

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	1-11
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

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
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
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	2-28
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	3-3
About Member Levels and Members	3-5
About Measures	3-6
About User-Defined Dimensions and Members	3-7
About Naming	3-9
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	3-19
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	3-24
Renaming Dimensions, Member Levels, and Measures	3-25
Renaming in the Right Frame	3-25
Renaming in the Dimension Properties or Member Properties Dialog Box	3-26
Renaming a Single Measure	3-27
Deleting Dimensions, Member Levels, and Measures	3-29
Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines	3-30
Viewing Defined Measures	3-32

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
Setting Load Properties for Member Names	4-11
Transforming Member Names	4-11
Replacing Parts of Member Names	4-13
Dropping Spaces From Around Member Names	4-16
Converting Spaces Within Member Names to Underscores	4-16
Attaching Prefixes and Suffixes to Member Names	4-17
Changing the Case of Member Names	4-21
Setting Load Properties for Members	4-22
Sorting Members	4-23
Setting the Member and Data Load Build Level	4-25
Setting Load Properties for Measures	4-26
Scaling Measures	4-27
Reversing Measure Signs	4-28
Changing the Column Associated with a Measure	4-29

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	5-11
Deleting a Metaoutline Filter	5-12
Creating a Member Filter Interactively	5-13
Typing a Member Filter	5-17
Creating a Measure Filter Interactively	5-18
Typing a Measure Filter	5-22

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	6-5
Specifying Dimension Storage Properties	6-6
Specifying Data Storage Properties for Dimensions and Members	6-8
Setting Aliases for Dimensions and Members	6-10
Specifying Two-Pass Calculation on Dimensions and Members	6-12
Specifying Member Consolidation Properties	6-14
Setting Members as Dynamic Time Series	6-16
Specifying Account Properties	6-17
Specifying Time Balance Properties	6-18
Specifying Variance Reporting Properties	6-21
Specifying UDAs	6-23
Understanding Rules for UDAs	6-23
Defining a UDA	6-24
Specifying Formulas	6-25
Viewing Hyperion Essbase Properties	6-27

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	7-16
Starting the Windows NT Schedule Service Automatically	7-17
Starting the Windows NT Schedule Service Manually	7-18
Viewing Scheduled Member and Data Loads	7-18
Deleting Scheduled Member and Data Loads	7-21

Chapter 8: Loading Data

Understanding the Metaoutline Workflow	8-2
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	8-11

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	9-10
Adding Drill-Through Columns	9-12
Setting the Size of a Query	9-14
Specifying the Format of a Report	9-15
Specifying Column Ordering	9-16
Specifying Row Ordering	9-18
Specifying Spreadsheet User Actions	9-20
Customizing a Report	9-21
Specifying the OLAP Integration Server Name	9-22

Specifying a Database Username and Password	9-23
Working with Reports	9-25
Renaming a Report in the Drill-Through Reports Dialog Box	9-25
Renaming a Report in the Report Properties Dialog Box	9-26
Deleting an Individual Report, an Intersection Level, or a Drill-Through Column	9-27
Deleting All Intersection Levels or Drill-Through Columns in a Report	9-27

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

Glossary

Index

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 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 an OLAP 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.

Glossary, a list of key Hyperion Integration Server terms and their definitions.

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

Sample Applications

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 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 Excel*, for information about how to use Hyperion Essbase Spreadsheet Add-in with Microsoft Excel.
- *Hyperion Essbase Spreadsheet Add-in User's Guide for 1-2-3*, for information about how to use Hyperion Essbase Spreadsheet Add-in with Lotus 1-2-3 for Windows.
- The online *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 i: 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 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 i: 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 1-11
- “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.

The 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 do all of the following:

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 forecast data, input new product sales scenarios (for example, What if I sell 2000 widgets to our biggest corporate customer?), update the forecast data, and provide revised figures 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.

Spreadsheet operations

Drilling down or drilling up on data 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, while 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 for Excel* or the *Hyperion Essbase Spreadsheet Add-in User’s Guide for 1-2-3*.

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.

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.

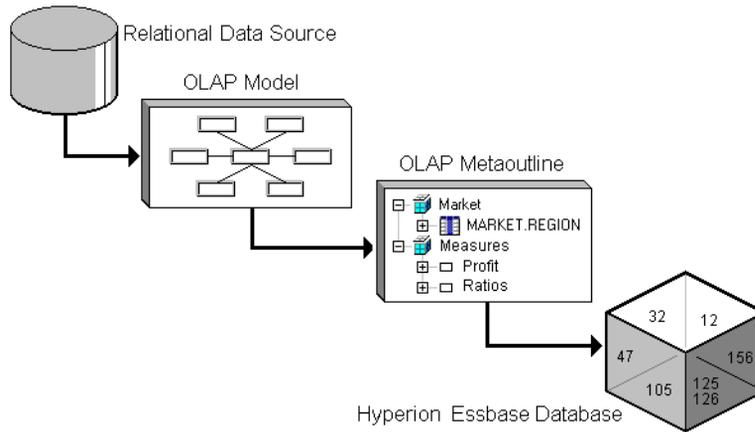


Figure 1-1: Workflow for Creating a Hyperion Essbase Database from a Relational Data Source

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.

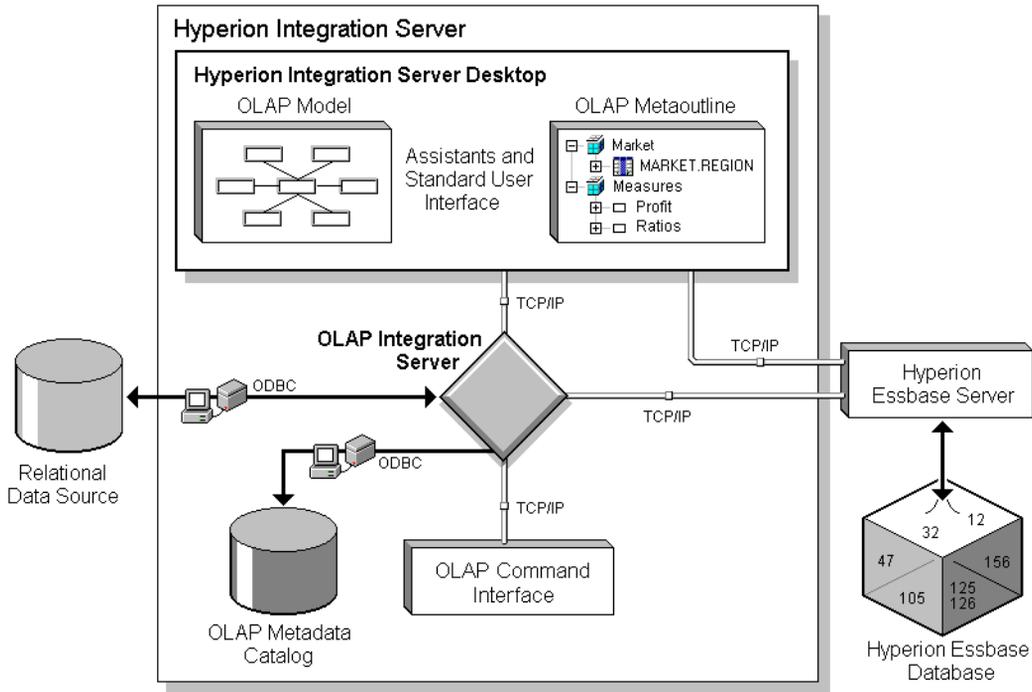


Figure 1-2: Hyperion Integration Server

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 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.

The 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 the 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:

- Are reusable. You can use a metaoutline as the basis for more than one Hyperion Essbase outline.
- Are centralized. You can define a metaoutline at a central location and use it to create multiple Hyperion Essbase outlines in multiple locations.
- Allow you to create a Hyperion Essbase outline that is based on criteria that you specify.
- Allow 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.
- Allow you to view sample Hyperion Essbase outlines before you build them.
- Automatically generate the SQL necessary to retrieve data from an external data source.
- Allow you to filter the data from the relational data source before you build the associated Hyperion Essbase outline.
- Allow 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:

- 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 Measures” on page 3-6.
- 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 Market, Year, Scenario, and Product. The metaoutline also contains a dimension named Measures that contains SALES and COGS measures. The Year dimension contains a hierarchy—Quarter and Month.

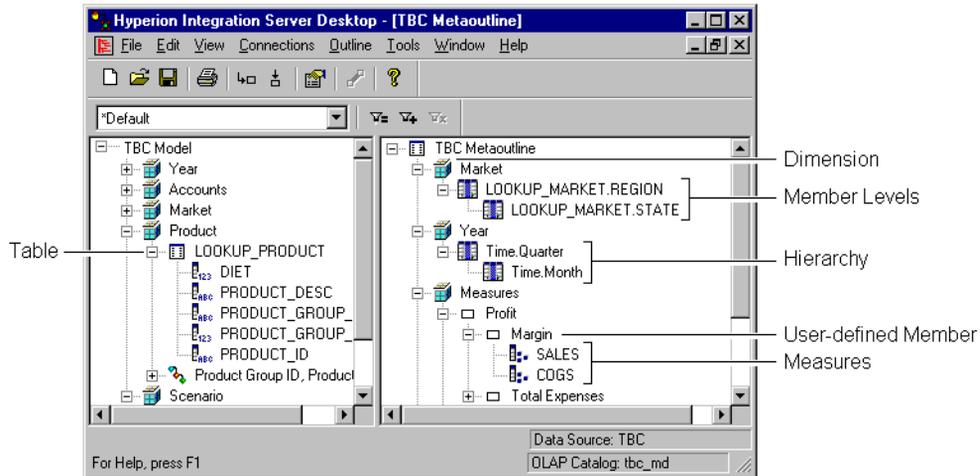


Figure 1-3: Sample Metaoutline

Workflow for Creating a Metaoutline

Before you can create a metaoutline, you must have the following 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.

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 3-30.

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 for Excel* or the *Hyperion Essbase Spreadsheet Add-in User’s Guide for 1-2-3*.

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

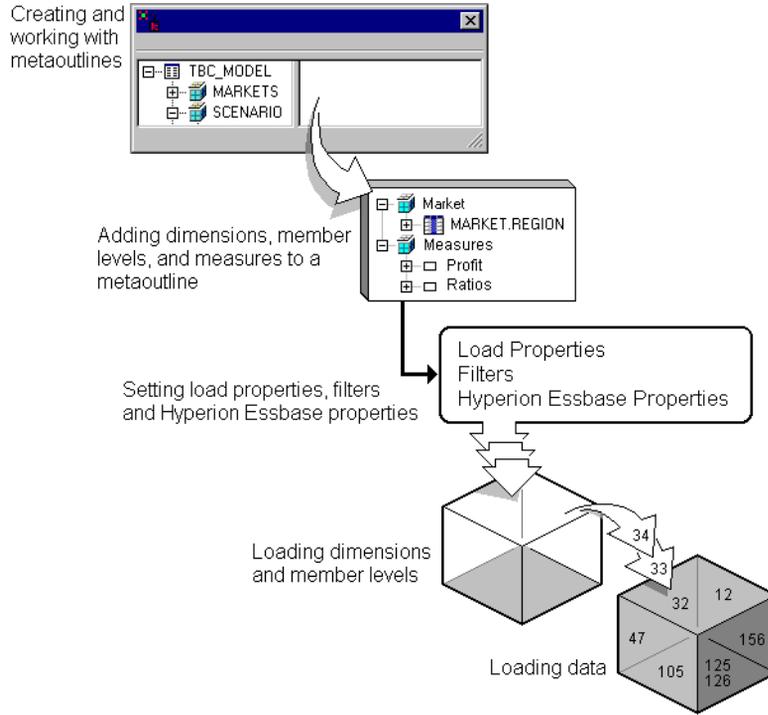


Figure 1-4: Metaoutline Workflow

When you finish, you have created the following:

- 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 2-28
- “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.

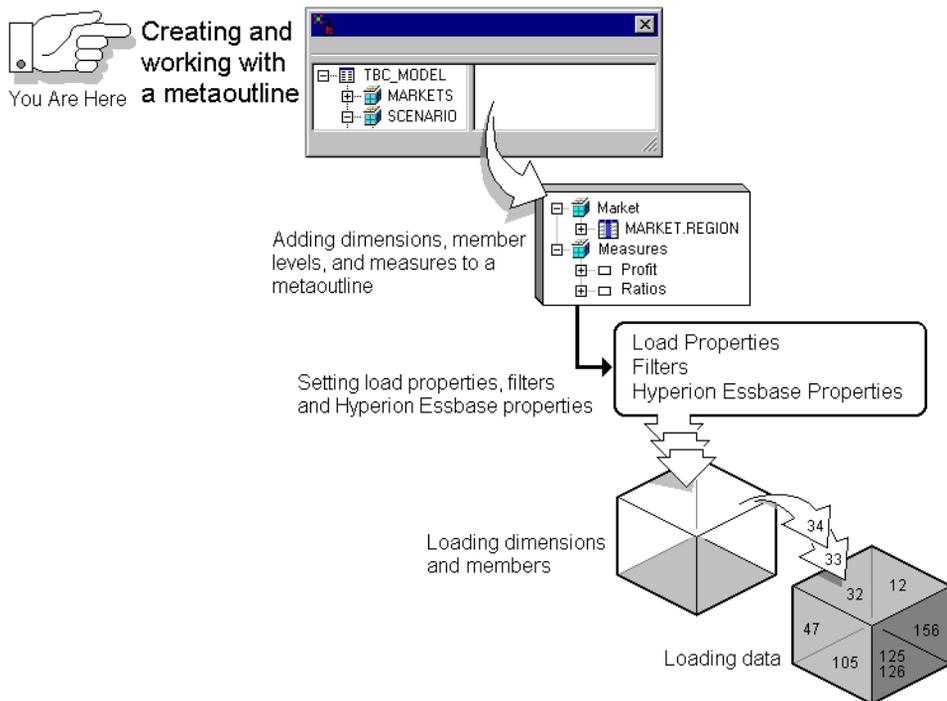


Figure 2-1: Metaoutline Workflow

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, an 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.

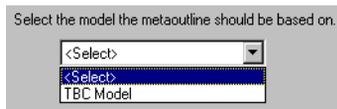
If you are already running Hyperion Integration Server Desktop and are connected to OLAP Metadata Catalog, select File > New or File > Open to open the **Welcome** dialog box.

