections, defined, 1-3
- invalid metaoutlines, saving, 2-19
- is directory, 2-3
- Isolation Level setting, 10-5, 10-7

J

- jobs, viewing scheduled, 7-20, 7-22

L

- label only property, 6-10
- last time balance property, 6-20
- leaf members, setting, 4-27
- leaf node
 - attribute members and, 3-39
- leaf node, example, rejected dimensions or members, 7-16, 8-11
- left frame
 - defined, 2-12
 - selecting multiple objects, 2-15
- less than or equal to (<=), filter operator, 5-15

- less than signs (<)
 - filter operator, 5-15
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- levels
 - Drill-Through reports and, 9-7
 - level 0 members, setting, 4-27
 - setting build, 4-25
- linked partitions, 6-3
- load
 - OLAP command interface and, 8-7
- load balancing, Drill-Through reports and, 9-23
- load members
 - setting, 3-21
 - user-defined dimensions, 3-7
 - validation and, 2-21
- load properties
 - build level, 4-25
 - columns, changing for measures, 4-30
 - defined, 4-1
 - dimensions, setting, 4-3
 - duplicate member names, 4-3
 - measures, manipulating, 4-28
 - measures, setting, 4-27
 - member names
 - converting spaces to underscores, 4-17
 - determining capitalization, 4-22
 - dropping spaces, 4-16
 - prefixes and suffixes, 4-18
 - replacing, 4-13
 - setting, 4-11
 - members, new, 4-8
 - reversing plus and minus signs, 4-29
 - shared members, 4-6
 - sorting members, 4-24
- LOADDATA command, 8-7
- loading
 - attribute members, 8-6
 - build level, setting, 4-25
 - daily, 7-17, 8-12
 - defined, 8-1
 - deleting scheduled loads, 7-19, 7-22
 - Hyperion Essbase outlines, 8-1
 - immediate data load, 8-8
 - immediate member load, 7-13

Index

- load members and, 3-21
 - metaoutline filters and, 5-5
- OLAP Command Interface and, 7-12
- other sources, 3-7
- scheduling, 7-17
 - scheduling data loads, 8-12
 - viewing scheduled loads, 7-22
 - weekly, 7-17, 8-12
- loading process, 7-3
- LOADMEMBER command
 - example of, 7-12
- loads, viewing scheduled, 7-20
- locking, Hyperion Essbase outlines and, 10-4

M

- Match Whole Word option, in a replace operation, 4-15
- maximum rows per query, setting, 7-6, 9-16
- maximum time per query, setting, 7-6, 9-16
- measure dimension, creating, 3-23
- measure filters
 - See also* filters
 - conditions and, 5-24
 - creating, 5-7
 - metaoutline filters and, 5-4
 - operators in, 5-15
 - typing directly, 5-27
- Measure Properties dialog box
 - Filters tab, 5-23, 5-26
 - General tab, 3-48, 4-31
 - Transformations tab, 4-28, 4-29
- measure tag, 2-20, 3-23
- measures
 - adding, 3-1
 - column, changing, 4-30
 - creating dimensions from, 3-12
 - defined, 1-10, 3-6
 - deleting, 3-49, 3-53
 - Drill-Through reports and, 9-7
 - editing, 3-53
 - filtering, 5-7
 - maximum allowed, 2-20
 - metaoutlines, 3-15
 - ordering, 3-16
 - renaming, 3-44

- renaming a single, 3-47
- required number in metaoutline, 2-20
- user-defined members and, 3-7
- viewing, 3-52
- member filters
 - See also* filters
 - creating, 5-6
 - metaoutline filters and, 5-4
 - operators in, 5-15
 - typing directly, 5-22
- member levels
 - adding, 3-1
 - attributes, 3-39
 - defined, 1-11, 3-5
 - example of, 3-5
 - ordering, 3-16
 - renaming, 3-44
- Member Load dialog box, 7-15
- member load, stopping, 7-15
- member loading
 - attributes, 7-11, 8-6
 - build level, setting, 4-25
 - command line, 7-12
 - defined, 7-1
 - deleting scheduled loads, 7-19, 7-22
 - failing, 10-4
 - Hyperion Essbase outline, 8-1
 - immediately, 7-13
 - improving performance of, 7-11, 8-7
 - incomplete, 10-6
 - incremental, 8-6
 - incrementally, 7-11
 - metaoutline filters and, 5-5
 - new members, 4-8
 - partial load, 10-5, 10-7
 - prerequisites for, 7-4
 - problems, 10-3, 10-6
 - rules file, 7-16, 8-11
 - scheduling, 7-17
 - scheduling problems, 10-5
 - sorting, performance and, 4-25
 - starting, 7-8
 - stopping, 8-10, 10-4
 - text file, 7-16, 8-11
 - user-defined load member, 3-7
 - viewing scheduled loads, 7-19, 7-22