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, 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 **OLAP Metaoutline Assistant**.
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**.

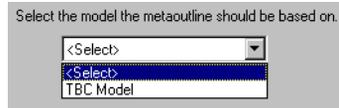
Exclusive Access enables you to modify and save the metaoutline. If **Exclusive Access** is not selectable, another user has write-lock privileges and you have read-only privileges or no privileges.

4. Click Open.

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



1. Select **OLAP Metaoutline**.
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** group 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** group 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.

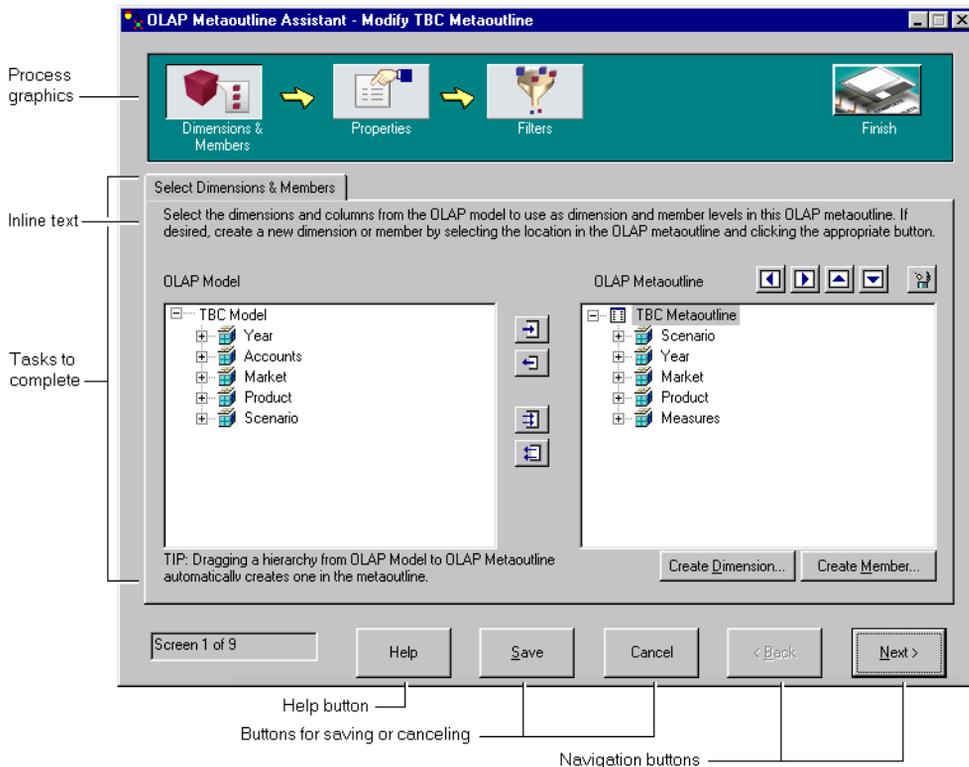


Figure 2-2: Select Dimensions & Members Tab in OLAP Metaoutline Assistant

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 buttons, 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	For more information, see...
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

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

Tab Name	For more information, see...
Set Account Properties	“Specifying Account Properties” on page 6-17
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 5-11
Preview Outline	“Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-30
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.

- 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, refer to the assistant overview information (or click Overview in the **Welcome** dialog box), 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 click the OLAP Metaoutline icon, the OLAP Metaoutline main window of 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.

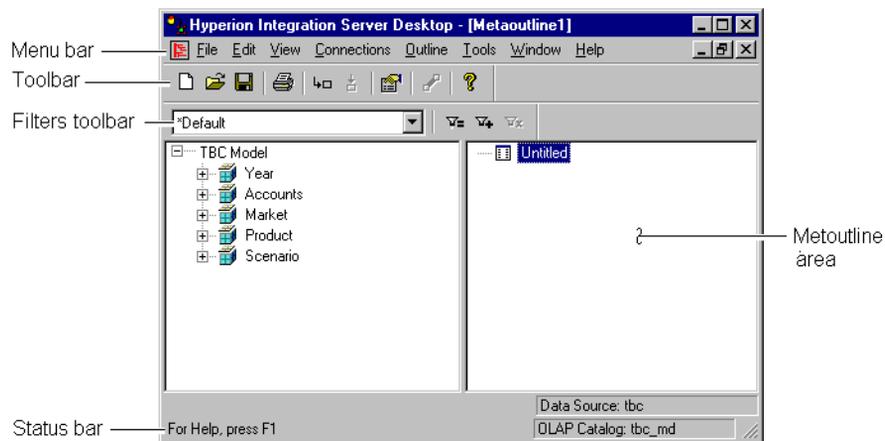


Figure 2-3: OLAP Metaoutline Main Window

Left Frame

The left frame lists the dimensions defined in the OLAP model. You can use the 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 the 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 6-27.

You can also 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**.

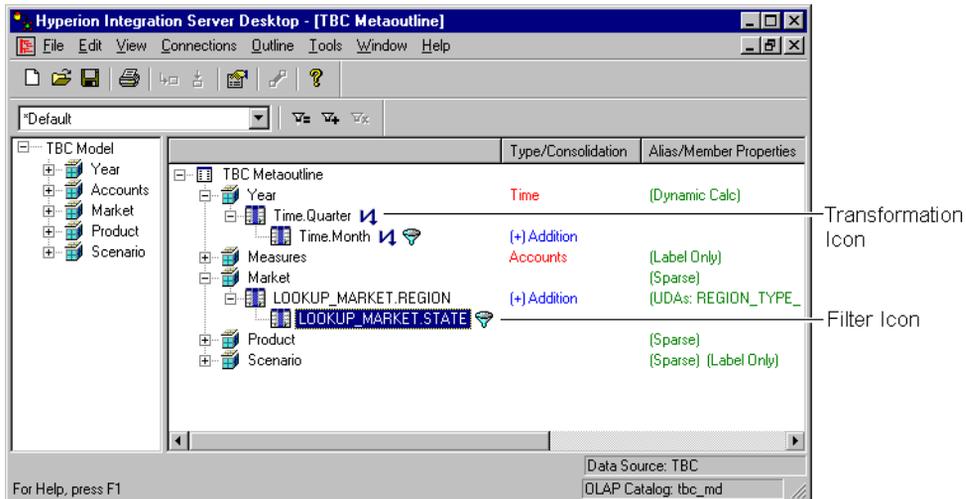


Figure 2-4: Viewing Hyperion Essbase Properties in Columnar Format

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.

As shown in Figure 2-5, if you drag the ADDITIONS, COGS, CITY, and REGION members into the right frame, the Accounts and Market dimensions are created automatically.

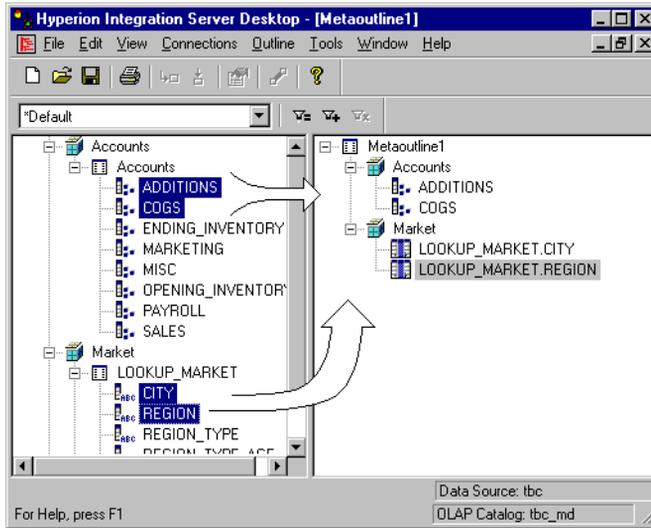


Figure 2-5: Selecting and Dragging Multiple Members into the Right Frame

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

Selecting two or more...	Activates these commands on the Edit or pop-up menu...
Dimensions	Delete command Properties command Drill-Through command
Member levels	Delete command Properties command Drill-Through command
Some combination of dimensions, member levels, and measures	Delete command Drill-Through command
Measures within the accounts dimension, including the accounts dimension	Delete command Drill-Through command
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 and Tab and Command Availability

When you select two or more...	Tabs that are available on...	Commands that are not selectable
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
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 and Tab and Command Availability (Continued)

When you select two or more...	Tabs that are available on...	Commands that are not selectable
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
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 manually validate a metaoutline 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:

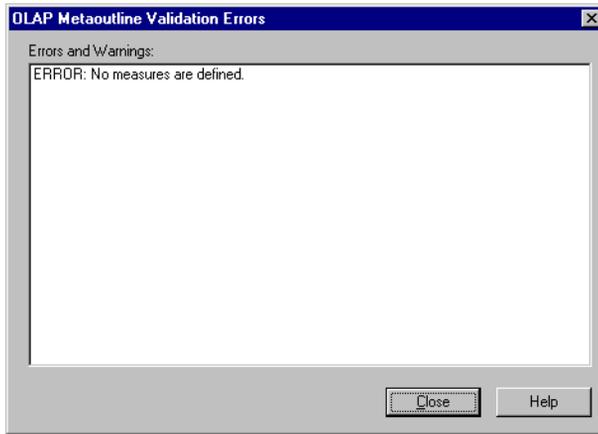


Figure 2-6: Invalid OLAP Metaoutline Validation Error Message Listing

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 3-24.
- 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** on the **Welcome** screen. Opening an existing metaoutline with **Exclusive Access** guarantees that no other user can modify the metaoutline.

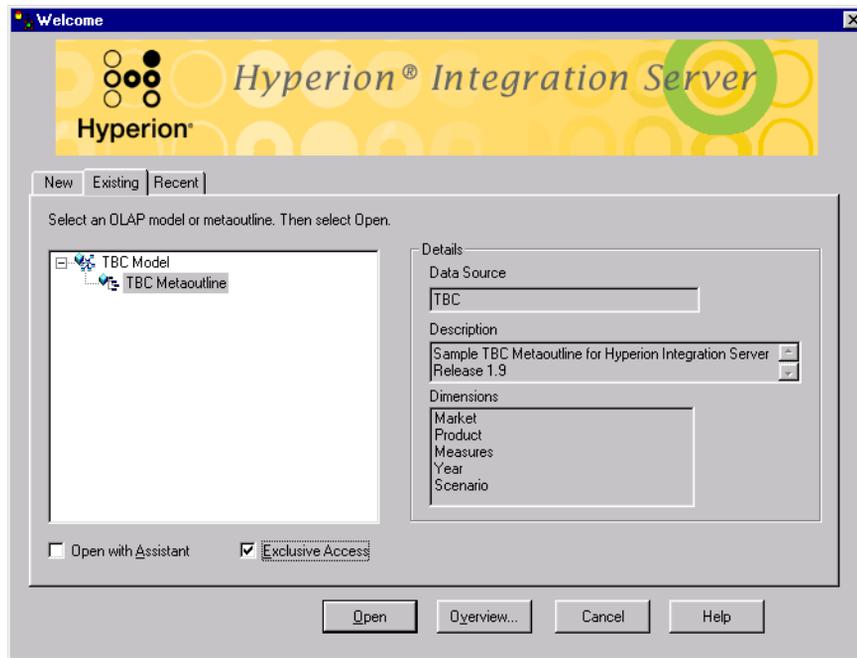


Figure 2-7: Selecting the Exclusive Access Option

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

- 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, use 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.

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

- “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

Note: You cannot save a 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. Do one of the following:
 - 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.

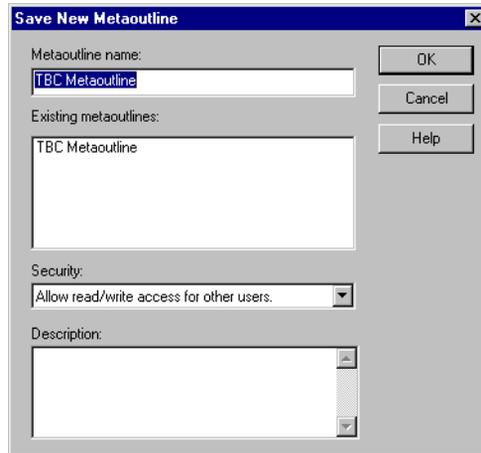


Figure 2-8: Saving a Metaoutline

2. In the **Metaoutline name** text box, enter the name of the metaoutline. See “About Naming” on page 3-9.

3. To determine who can access the metaoutline, from the **Security** drop-down list box, select one of the following options:

Table 2-4: Read/Write Settings

Select...	To give other users permission to...
Allow read/write access for other users	Both read and write to the metaoutline. This setting is the default.
Allow read access for other users	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	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 use 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 using 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:

- 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 an 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 3-9.

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.

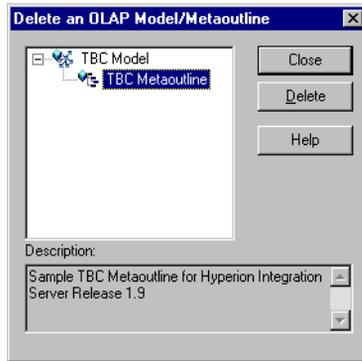


Figure 2-9: Deleting a Metaoutline

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.

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 the Hyperion Essbase outline.

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
- “Renaming Dimensions, Member Levels, and Measures” on page 3-25
- “Deleting Dimensions, Member Levels, and Measures” on page 3-29
- “Viewing Sample Dimensions, Members, and Hyperion Essbase Outlines” on page 3-30
- “Viewing Defined Measures” on page 3-32

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.”

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

This chapter focuses on adding dimensions, member levels, and measures to a metaoutline.

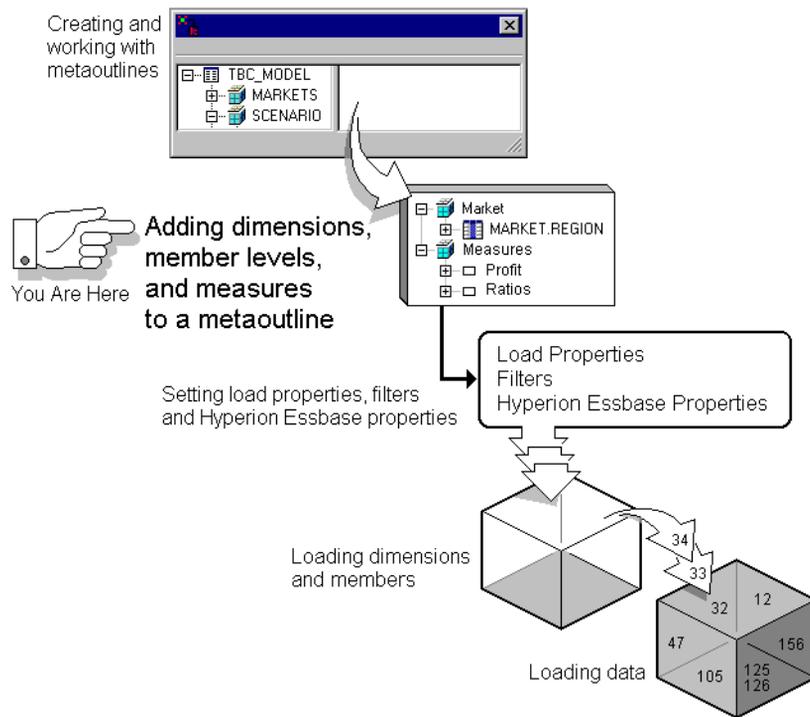


Figure 3-1: Metaoutline Workflow

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 3-3
- “About Member Levels and Members” on page 3-5
- “About Measures” on page 3-6
- “About User-Defined Dimensions and Members” on page 3-7
- “About Naming” on page 3-9

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 exactly 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, Product, Market, and Scenario.

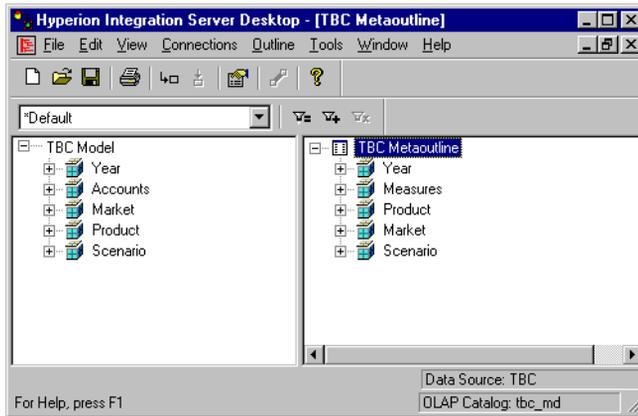


Figure 3-2: Dimensions in TBC Metaoutline

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, STATE, and CITY columns of the OLAP model become the REGION, STATE, and CITY member levels of the metaoutline. The members of the member levels REGION, STATE, and CITY 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 MARKET.REGION, MARKET.STATE, and MARKET.CITY.

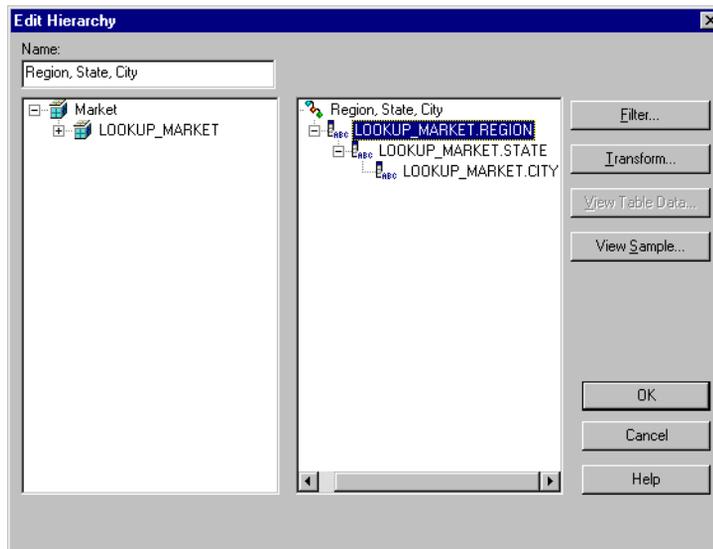


Figure 3-3: Example Hierarchy in an OLAP Model

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:

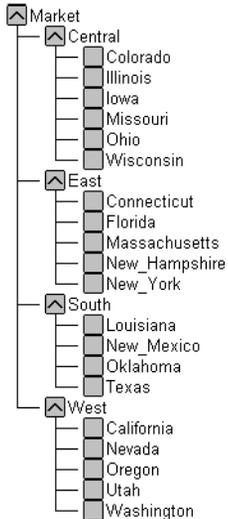


Figure 3-4: Example Hyperion Essbase Outline

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 exactly to dimensions and members in a related Hyperion Essbase outline. For example, if you have a user-defined ACTUAL member in a metaoutline, you have an Actual 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 Scenario dimension. The Scenario dimension contains both actual data, to track measures such as revenue, and forecast data, to project values for revenue and profit. The relational data source contains the actual data. A worksheet created by the budgeting department contains the forecast data.

So how do you combine the actual and forecast data in one dimension in the Hyperion Essbase database? Because there is no ACTUAL or FORECAST member in the OLAP model, you must create Actual and Forecast as user-defined members.

- To create Actual and Forecast as user-defined members:
 1. In the metaoutline, create a Scenario dimension.
See “Creating Dimensions Associated with an OLAP Model” on page 3-12.
 2. To the Scenario dimension, add the Actual and Forecast user-defined members.
See “Creating User-Defined Dimensions and Members” on page 3-19.

3. To the Actual member, add the 1996 and 1997 members from the OLAP model.

See “Creating Dimensions Associated with an OLAP Model” on page 3-12.

When Hyperion Integration Server builds the Hyperion Essbase outline, it creates the Scenario dimension, as shown in Figure 3-5:

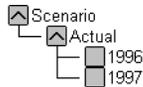


Figure 3-5: User-Defined Scenario Dimension

4. Because Actual receives its data from the data source, tag ACTUAL as the default load member.

See “Defining the Load Member for a User-Defined Dimension” on page 3-21.

5. Load members and data from the forecasting worksheet into the Forecast member of the Hyperion Essbase outline.

Thus, the Forecast member and the 1998 member are added to the Scenario dimension in the Hyperion Essbase outline.

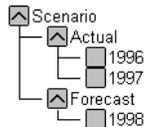


Figure 3-6: Final Hyperion Essbase Outline

See the *Hyperion Essbase Database Administrator’s Guide* for more information about loading members and data from worksheets.

About Naming

When naming metaoutlines, dimensions, member levels, aliases, UDAs, and metaoutline filters in the metaoutline, follow these rules:

- Do not use more than 80 characters.
- Do 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, UDAs, 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)	()	(parenthesis)
\	(backslash)	.	(period)
{ }	(brace)	+	(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
 - Calc 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, member level (unless the member is shared), alias, or combination alias; a generation name; or a level name

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 Database Administrator's Guide* for more information.

When you use a member level name in a calc 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)
[] (bracket)	() (parenthesis)
: (colon)	% (percent sign)
, (comma)	+ (plus sign)
- (dash, hyphen, or minus sign)	; (semicolon)
= (equal sign)	/ (slash)
\ (backslash)	
- The name starts with an ampersand (&).

In calc scripts and formulas, you must enclose the following member level names 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 3-7, 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.

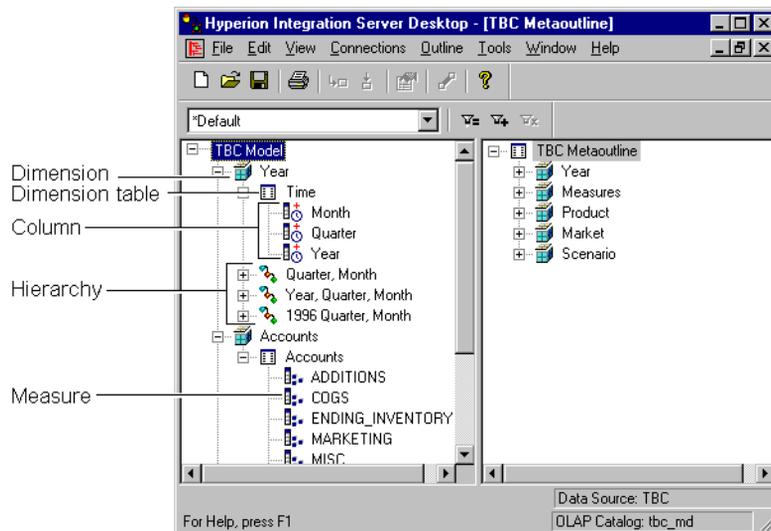


Figure 3-7: Objects in the OLAP Model

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 3-19
- “Defining the Load Member for a User-Defined Dimension” on page 3-21
- “Creating a Measure Dimension” on page 3-24
- “Creating a Measure Dimension” on page 3-24

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.

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

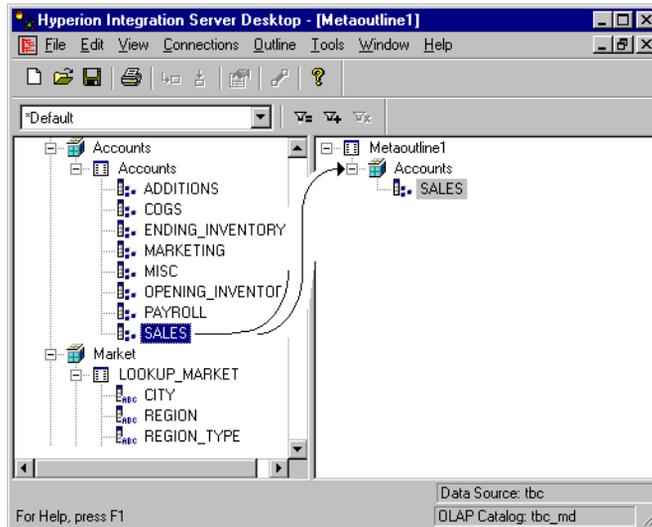


Figure 3-8: Using a Measure to Create a Dimension

If you drag a...	The selected object creates a...	For 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.
Text column, 	New dimension that contains a member level that is based on the text column.	The PRODUCT_ID column of the left frame creates in the right frame the Product dimension with the PRODUCT_ID member level.

If you drag a...	The selected object creates a...	For example...
Numeric column, 	New dimension that contains a member level that is based on the numeric column.	The STATE_TYPE_ID column of the left frame creates in the right frame the Market dimension with the STATE_TYPE_ID 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.
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 3-25.

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 3-24.

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 1996 Year, Quarter, Month hierarchy into the right frame, the hierarchy creates a Year dimension with the hierarchy inserted under it, as illustrated in Figure 3-9.

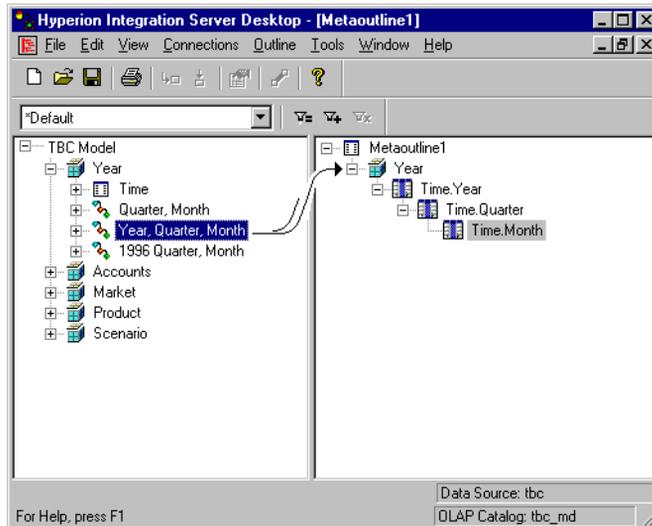


Figure 3-9: Using a Hierarchy to Create a Dimension

If you drag a column, hierarchy, or measure onto a...	The selected object creates a...	For example...
Blank space	<p>New dimension and inserts the member level, the hierarchy of member levels, or the measure 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 an Accounts dimension with an ADDITIONS member level. You can rename the Accounts dimension to Measures.</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 PRODUCT_ID column onto the Product dimension creates a PRODUCT_ID 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 DIET column onto the PRODUCT_ID member level creates a DIET member level under the PRODUCT_ID 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, do one of the following:

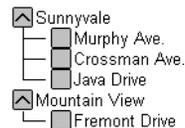
- Drag the newly-created member level or measure to the appropriate location in the dimension.
- Rename the newly-created member level or measure.
See “Renaming Dimensions, Member Levels, and Measures” on page 3-25.
- Delete unwanted member levels of a hierarchy, from the leaf level up, by selecting 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 *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 the 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:
 - To create a dimension, select the metaoutline name, for example, TBC Metaoutline,  TBC Metaoutline.
 - To create a member, select a dimension name, for example, Scenario,  Scenario.

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.

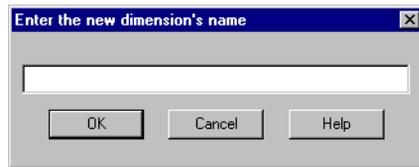


Figure 3-10: Naming a New Dimension

2. Enter the new dimension or member name, for example, Scenario.
See “About Naming” on page 3-9.
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, do one of the following:
 - 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, Profit,  Profit.

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.

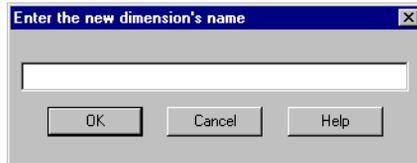


Figure 3-11: Naming a New Dimension

3. Enter the new dimension or member name, for example, Scenario.
See “About Naming” on page 3-9.
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 for OLAP Integration Server to load data into. 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.