- member names, maximum length, 3-8
- Member Properties dialog box
 - Account Info tab, 6-22, 6-24, A-23
 - Formula tab, 6-28
 - General tab, 3-46
 - Member Info tab, 6-13, 6-15, 6-17, 6-19, A-22
 - UDAs tab, 6-26
- members
 - aliases, setting, 6-12
 - alphabetizing, 4-24
 - branches, collapsing or expanding, 2-12
 - changing case of, 4-12, 4-22
 - checking before loading, 3-51
 - consolidation, 1-3
 - consolidation, operators, 6-16
 - creating. *See* creating, members
 - data loading and, 4-8
 - defined, 1-3, 3-5
 - dimensions, 1-3
 - duplicate, defined, 4-3
 - examples, 1-3
 - extracting from a relational data source, 7-3
 - filtering, 5-6
 - ignoring duplicate, 4-3
 - loading. *See* member loading
 - missing, 10-4
 - names, filtering on, 5-6
 - naming rules, 3-8
 - ordering, 4-24
 - prefixes and, 4-18
 - rejected, 7-15, 8-10, 10-5, 10-7
 - sample, problems with, 10-4
 - shared. *See* shared members
 - sorting, 4-24
 - suffixes and, 4-18
 - transforming, 4-12
 - two-pass calculation, 6-14
 - user-defined, 3-18
 - viewing sample, 3-50
- menu bar, defined, 2-13
- menus, right-click, 2-13
- metadata repository. *See* OLAP Metadata Catalog
- metadata, defined, 1-9
- metadata, OLAP Metadata Catalog storage, 1-9
- metaoutline filters
 - See also* filters
 - creating, 5-10
 - creating multiple Hyperion Essbase databases
 - using, 5-5
 - default, 5-4, 5-9, 5-12, 5-14
 - defined, 5-1, 5-4
 - deleting, 5-13
 - measure filters and, 5-4
 - member filters and, 5-4
 - multiple conditions in, 5-10
 - naming rules, 3-8
 - problems with, 10-3
 - setting current, 5-8
 - viewing SQL statements, 5-19
 - viewing the contents of, 5-12
- Metaoutline main window
 - customizing view, 2-13, 6-29
 - defined, 2-11
 - example of, 2-11
 - multiple selection in left frame, 2-15
 - multiple selection in right frame, 2-16, 2-17
 - properties displayed in columnar format, 2-13
 - properties displayed in condensed format, 2-13
 - working in, A-5
- metaoutlines
 - advantages of, 1-10
 - components of, 1-10
 - confirming a deletion, 2-29
 - copying, 2-27
 - creating, 2-11, A-3
 - defined, 1-8
 - deleting, 2-28
 - deleting, problems with, 10-2
 - description of, 2-25
 - filters, viewing, 5-12
 - Hyperion Essbase databases, creating multiple, 5-5
 - Hyperion Integration Server sample, xiii
 - introducing, 1-1
 - locks, deleting, 2-29
 - measures in, viewing, 3-52
 - navigating, 2-12
 - opening, 2-4
 - populating, 3-1

Index

- prerequisites for, 1-1
- printing, 2-27
- renaming, 2-27
- sample, xiii
- saving invalid, 2-19
- saving with Exclusive Access, 2-21
- saving, problems with, 10-2
- shared members in, 4-3
- using icon to start, 2-3
- viewing current, 5-8
- workflow to create a metaoutline, 1-12
- working with, 2-3

minus signs (-)

- consolidation property, 6-16
- in dimension and member level names, 3-9
- in dimension and member names, 4-15, 4-18
- in names in scripts and formulas, 3-10
- reversing, 4-29