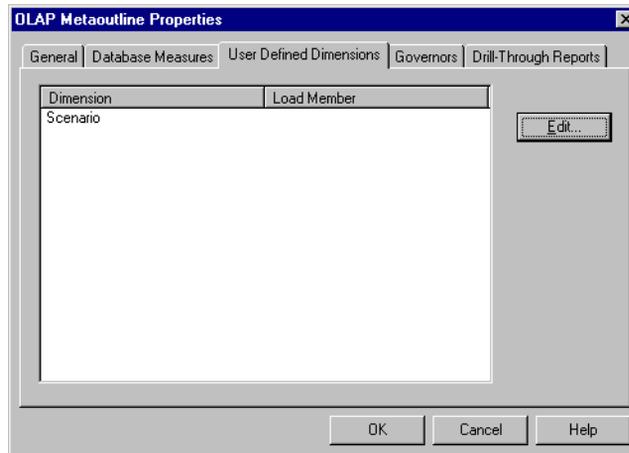


Figure 3-12: Viewing User-Defined Dimensions

4. Select the user-defined dimension for which you want to define the default load member, for example, Scenario.

5. Click Edit.

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

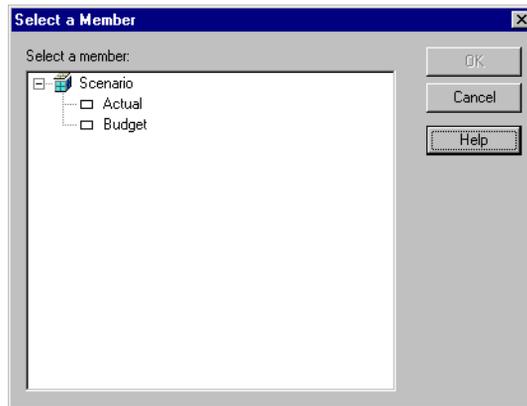


Figure 3-13: Setting the Default Load Member

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

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 3-19.

- 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.

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 3-25
- “Renaming in the Dimension Properties or Member Properties Dialog Box” on page 3-26
- “Renaming a Single Measure” on page 3-27

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 left-click the dimension, member level, or measure; for example,
 
 3. Enter the new name.
See “About Naming” on page 3-9.
 4. Do one of the following:
 - 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.

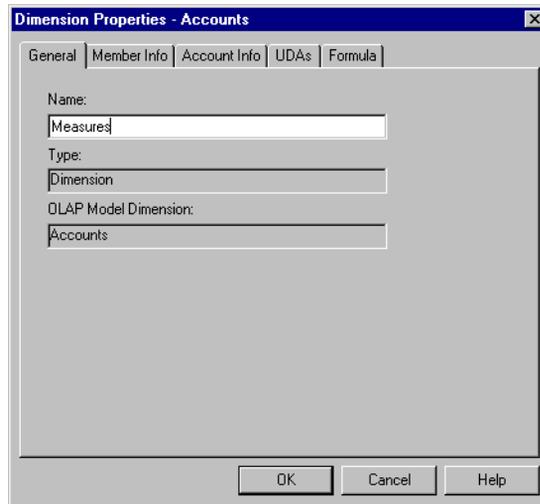


Figure 3-14: Renaming a Dimension or Member Level

3. In the **Name** text box, enter the new name.
See “About Naming” on page 3-9.
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.

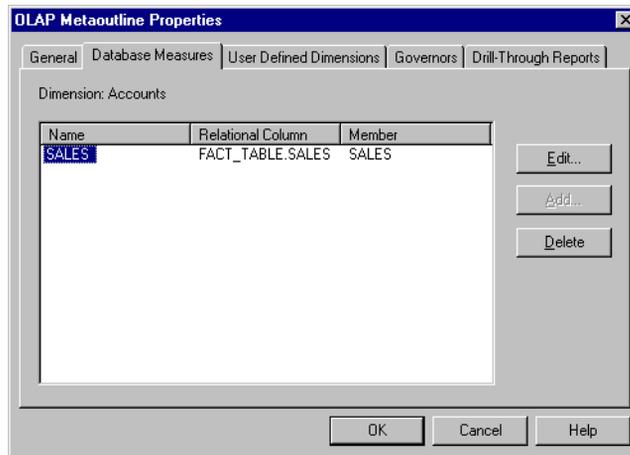


Figure 3-15: Renaming a Single Measure

4. Select a measure and click Edit.

The **Measure Properties** dialog box is displayed.

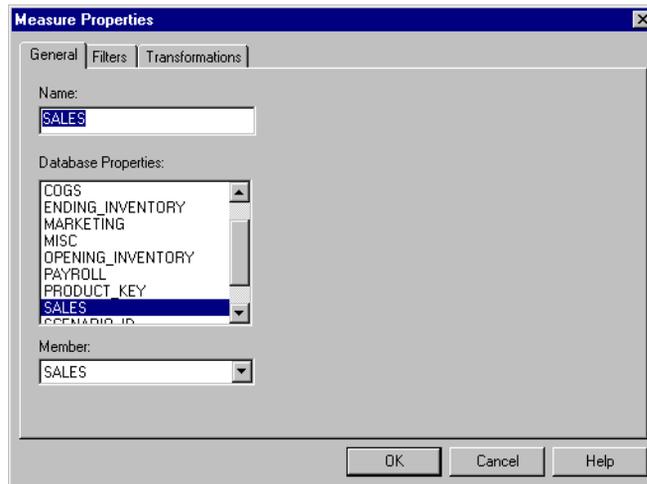


Figure 3-16: Using the Measure Properties Dialog Box

5. In the **Name** text box, enter the new name.
See “About Naming” on page 3-9.
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 the TBC Metaoutline, you delete all members of the Year dimension, that is, Time.Quarter and Time.Month.

CAUTION: You cannot undo a delete operation.

To delete a dimension, member level, or measure from a metaoutline, use one of the following techniques:

- Select the dimension, member level, or measure and press the Delete key.
- Right-click the dimension, member level, or measure and select Delete from the pop-up menu.
- 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, **Preview Filter Results** dialog box, or the **Preview Transformation Results** dialog box.

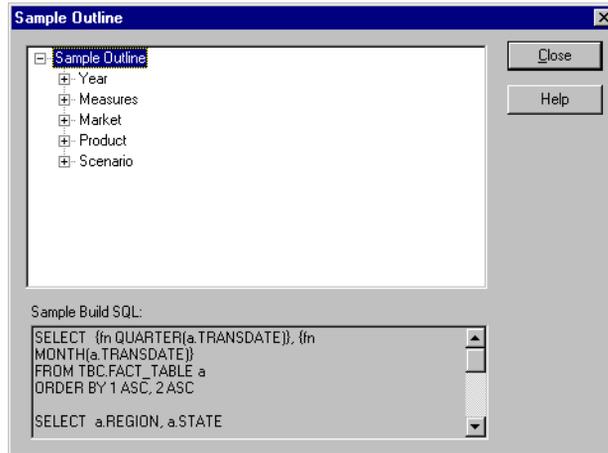


Figure 3-17: Viewing a Sample Hyperion Essbase Outline

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 generated to create the sample Hyperion Essbase outline. Use the sample SQL to do any of the following:

- 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 may want to 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.
 3. Select the **Database Measures** tab.

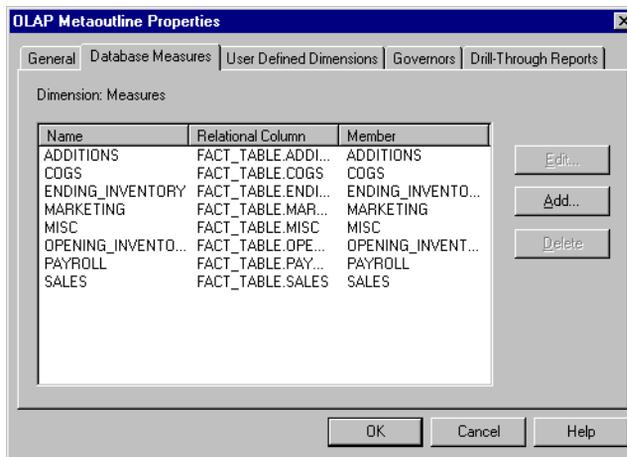


Figure 3-18: List of Measures in the Metaoutline

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:

- The name of the measure. See “Renaming a Single Measure” on page 3-27.
- 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 4-29.
- Transformations to perform on the measure. See “Scaling Measures” on page 4-27 and “Reversing Measure Signs” on page 4-28.
- Filters defined on the measure. See “Creating a Measure Filter Interactively” on page 5-18.

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 4-22
- “Setting Load Properties for Measures” on page 4-26

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.

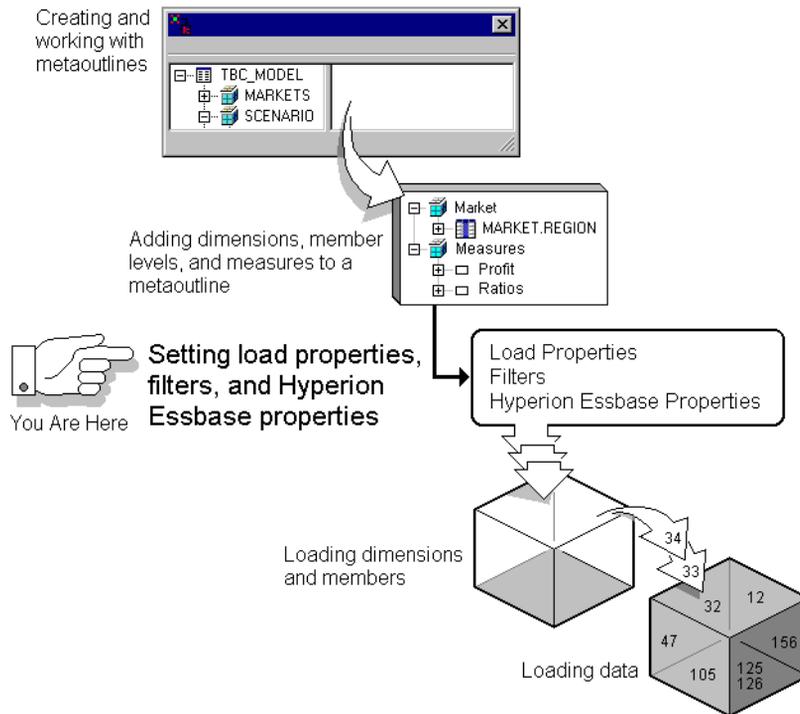


Figure 4-1: Metaoutline Workflow

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 `PRODUCT.PRODUCT_ID` column twice, once under `PRODUCT_GROUP_ID` and once under `Diet`.

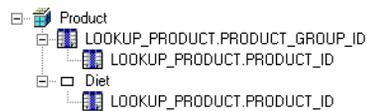


Figure 4-2: PRODUCT.PRODUCT_ID Column

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

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

- 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 100-20 member (under Diet) is stored in the first 100-20 member (under Colas).

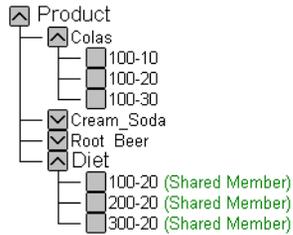


Figure 4-3: Shared Members

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

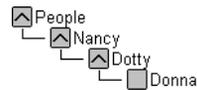
For example, suppose that you have the following manager-employee organizational chart under the dimension People:

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

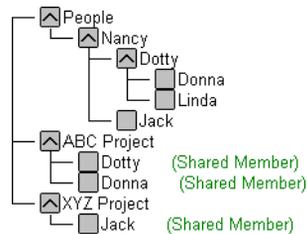
The hierarchy created by loading the first record is as follows:



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

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

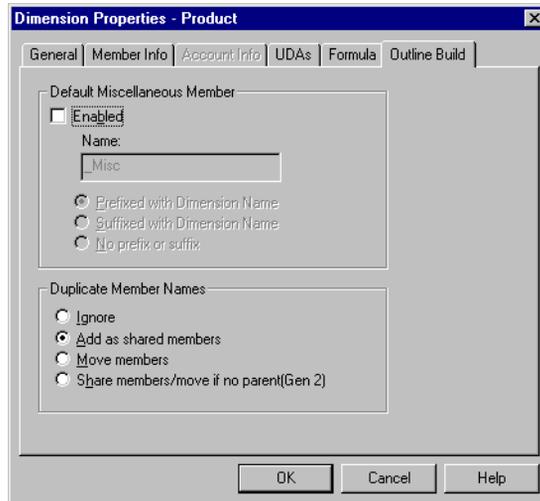


Figure 4-4: Handling Duplicate Member Names

4. From the **Duplicate Member Names** group, do one of the following:
- Select **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.
 - Select **Add as shared members** to create the duplicate member names as shared members in the Hyperion Essbase outline.
 - Select **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 6-8.

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 following:



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 following:



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

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

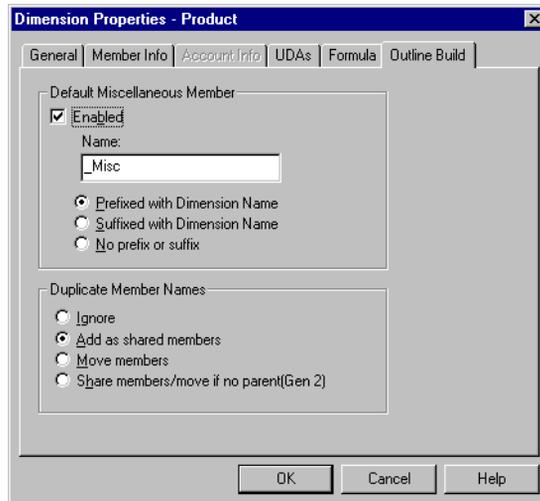


Figure 4-7: Setting the Miscellaneous Member

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 3-9.

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 `Misc` member looks like the following:



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

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 4-11
- “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 4-16
- “Attaching Prefixes and Suffixes to Member Names” on page 4-17
- “Changing the Case of Member Names” on page 4-21

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.

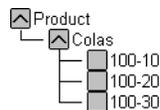


Figure 4-9: Product Dimension

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

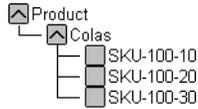


Figure 4-10: Product Members with Prefixes

Note: If you sort members, they are sorted before OLAP Integration Server applies transformations to member names. See “Sorting Members” on page 4-23.

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

1. Drops spaces from around a member name.
See “Dropping Spaces From Around Member Names” on page 4-16.
2. Converts spaces to underscores.
See “Converting Spaces Within Member Names to Underscores” on page 4-16.
3. Changes case.
See “Changing the Case of Member Names” on page 4-21.
4. 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 OLAP Metaoutline Transformations dialog box.
See “Replacing Parts of Member Names” on page 4-13.
5. Applies prefix operations.
See “Attaching Prefixes and Suffixes to Member Names” on page 4-17.
6. Applies prefix separator operations.
7. Applies suffix operations.
8. 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, MARKET.STATE.
 2. Right-click the member level and, from the pop-up menu, select Transformation to open the **Edit OLAP Metaoutline Transformations** dialog box.

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

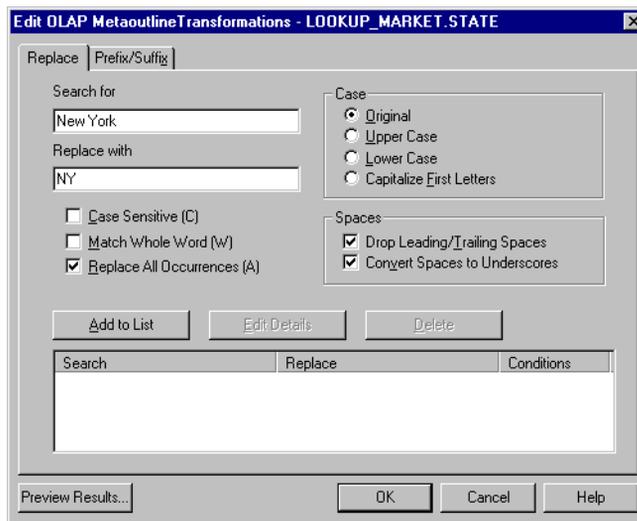


Figure 4-11: Replacing Text Strings

3. In the **Search for** text box, enter the text string that you want to replace; for example, **New York**.

4. In the **Replace with** text box, enter the text with which to replace the text string; for example, **NY**.

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)	()	(parenthesis)
\	(backslash)	.	(period)
{ }	(brace)	+	(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 3-30.
9. Click OK.

You can use the search and replace operation of the **Edit OLAP Metaoutline Transformation** 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 OLAP Metaoutline Transformation** 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 4-23.
 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:
 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 OLAP Metaoutline 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 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 OLAP Metaoutline 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)
{ }	(brace)	+	(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 OLAP Metaoutline Transformations** dialog box.

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

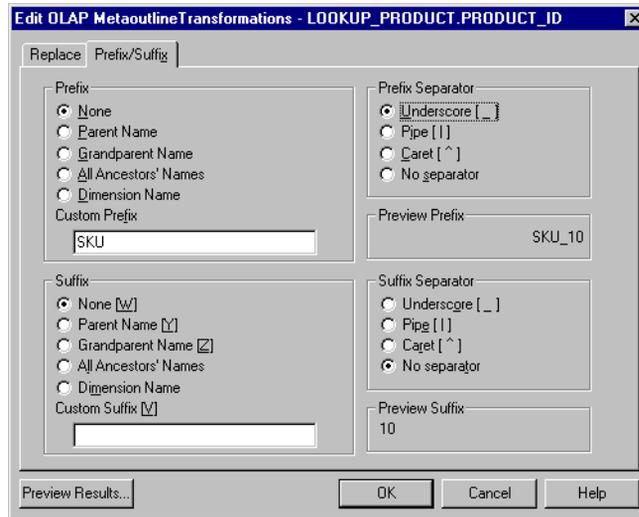
3. Select the **Prefix/Suffix** tab.

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 `PRODUCT_ID` of the Product dimension, as shown in the following sample TBC Metaoutline:

```

Product
├── LOOKUP_PRODUCT.PRODUCT_GROUP_DESC
│   ├── LOOKUP_PRODUCT.PRODUCT_GROUP_ID
│   └── ...
└── LOOKUP_PRODUCT.PRODUCT_ID

```

- **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 3-30.
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, Month.

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

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

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

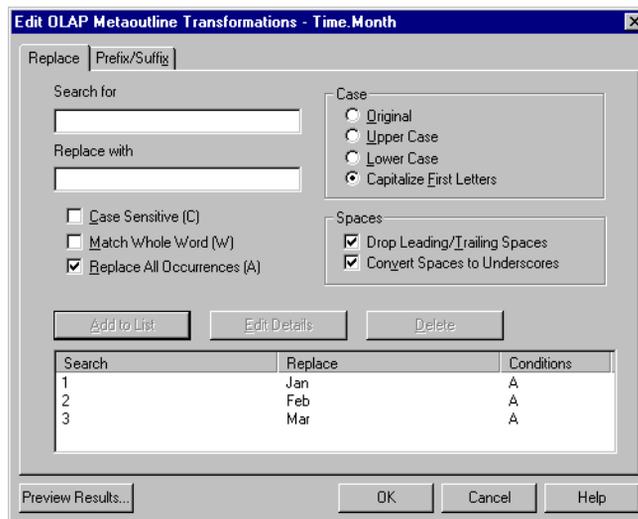


Figure 4-13: Capitalizing Member Names

3. From the **Case** group, select the case to change the member name to:
 - **Original** does not change the case.
 - **Upper Case** makes all letters of the member name uppercase, for example, january to JANUARY or January to JANUARY.
 - **Lower Case** makes all letters of the member name lowercase, for example, JANUARY to january or January to january.
 - **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, qtr1_january or QTR1_JANUARY to Qtr1_January.
4. 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 3-30.
5. 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 4-23
- “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.REGION.
 2. Right-click the member level and, from the pop-up menu, select Filter to open the **Member Load Filtering** dialog box.

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

3. Select the **Ordering** tab.

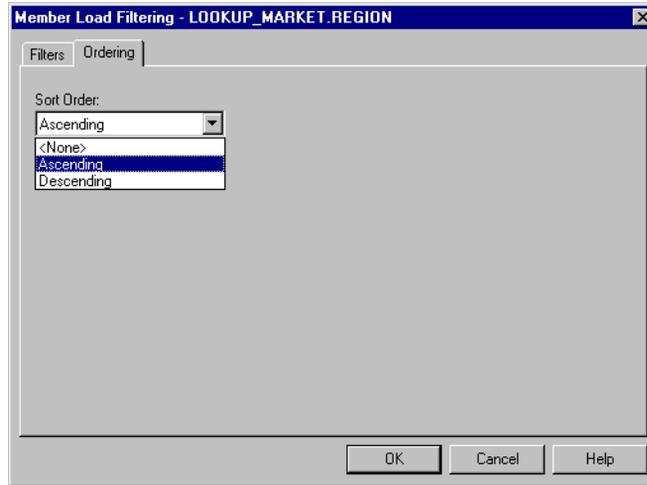


Figure 4-14: Setting Member Ordering

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.

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 4-15.

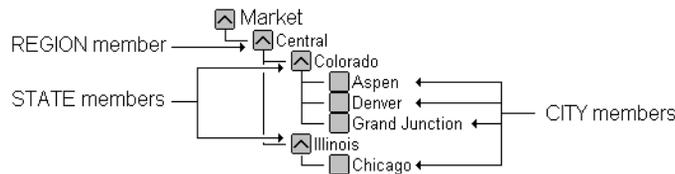


Figure 4-15: Members of the Market Dimension

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 4-16.

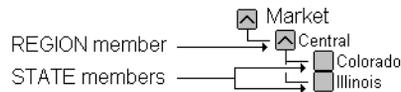


Figure 4-16: Reduced Number of Levels

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, STATE.
 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).



Figure 4-17: 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 that a measure is associated with. This topic contains the following subtopics:

- “Scaling Measures” on page 4-27
- “Reversing Measure Signs” on page 4-28
- “Changing the Column Associated with a Measure” on page 4-29

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.

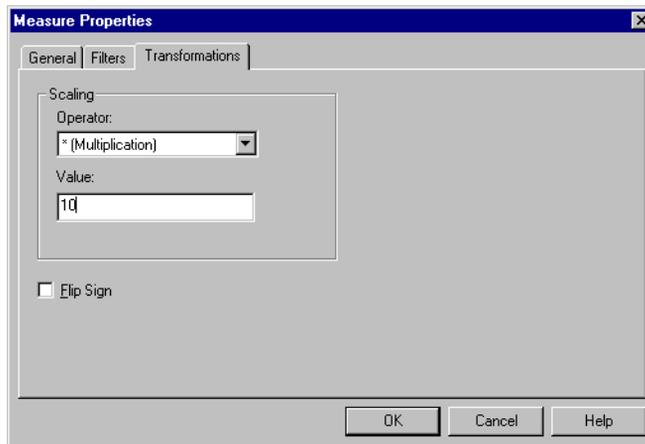


Figure 4-18: Scaling Measures

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, select 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.

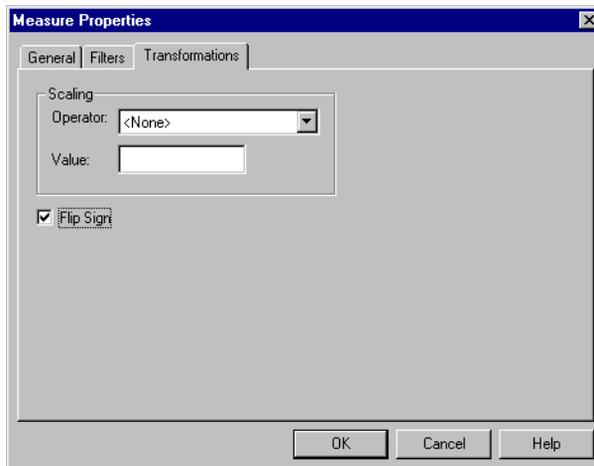


Figure 4-19: Reversing Signs for Measures

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.

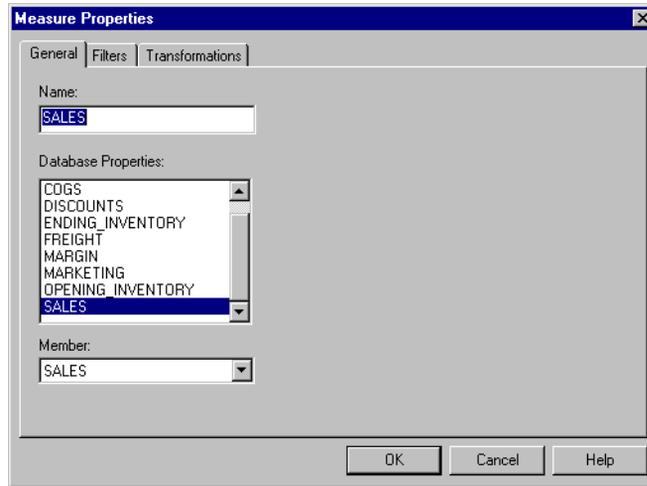


Figure 4-20: Changing the Column Associated with a Measure

4. From the **Database Properties** list, select the desired column.
5. Click OK.

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 5-13
- “Typing a Member Filter” on page 5-17
- “Creating a Measure Filter Interactively” on page 5-18
- “Typing a Measure Filter” on page 5-22

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.

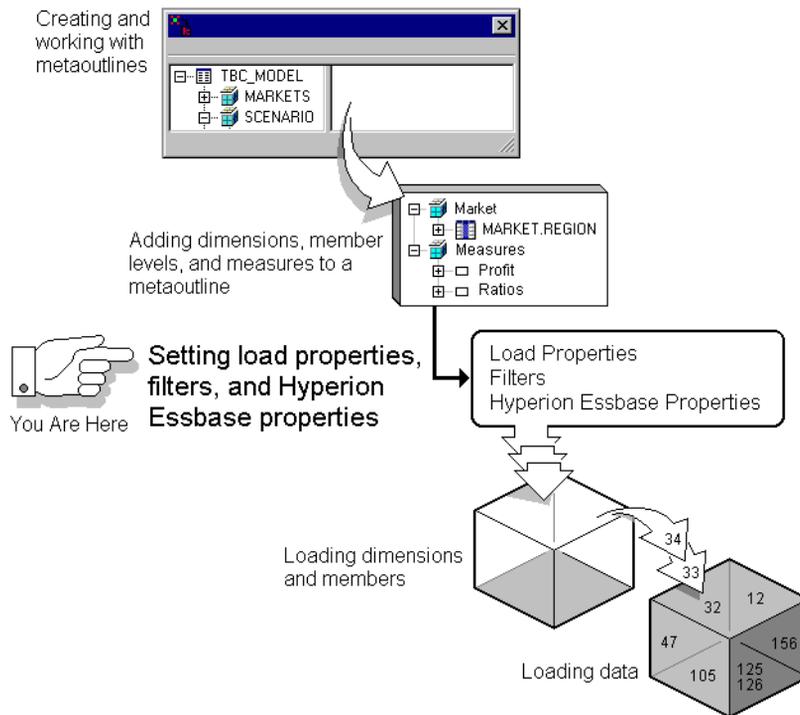


Figure 5-1: Metaoutline Workflow

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 an OLAP model contains data for 1996 and 1997.

If you want to create a Hyperion Essbase outline that tracks the 1996 data, you create a filter that loads only the members related to 1996.

Any filter that you create becomes part of an automatically generated SQL WHERE clause. For example, if you create the 1996 filter (described in the previous paragraph) on TBC Metaoutline, the filter produces the following WHERE clause:

```
SELECT DISTINCT a.YEAR
FROM hyperion.TIMESEASONS a
WHERE (a.YEAR IN ('1996'))
```

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*.

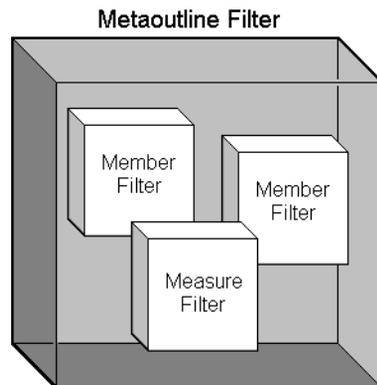


Figure 5-2: Contents of a Metaoutline Filter

About Metaoutline Filters

OLAP Integration Server automatically stores all member level and all measure filters in a metaoutline filter named *Default. You can, however, create one or more additional metaoutline filters and define a specific set of member level and measure filters to 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 the data values that are specified by the metaoutline filter. For example, you can create a metaoutline filter for each year—a 96 DATALOAD filter and a 97 DATALOAD filter. Then you can select the 96 DATALOAD filter and build a Hyperion Essbase database that tracks only information for 1996, as illustrated in Figure 5-3. Likewise, you can select the 97 DATALOAD filter and build a Hyperion Essbase database that tracks only information for 1997.