Miscellaneous members, 4-8

missing

- See also* #MISSING

models. *See* OLAP models

monthly, member and data load, 7-18

months, sorting chronologically, 4-16

multidimensional databases

- defined, 1-3
- source data, 1-5
- uses, 1-3
- workflow, 1-5, 1-9

multiple selection

- left frame, 2-15
- menu commands availability, 2-17
- right frame, 2-16

multiplying, values while loading, 4-28

N

naming conventions, case sensitivity, 3-8

naming rules

- applications and databases, 7-4
- attribute calculations, 3-28
- members, 3-8
- metaoutlines, 3-8
- user-defined dimensions, 3-9

navigating metaoutlines, 2-12

negative data values, reversing signs, 4-29

- never share property, 6-10
- non expense, variance reporting property, 6-23
- not equal to filter operator (<>), 5-15
- not in filter operator, 5-15
- Note, 3-27
- numeric ranges
 - attributes, 3-42
 - start and size, 3-43

O

ODBC

- data sources, multiple OLAP Integration Servers and, 9-23
- SQL versions supported, 1-5

OLAP Architect. *See* OLAP Model

OLAP Builder. *See* metaoutlines

OLAP Catalog. *See* OLAP Metadata Catalog

OLAP Command Interface

- batch processing, 1-9
- defined, 1-9
- loading data with, 8-7
- loading members with, 7-12

OLAP Command Interface commands

- LOADDATA, 8-7
- LOADMEMBER, 7-12

OLAP Integration Server

- connection problems, 2-6, 10-3
- defined, 1-8
- Drill-Through reports and, 9-23
- ODBC data source names, 9-23

OLAP intersection levels

- adding, 9-6, 9-11
- defined, 9-2
- deleting, 9-28
- deleting all, 9-28
- rules for, 9-7
- setting, 9-7

OLAP Metadata Catalog

- configuring for ODBC, 1-9
- connecting to, 2-28
- connection problems, 2-6
- defined, 1-9
- locks, deleting, 2-29

OLAP Metaoutline Assistant, 1-8, 2-7

- OLAP Metaoutline Properties dialog box
 - Database Measures tab, 3-52
 - Drill-Through Reports tab, 9-24, 9-25
 - Governors tab, 7-6, 9-15
 - User Defined Dimensions tab, 3-22
- OLAP metaoutline. *See* metaoutlines
- OLAP Model
 - startup programs necessary, 2-3
 - uses, 1-1
- OLAP model, 3-36
- OLAP Model Assistant, 1-7
 - defined, 1-7
- OLAP models
 - attribute data types, 3-37, A-8
 - dimensions associated with, 3-12
 - dimensions not associated with, 3-7, 3-18
 - Hyperion Integration Server sample, xiii
 - members, 1-3
 - members associated with, 3-15
 - members not associated with, 3-7, 3-18
 - missing, 10-1
 - sample, xiii
 - specifying user access permissions, 2-25
 - viewing dimensions, A-5
- OLAP reporting variances, A-21, A-23
- OLAP Server, defined, 1-2
- olapbldr program, 2-3
- OLTP data, defined, 1-5
- Online Analytical Processing (OLAP), defined, 1-2
- online guide, xiv
- online help, xiv
- Online Transaction Processing (OLTP), examples of
 - contents, 1-5
- Open Database Connectivity. *See* ODBC
- opening balance, 6-20
- operational data, 1-5
- operations, order of, 4-12
- operators
 - consolidation, 6-16
 - member and measure filters, 5-15
- order
 - calculations and, 6-14
 - transformations performed, 4-12
- Order By dialog box, 9-21

- ordering
 - columns in Drill-Through reports, 9-17
 - dimensions, 3-12
 - measures, 3-16
 - member levels, 3-16
 - members, 4-24
 - rows in Drill-Through reports, 9-19
- Outline menu
 - Data Load command, 8-4
 - Member Load command, 7-8
- outline tree, collapsing or expanding a member
 - branch, 2-12
- outlines. *See* Hyperion Essbase outlines

P

- parentheses
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
- parents
 - attaching name to children, 4-20
 - rolling up into different, 4-4
 - two-pass calculation and, 6-14
- partitions, 6-3
- percent signs (%)
 - consolidation property, 6-16
 - Hyperion Essbase formulas, 6-28
 - in names in scripts and formulas, 3-10
- performance
 - build level and, 4-25
 - Drill-Through reports and, 9-23
 - sorting and, 4-25
 - tuning, data sources, 3-51
- periods (.)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- permissions
 - changing, 2-27
 - saving metaoutlines and, 2-25
- pivoting
 - defined, 1-5
 - using to change data view, 1-5