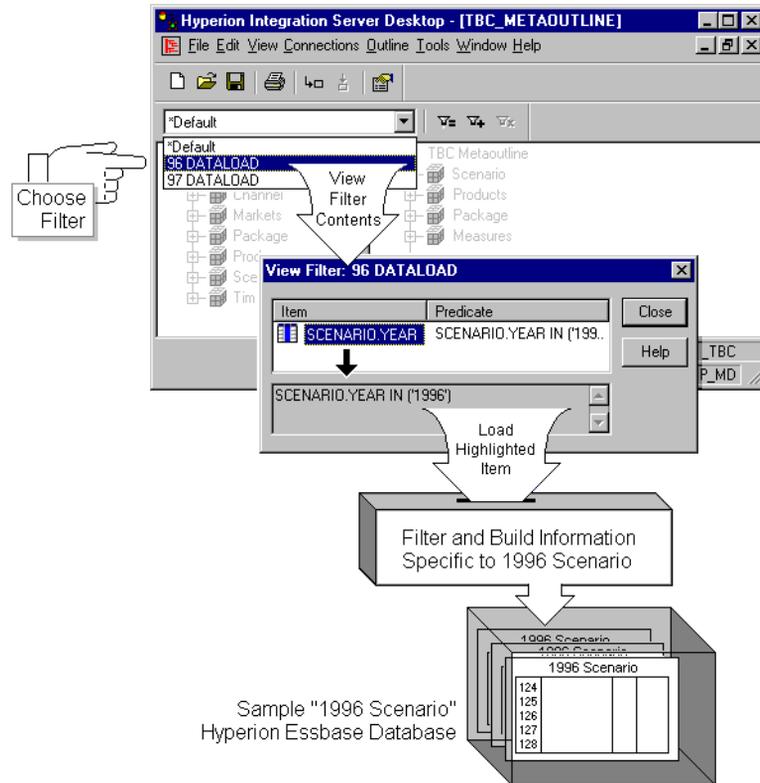


Figure 5-3: Metaoutline Filter Example

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.

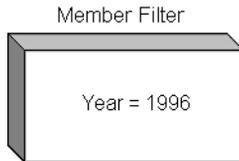


Figure 5-4: Member Filter Example

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 1996, OLAP Integration Server loads the 1996 values but not the values from other years, such as 1997.

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.

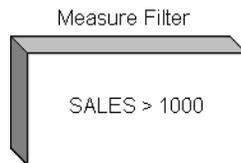


Figure 5-5: Measure Filter Example

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 5-11
- “Deleting a Metaoutline Filter” on page 5-12

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 metaoutline filter 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.

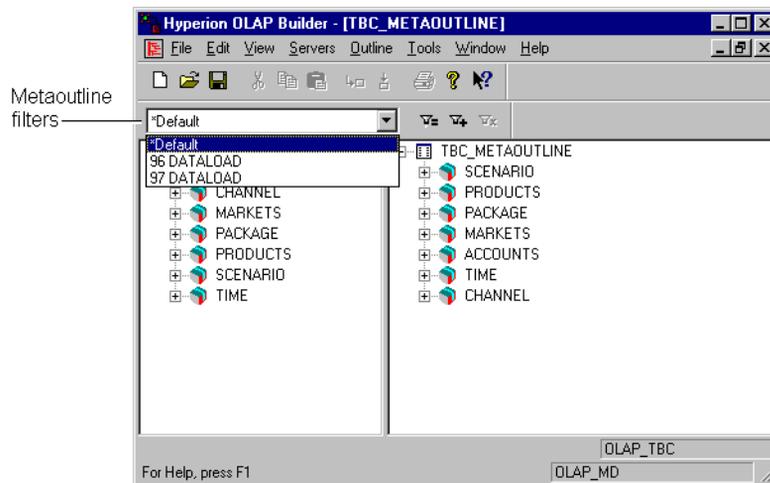


Figure 5-6: Setting a Metaoutline Filter

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.

When you create a new metaoutline filter, it inherits the member and measure filters that are defined in the *Default metaoutline filter.

► 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, **96 Dataload**.

See “About Naming” on page 3-9.

4. Click OK.

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

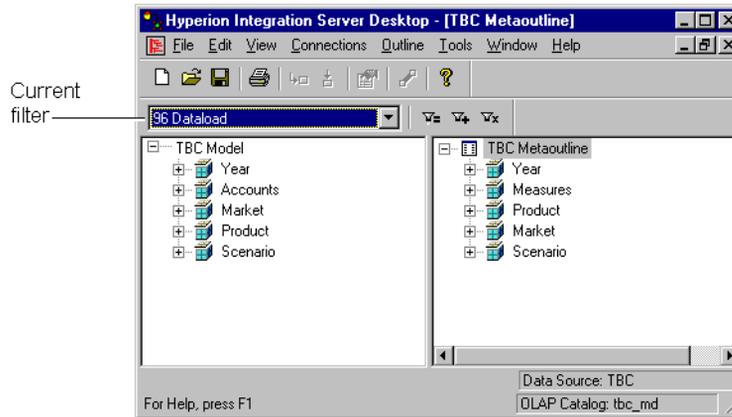


Figure 5-8: Viewing the Current Filter Name

5. In the metaoutline, create member and measure filters.

See “Creating a Member Filter Interactively” on page 5-13 and “Creating a Measure Filter Interactively” on page 5-18.

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 current metaoutline filter:

1. Open the metaoutline.

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

2. 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.

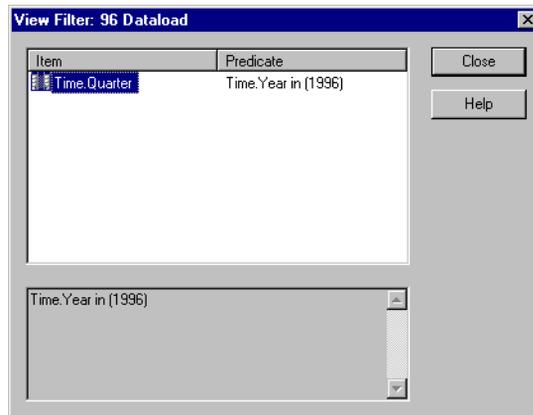


Figure 5-9: Viewing the Contents of the Current Metaoutline Filter

3. To view the conditions defined for a specific filter, select the filter, for example, Time.Quarter.
The filter conditions are displayed in the bottom box.
4. 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 5-17.

- To create a member filter interactively:
 1. In the metaoutline, select the member level to attach to the filter, for example, MARKET.STATE.
 2. To create a row filter, right-click the member level and, from the pop-up menu, select Filter to display the **Member Load Filtering** dialog box.
 3. Select the **Filters** tab.

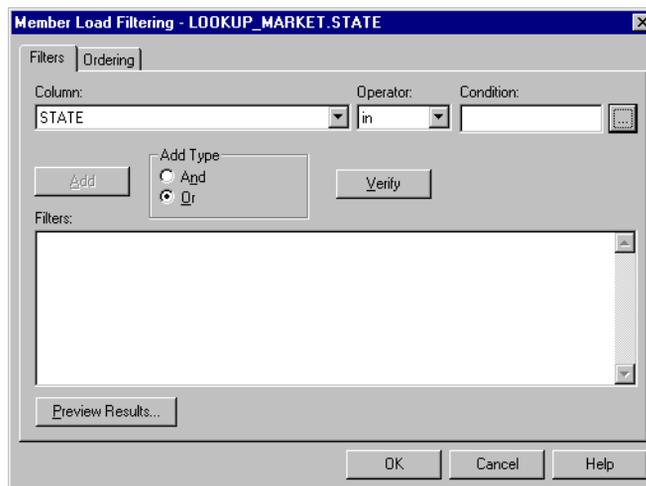


Figure 5-10: Setting a Member Load Filter Interactively

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

Table 5-1: Filter Operators

Operator	Description
in	Setting one or more condition. 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	Looking 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 like. 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.

6. To enter a condition in the **Condition** text box, type the condition, for example, **1996**, 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.

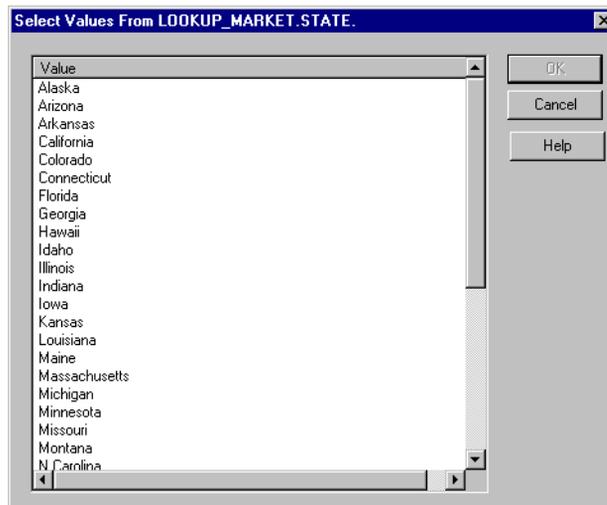


Figure 5-11: Setting Values for Conditions

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

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.

7. In the **Member Load Filtering** dialog box, click Add to add the condition to the **Filters** list.
8. If you want to define an additional condition, do the following:
 - a. Repeat steps 5 through 7.
 - b. Do one of the following:
 - 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, do the following:
 - a. Use the SQL syntax required by the relational data source.
 - b. In the **Member Load Filtering** dialog box, 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 3-30.
11. 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 5-13.

- To type a member filter:
 1. In the metaoutline, select the member level to attach to the filter, for example, MARKET.STATE.
 2. To create a row filter, right-click the member level and, from the pop-up menu, select Filter.

The **Member Load Filtering** dialog box is displayed.

3. Select the **Filters** tab.

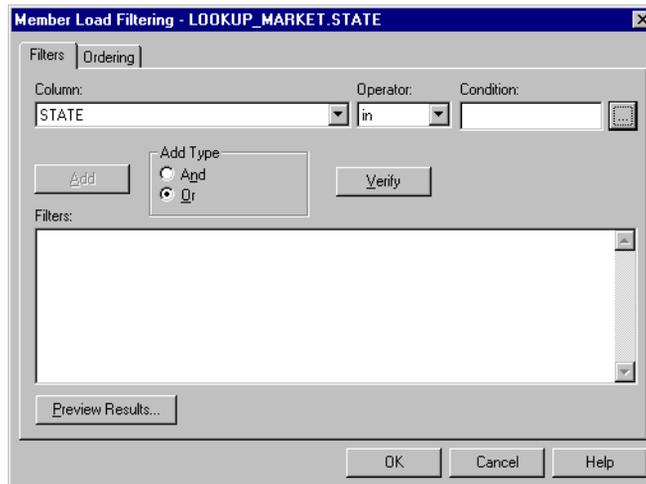


Figure 5-12: Typing a Member Load Filter

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

TableName.ColumnName Operator Condition

For example, a row filter might be as follows:

```
LOOKUP_MARKET.STATE in ('California')
```

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

5. Click Verify to check the SQL syntax.
6. If you want to define an additional condition on the filter, do the following:
 - a. Repeat steps 4 and 5.
 - b. Review step 8b of the previous procedure, “Creating a Member Filter Interactively” on page 5-13, and complete the steps appropriate to your situation.
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 3-30.
8. Click OK.

Creating a Measure Filter Interactively

You can interactively select the parts of a measure filter 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 5-22.

- To create a measure filter interactively:
 1. In the metaoutline, 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.

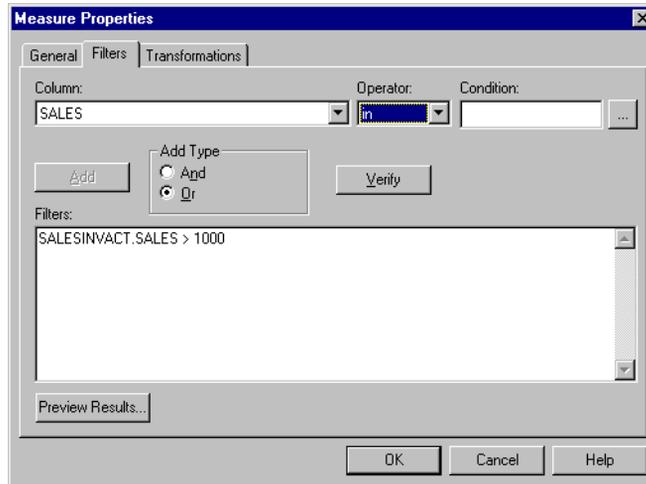


Figure 5-13: Creating a Measure Filter Interactively

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, 1000, 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.

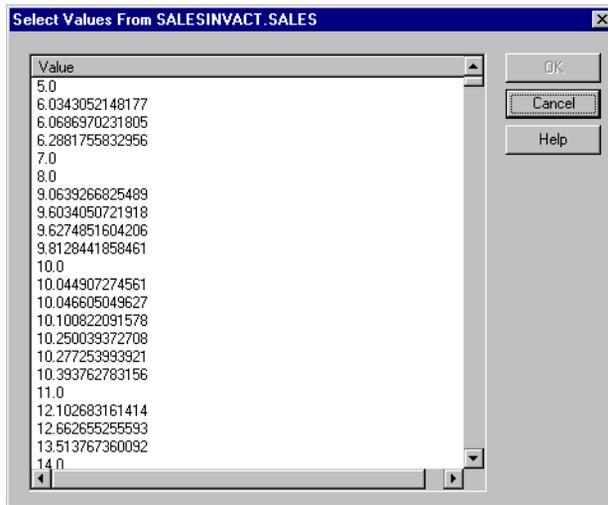


Figure 5-14: Setting Values for Conditions

- b. Select the desired value (such as 1000).

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, do the following:
 - a. Repeat steps 5 through 7.
 - b. Do one of the following:
 - 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, do the following:
 - 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 3-30.
11. Click OK.

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 5-18.

- 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.