Index

- plus signs (+)
 - consolidation property, 6-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - in names in scripts and formulas, 3-10
 - reversing, 4-29
 - populating, metaoutlines, 3-1, 7-1, 8-1
 - pop-up menu, defined, 2-13
 - positive data values, reversing sign, 4-29
 - prefixes
 - adding to fields, 4-18
 - attribute names, 3-27
 - example, 4-12
 - new member names and, 4-10
 - operations, order of, 4-13
 - prerequisites
 - data, loading, 8-3
 - metaoutlines, creating, 7-4
 - Preview Filter Results dialog box, 3-51, 5-20, 5-23
 - Preview Transformation Results dialog box, 3-51, 4-16
 - Print button, Metaoutline toolbar, 2-27
 - printing
 - metaoutlines, 2-27
 - SQL, 3-51
 - privileges. *See* permissions
 - problems
 - connecting, 2-6
 - deleting metaoutlines, 2-29, 10-2
 - loading members and data, 10-6
 - product comparisons, 1-2
 - properties
 - accounts, 6-19
 - consolidation, 6-16
 - dimension storage, 6-8
 - Dynamic Calc, 6-10
 - Dynamic Calc and Store, 6-10
 - expense, setting, 6-23
 - first time balance, 6-20
 - Hyperion Essbase, 6-1
 - label only, 6-10
 - last time balance, 6-20
 - load. *See* load properties
 - never share, 6-10
 - skip, 6-21
 - time balance, 6-20
 - two-pass calculation, 6-14
 - variance reporting, 6-23
 - zeros, 6-21
- ## Q
- query governors
 - defined, 7-5, 9-15
 - problems with, 10-4
 - setting, 7-5, 9-15
 - query size, controlling, 7-5, 9-15
 - query time, controlling, 7-5, 9-15
 - quotation marks
 - in dimension and member level names, 3-8
 - in dimension and member names, 3-9, 4-15, 4-18
 - in scripts and formulas, 3-10
- ## R
- read access, setting, 2-25
 - read/write access. *See* read access
 - recursive dimensions, 3-17
 - recursive hierarchies
 - defined, 4-4
 - example of, 4-4
 - reject text file, 7-15, 8-10, 10-5, 10-7
 - relational databases. *See* data sources
 - removing. *See* deleting
 - renaming
 - attribute calculations, 3-29
 - dimensions, 3-44
 - Drill-Through reports, 9-26, 9-27
 - measures, 3-44
 - member levels, 3-44
 - metaoutlines, 2-27
 - single measure, 3-47
 - repeated member names. *See* duplicate members
 - replace operations, order of, 4-13
 - replacing case-sensitive strings, 4-15
 - replicated partitions, 6-3
 - Report Properties dialog box
 - General tab, 9-27
 - Governors tab, 9-15
 - report scripts
 - names with special characters, 3-10
 - user-defined attributes and, 6-25

reporting variances, A-21, A-23

reports

- aliases and, 6-12
- attributes in, 3-44
- copying, 9-5
- creating, 9-1, 9-3
- customizing, 9-22
- database username and, 9-24
- deleting, 9-28
- Drill-Through, missing, 10-2
- load balancing, 9-23
- OLAP Integration Server name, 9-23
- renaming, 9-26, 9-27
- security and, 9-24
- valid columns for, 9-13

repository. *See* OLAP Metadata Catalog

reserved word

- \$\$\$UNIVERSE\$\$\$, 3-9
- Other, 3-9

reserved words, 3-9

restructure, incremental, 7-11, 8-6

result sets, controlling size of, 7-5, 9-15

reversing, plus and minus signs, 4-29

right frame

- defined, 2-12
- selecting multiple objects, 2-16, 2-17

right-click menus, xvi, 2-13

roll-ups

- average property and, 6-20
- defined, 1-3
- last property and, 6-20

row filters

- defined, 5-6

rows

- example of, 9-19
- maximum in query, setting, 7-5, 9-15
- none returned, 7-6
- ordering, 9-19

rules

- intersection levels and, 9-7
- naming, 3-8
- naming applications and databases, 7-4
- shared members, 4-6

rules file, 7-16, 8-11

S

sample

- applications, xiii, A-1
- Drill-Through reports, 9-8
- Hyperion Essbase outlines, problems with, 10-4
- Hyperion Essbase outlines, viewing, 3-50
- members, viewing, 3-50
- metaoutline, description, xiii
- OLAP model, description, xiii
- TBC metaoutline, A-1