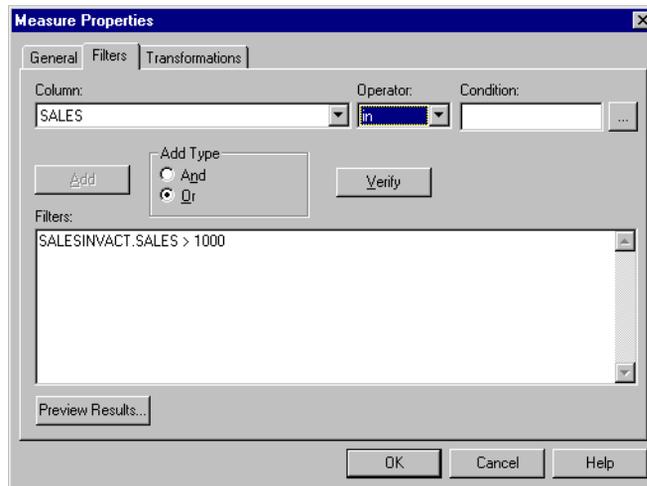


Figure 5-15: Typing a Measure Filter

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

TableName.ColumnName Operator Condition

For example, type the following:

```
SALESINVACT.SALES > 1000
```

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, do the following:
 - a. Repeat steps 4 and 5.
 - b. Do one of the following:
 - 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 3-30.
8. Click **OK**.

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.

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

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 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 6-17
- “Specifying UDAs” on page 6-23
- “Specifying Formulas” on page 6-25
- “Viewing Hyperion Essbase Properties” on page 6-27

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.

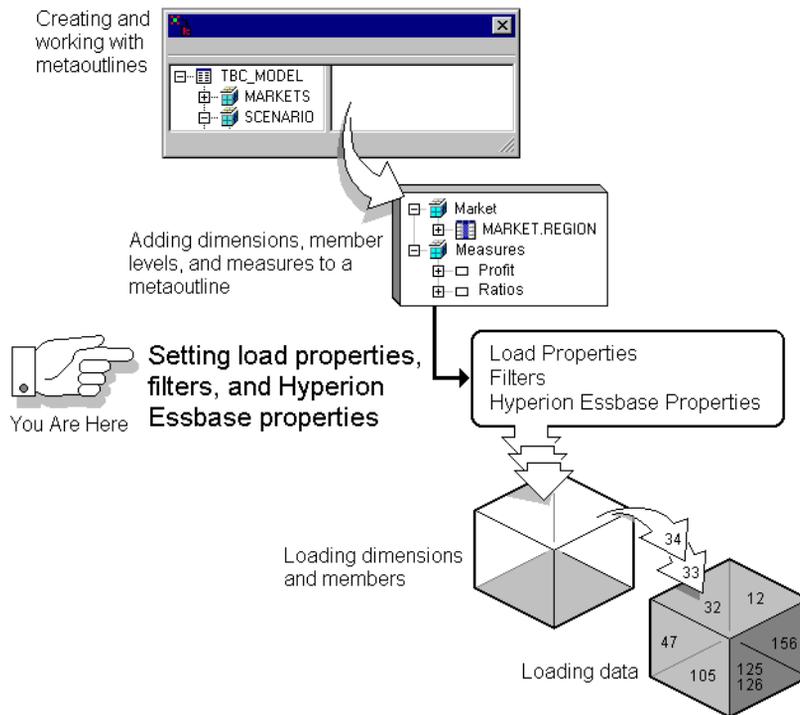


Figure 6-1: Metaoutline Workflow

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 6-5
- “Specifying Dimension Storage Properties” on page 6-6
- “Specifying Data Storage Properties for Dimensions and Members” on page 6-8
- “Setting Aliases for Dimensions and Members” on page 6-10
- “Specifying Two-Pass Calculation on Dimensions and Members” on page 6-12
- “Specifying Member Consolidation Properties” on page 6-14
- “Setting Members as Dynamic Time Series” on page 6-16

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 as type 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. 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 measure in a metaoutline corresponds to the dimension tagged as accounts in the related Hyperion Essbase database.</p> <p>Follow these rules when tagging a measure (accounts) dimension in a metaoutline:</p> <ul style="list-style-type: none"> • You can tag only one dimension in a metaoutline as measure (accounts). • When you tag a dimension as measure (accounts), all members in the dimension inherit the accounts property. • You can create a metaoutline that does not have a dimension tagged as measure (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 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 6-12.

Table 6-1: Dimension Types (Continued)

Type	Description
time	<p>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>Follow these rules when tagging a time dimension in a metaoutline:</p> <ul style="list-style-type: none"> • You can tag only one dimension in a metaoutline as time. • When you tag a dimension as time, all members in the dimension inherit the time property. • 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. • You can create a metaoutline that does not have a dimension tagged as time. • You can add time members to a dimension that is not tagged as time. • 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 Database Administrator's Guide</i>. • 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 6-12.

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 may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

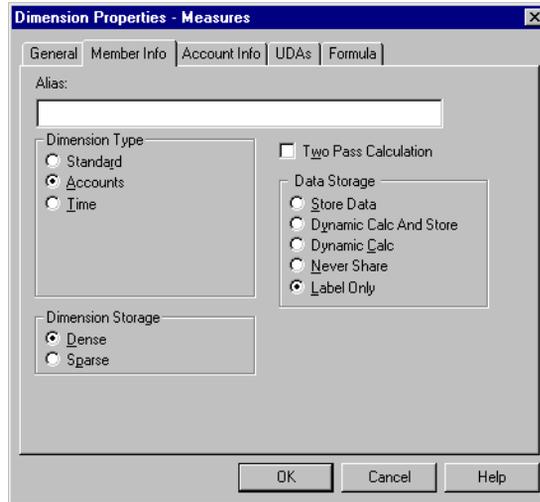


Figure 6-2: Setting a Dimension Type

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 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 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 may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

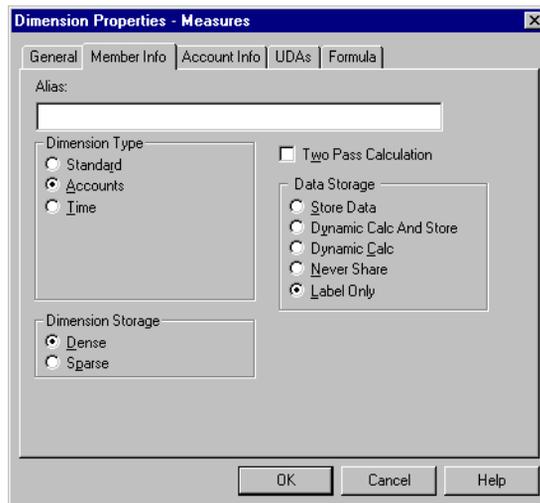


Figure 6-3: Setting Dense or Sparse Dimensions

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 may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

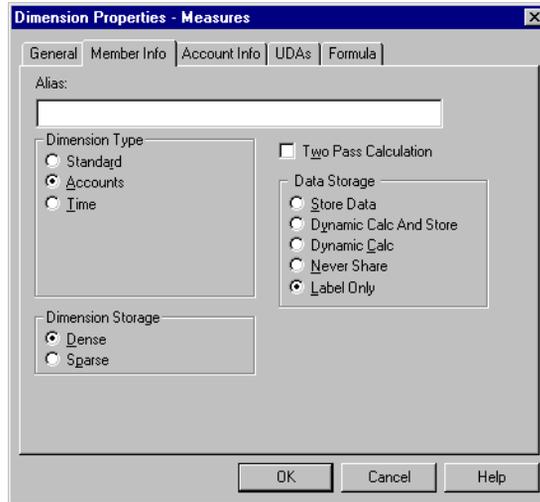


Figure 6-4: Setting Storage Properties

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 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 `PRODUCT_ID` column and member aliases from the `PRODUCT_DESC` column.

<code>PRODUCT_ID</code>	<code>PRODUCT_DESC</code>
100-10	Kool 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 may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

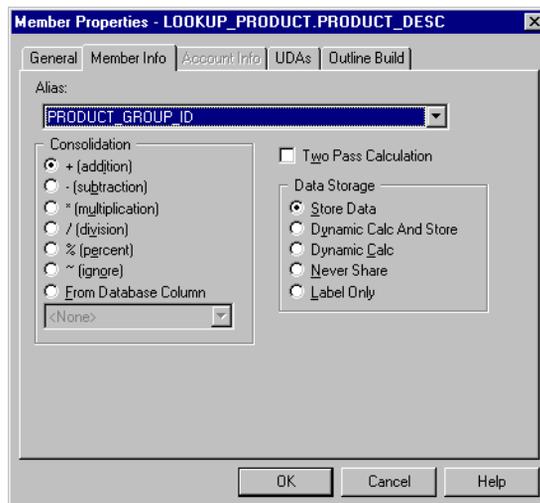


Figure 6-5: Assigning Aliases

4. Do one of the following:
 - 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 3-9.

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 may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

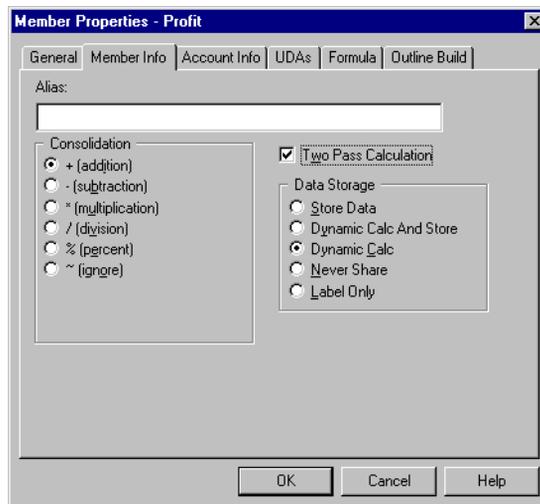


Figure 6-6: Specifying Two-Pass Calculation

4. Select **Two Pass Calculation**.

For more information about two-pass calculation, see the *Hyperion Essbase 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 6-3 describes each consolidation operator.

Table 6-3: 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.

You may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

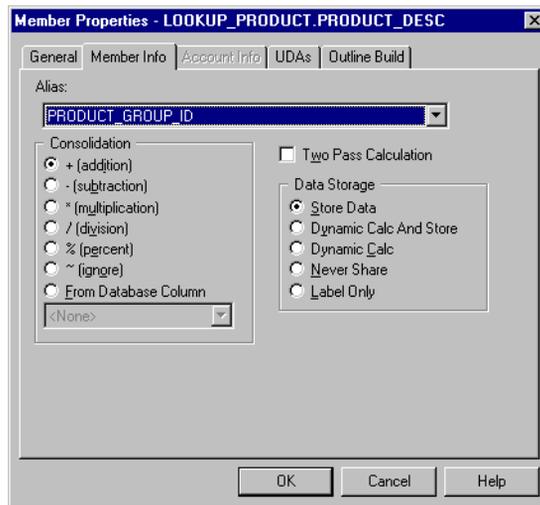


Figure 6-7: Choosing Member Consolidation

4. From the **Consolidation** group, select the appropriate consolidation operator. For more information about member consolidation, see Table 6-3 and the *Hyperion Essbase 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 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 must 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.

You may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

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

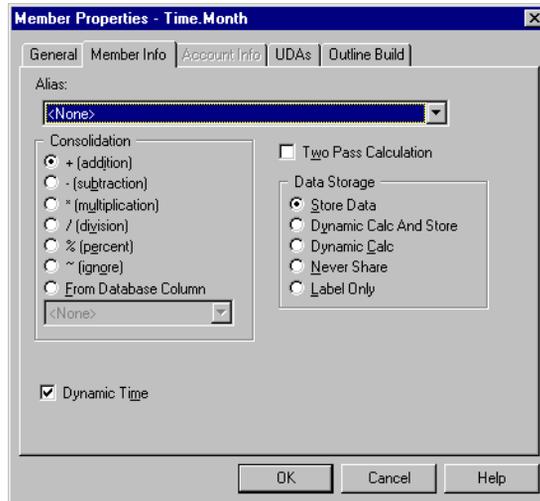


Figure 6-8: Setting a Member as Dynamic Time Series

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 6-18
- “Specifying Variance Reporting Properties” on page 6-21

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 6-4: 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 6-5 describes each skip property.

Table 6-5: 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, OPENING_INVENTORY.
 2. Right-click the member level and, from the pop-up menu, select Properties.

You may also open the dialog box by clicking  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 6-5.

The **Account Info** tab is not selectable for non-account type dimensions and members.

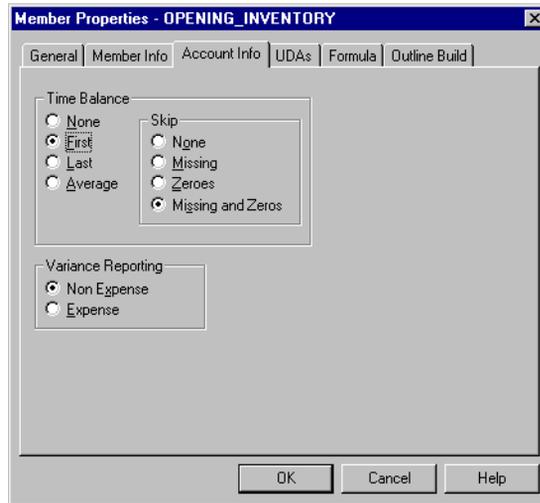


Figure 6-9: Setting Time Balance Properties

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 a @VAR or @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 6-25.

Table 6-6: 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, DISCOUNTS.
 2. Right-click the member level and, from the pop-up menu, select Properties.

You may also open the dialog box by clicking  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 6-5.

The **Account Info** tab is not selectable for non-account type dimensions and members.

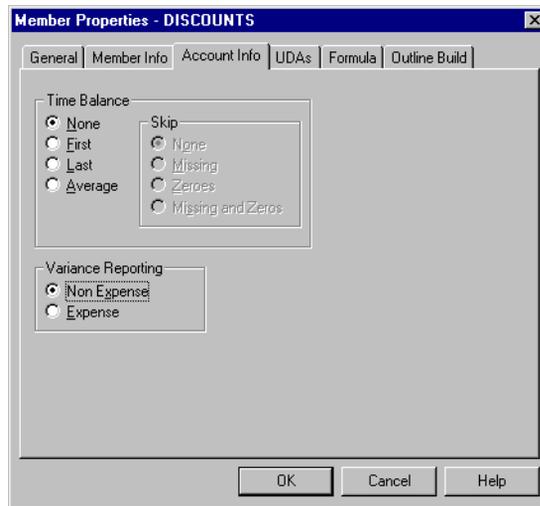


Figure 6-10: Tagging Members as Expense or Non-Expense

4. From the **Variance Reporting** group, select **Non Expense** or **Expense**.

For more information about variance reporting, see Table 6-6 and the *Hyperion Essbase 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 calc scripts and report scripts. For more information, see the *Hyperion Essbase 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 3-9.
- 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.