Sample Outline dialog box, 3-51

Save New Metaoutline dialog box, 2-24

saving

- metaoutlines, using a different name, 2-27
- metaoutlines, using File > Save, A-26
- metaoutlines, using File > Save As, 2-27
- problems with, 10-2

scaling, values, 4-28

scenarios, 1-4, A-25

Schedule Essbase Load dialog box, 7-14, 7-17, 8-9, 8-12

schedule service, starting, 7-19

Scheduled Jobs dialog box, 7-20, 7-22

scheduled loads

- problems with, 10-5
- unable to schedule, 10-5
- viewing and deleting, 7-19, 7-22

scheduling

- data loads, 8-12
- member loads, 7-17
- problems with, 10-5
- unable to complete, 10-5
- Windows NT schedule service, 7-19

scripts, calc scripts, 7-13, 8-8

searches, case-sensitive, 4-15

security

- Drill-Through reports and, 9-24
- Hyperion Essbase users, creating, 8-5

Select a Member dialog box, 3-22

Select Values From dialog box, 5-16, 5-24, 5-30

selecting

- calc scripts, 7-13, 8-8
- multiple conditions, 5-16, 5-25
- multiple objects, 2-15, 2-16

semicolons (;)

- in names in scripts and formulas, 3-10

Index

- servers, Hyperion Essbase users and, 8-5
- Services dialog box, 7-19
- Set Sort Order dialog box, 9-20
- setting
 - aliases, 6-12
 - build level, 4-25
 - calc scripts, 7-13, 8-8
 - expense property, 6-23
 - formulas, 6-27
 - Hyperion Essbase properties, 6-1
 - load member, 3-21
 - metaoutline filters, 5-8
 - OLAP intersection levels, 9-7
 - query governors, 7-5, 9-15
- Settings menu, Control Panels command, 7-19
- shared members
 - adding, 4-3
 - children and, 4-6
 - filters, problems and, 10-6
 - formulas and, 4-6
 - missing, 10-6
 - rules for, 4-6
 - user-defined attributes and, 6-25
- signs, reversing plus and minus, 4-29
- single measure, renaming, 3-47
- single quotation marks (')
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- size, query, controlling, 7-5, 9-15
- skip property, 6-21
- slashes (/)
 - consolidation property, 6-16
 - in names in scripts and formulas, 3-10
- sorting
 - Drill-Through reports and, 9-19
 - members, 4-24
 - months, chronologically, 4-16
 - performance and, 4-25
- source data, multidimensional databases, 1-5
- sources. *See* data sources
- spaces
 - converting to underscores, 4-17
 - dropping, 4-16
 - dropping and converting, order of, 4-12
 - in dimension and member names, 3-9
- sparse dimensions, defined, 6-8
- Spreadsheet Add-in drilling through from, 9-1
- spreadsheet files, data sources, 1-5
- spreadsheet users, customizing Drill-Through reports, 9-22
- SQL
 - checking, 3-51
 - data sources, 1-5
 - filters, 5-1, 5-4, 5-19
 - printing, 3-51
 - queries, controlling the size of, 7-5, 9-15
 - syntax, filter errors, 10-3
 - versions supported, 1-5
 - viewing for a dimension, 3-51
 - viewing in filter, 5-19
- SQL 89, 1-5
- SQL Drill-Through. *See* Drill-Through
- star schemas. *See* OLAP models
- starting
 - metaoutline, methods, 2-3
 - OLAP Model, programs necessary, 2-3
- status bar, defined, 2-13
- stopping, member or data load, 7-15, 8-10, 10-4
- storage
 - data properties, 6-9
 - dimensions, specifying, 6-8
- store properties, 6-10
- strategic planning, 1-4
- SUBSTRING function, 10-6
- suffixes
 - adding to fields, 4-18
 - attribute names, 3-27
 - new members and, 4-10
 - operations, order of, 4-13
- syntax
 - filters, errors, 10-3
 - formulas, incorrect, 10-2
- T**
 - tab, in names, 3-8
 - tables, capitalization, xiii
 - tagging
 - accounts dimension, 6-4
 - measure dimensions, 3-23
 - time dimension, 6-5

- TBC Metaoutline
 - dimensions in, 3-4
 - measures in, 3-6
 - parts of, 1-12
 - sample, xiii
 - TBC Model, xiii
 - TBC_MD, xiii
 - templates. *See* metaoutlines
 - text
 - dataload.txt, 8-10, 10-5, 10-7
 - files, data sources, 1-5
 - tilde, consolidation property (~), 6-16
 - time
 - format, 7-18, 8-13
 - query, controlling, 7-5, 9-15
 - time-out errors, 10-4
 - time balance
 - average, 6-20
 - first, 6-20
 - last, 6-20
 - missing properties, 6-21
 - properties, specifying, 6-20
 - skip, 6-21
 - time balance properties, 6-20
 - time dimension
 - dense dimension, 6-8
 - dimension type, 6-5
 - time-out errors, 10-4
 - timing, specifying for loads, 7-17, 8-12
 - toolbar, defined, 2-13
 - Tools menu, Scheduler command, 7-20, 7-22
 - Transformation command, 4-14
 - transformations
 - aliases and, 10-4
 - order of, 4-12
 - transforming member names, 4-12
 - transparent partitions, 6-3
 - tree editor, collapsing or expanding a member
 - branch, 2-12
 - troubleshooting
 - attribute names, 10-6
 - connecting, 2-6
 - data load and, 10-4
 - filters and, 10-3
 - member and data loads, 10-6
 - member load and, 10-4
 - metaoutlines and, 10-1
 - transformations and, 10-2
 - tuning
 - data sources, 3-51
 - viewing SQL, 3-51
 - two-pass calculations, specifying, 6-14
 - types of dimensions, 6-3
- ## U
- UDAs
 - calc scripts, 6-25
 - compared with attributes, 3-25
 - creating, 6-26
 - defined, 6-25
 - report scripts, 6-25
 - shared members and, 4-6, 6-25
 - unary operators, 6-16
 - underscores (_)
 - converting spaces to, 4-17
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
 - UNIVERSE reserved word, 3-9
 - unscheduled data loads, 8-8
 - unscheduled member loads, 7-13
 - updating
 - data. *See also* data loading
 - Hyperion Essbase outlines incrementally, 7-11, 8-6
 - members. *See also* member loading
 - user names, specifying, 7-10, 8-5
 - user privileges. *See* permissions
 - user-defined dimensions
 - creating, 3-18
 - defined, 3-7
 - example, 3-7
 - load member for, 3-21
 - valid, 2-21
 - user-defined members
 - creating, 3-18
 - defined, 3-7
 - formulas on, 6-27
 - users, Hyperion Essbase, creating, 7-10, 8-5

Index

V

- validation
 - metaoutlines, 2-19
 - problems, 10-1
- values. *See* data values
- variance reporting properties, 6-23
- variances
 - adding to Scenario dimension, A-25
 - reporting, A-21, A-23
- vertical bars (|)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-15, 4-18
- View Existing Job dialog box, 7-21
- View Filter dialog box, 5-12, 5-31
- View menu
 - Display Essbase Properties command, 2-13, 6-29
 - Metaoutline icons command, 2-14
 - Sample command, 3-50
 - Status bar command, 2-13
 - Toolbar command, 2-13
- viewing
 - columnar format, 6-29
 - condensed format, 6-29
 - displaying Hyperion Essbase properties, 6-29
 - filter SQL statements, 5-19
 - filters, 5-12
 - generated SQL, 3-51
 - Hyperion Essbase properties in columnar format, 6-30
 - Hyperion Essbase properties in condensed format, 6-31
 - measures, 3-52
 - sample dimensions, members, outlines, 3-50
 - sample dimensions, members, outlines, problems with, 10-4
 - scheduled loads, 7-19

W

- weekly, member and data load, 7-18
- what if scenarios, example, 1-4
- WHERE clauses. *See* filters
- white space, in dimension and member names, 3-9
- whitespace. *See* spaces
- Windows NT schedule service, 7-19
- Windows Start menu, using to start a metaoutline, 2-3
- wizard. *See* OLAP Metaoutline Assistant;
 - OLAP Model Assistant
- workflow
 - Hyperion Integration Server, 1-6
 - Hyperion Integration Server product family, 1-9
 - metaoutlines, creating, 1-12
- write
 - access, setting, 2-25
 - access, specifying for OLAP models, 2-25
 - locked metaoutline, 10-2
- write access. *See* read access

Y

- year filter, example, 5-5

Z

- zeros
 - property, 6-21
 - skipping during calculation, 6-21