You may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

3. Select the **UDAs** tab.

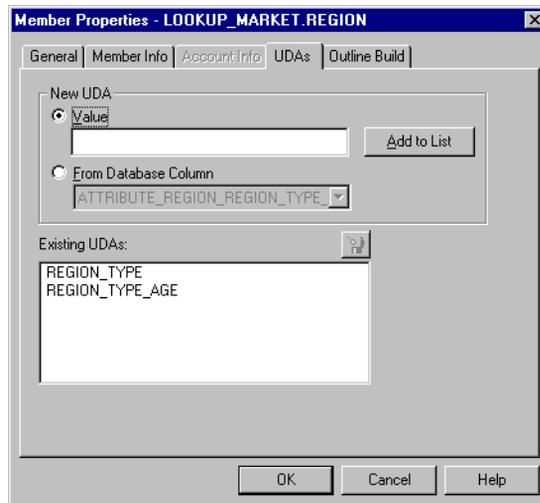


Figure 6-11: Defining UDAs

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 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.

You may also open the dialog box by clicking  from the toolbar or by selecting Properties from the Edit menu.

3. Select the **Formula** tab.

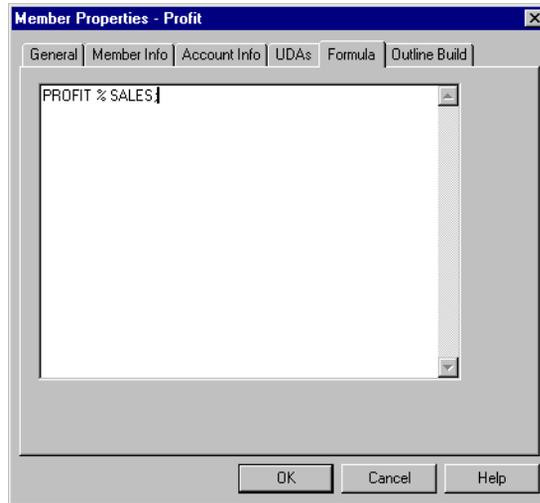


Figure 6-12: Typing Formulas

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 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 column 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 column 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.

5. If you want to display all Hyperion Essbase properties, select **All**.

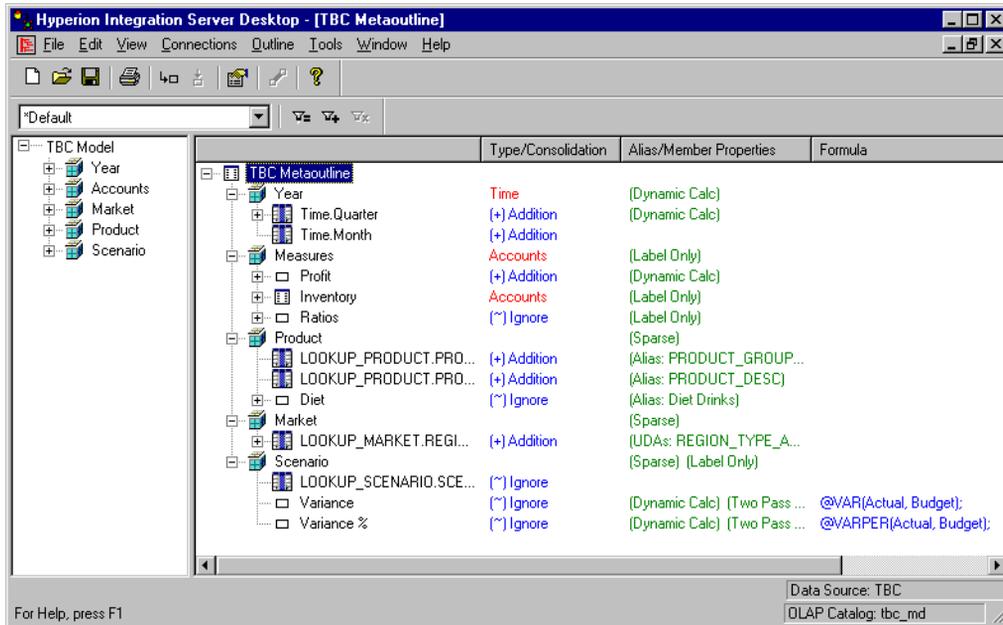


Figure 6-13: Viewing Hyperion Essbase Properties in Column Format

When the Display as Columns command is deselected, the column view is replaced by the condensed format view, as shown in Figure 6-13. To return to the column 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**.

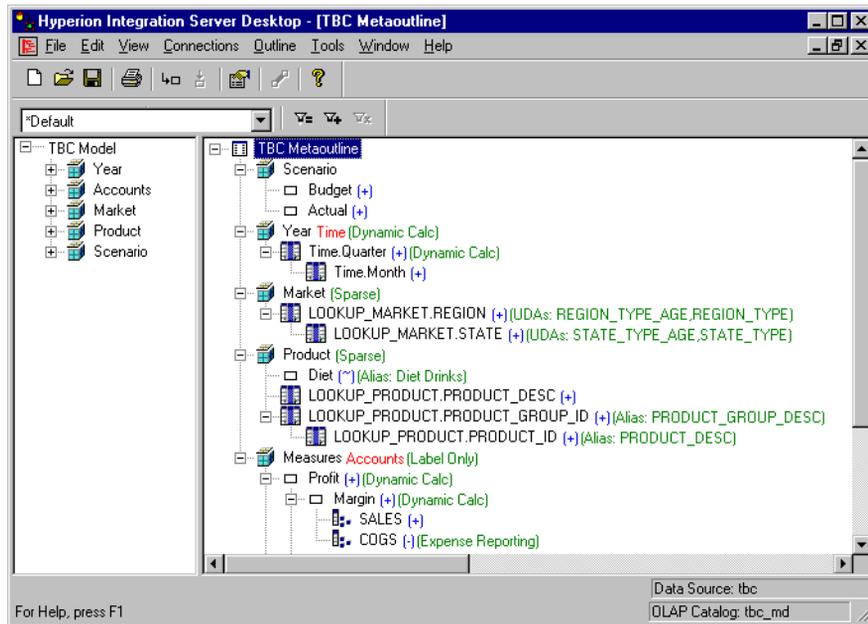


Figure 6-14: Viewing Hyperion Essbase Properties in Condensed Format

To return to the column view, select Display as Columns.

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 7-16
- “Viewing Scheduled Member and Data Loads” on page 7-18
- “Deleting Scheduled Member and Data Loads” on page 7-21

If you are not sure how to view a Hyperion Essbase outline, see the *Hyperion Essbase 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.

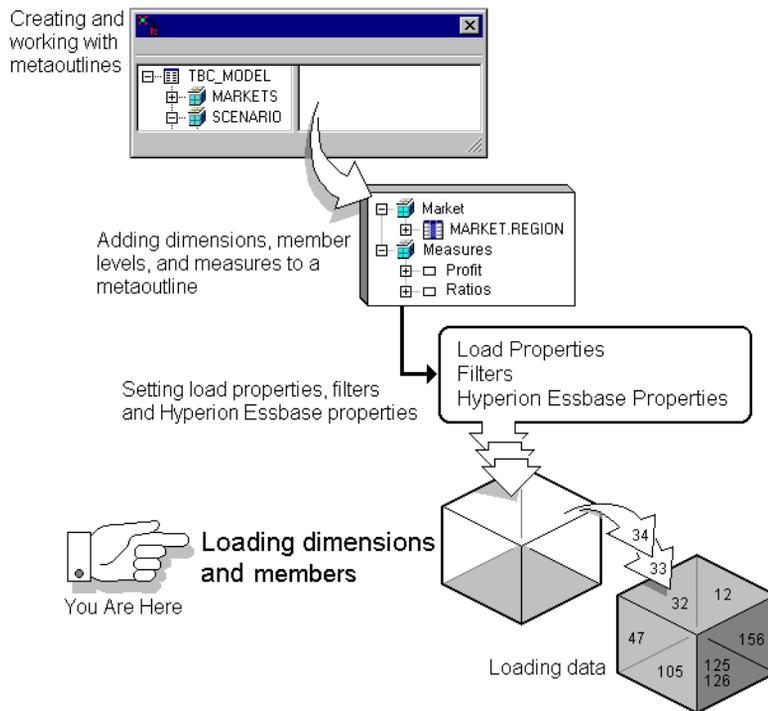


Figure 7-1: Metaoutline Workflow

About Loading

When you perform a load, OLAP Integration Server performs the following tasks:

- 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.
- Retrieves data from the external data source.
- Performs on the retrieved data the transformations specified in the OLAP model and the metaoutline.
- Loads the transformed data into the specified Hyperion Essbase application and database. If no Hyperion Essbase application and database exists for a member load, OLAP Integration Server creates an application and database.

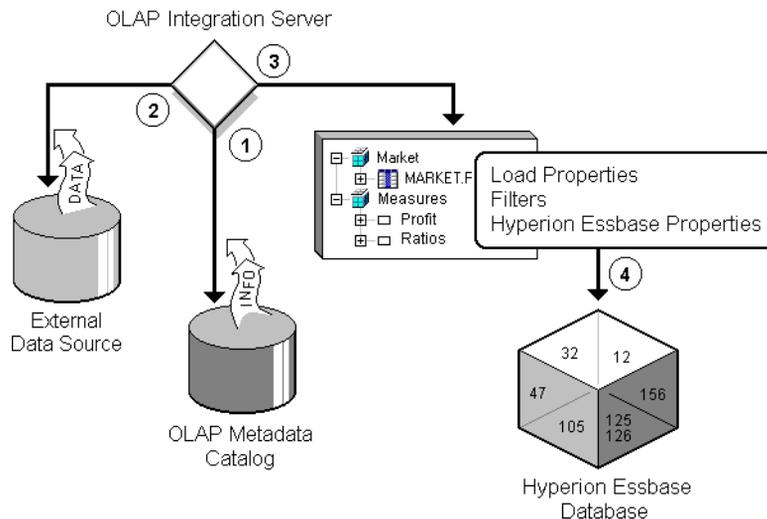


Figure 7-2: Loading Process

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)
[] (bracket)	. (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 where 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 the load.

See “Viewing the Contents of a Metaoutline Filter” on page 5-11.

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 if it is taking too long or retrieving too much data.

For example, if you set the maximum rows to retrieve 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 Database Administrator's Guide*.

- To set a query governor:
1. In the metaoutline, select the metaoutline name.
 2. Right-click the metaoutline name, and choose one of the following options:
 - From the pop-up menu, select Properties.
 - From the toolbar, select the  button.

3. Select the **Governors** tab.

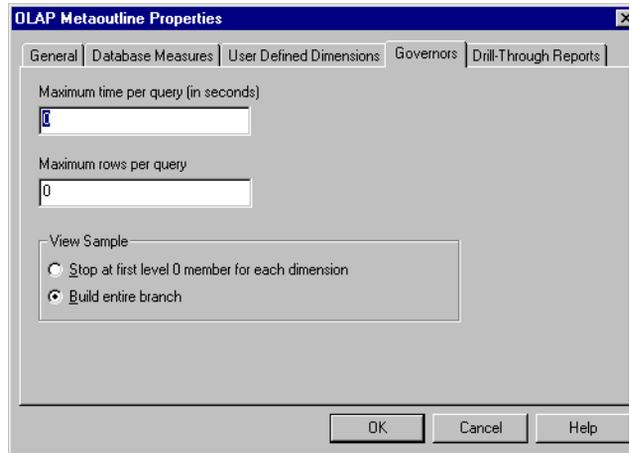


Figure 7-3: Setting a Query Governor

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**.

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 Database Administrator's Guide*.

6. If you are setting a query governor to view a sample outline or dimension, select one of the following:
 - **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 members below it in the hierarchy.
 - **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**.
7. 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.

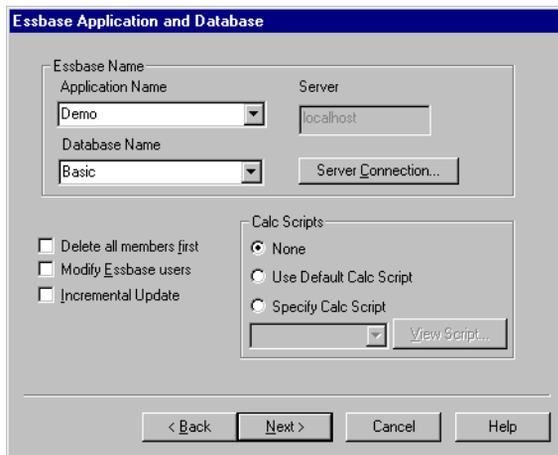


Figure 7-4: Choosing the Hyperion Essbase Application and Database

2. Do one of the following:

- 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 8 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, do the following:
 - a. Select **Modify Essbase users**.

When you select **Modify Essbase users**, the **Essbase Users** dialog box is displayed. The box shows all available and selected Hyperion Essbase users:

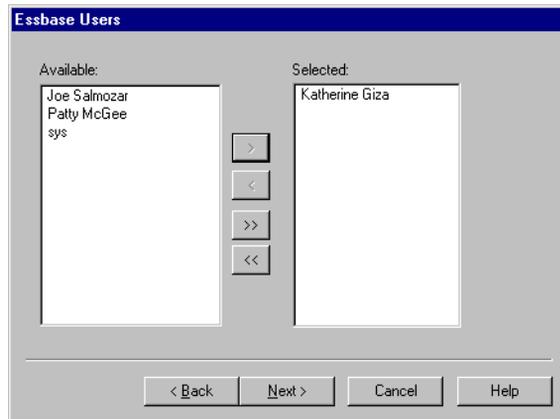


Figure 7-5: Selecting Hyperion Essbase Users

OLAP Integration Server retrieves the list of available users from the following places:

For a(n)	OLAP Integration Server retrieves the users from...
New application	The Hyperion Essbase server. All of the 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

- 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.

If you choose to update the Hyperion Essbase outline incrementally, the **Incremental Update** dialog box is displayed:

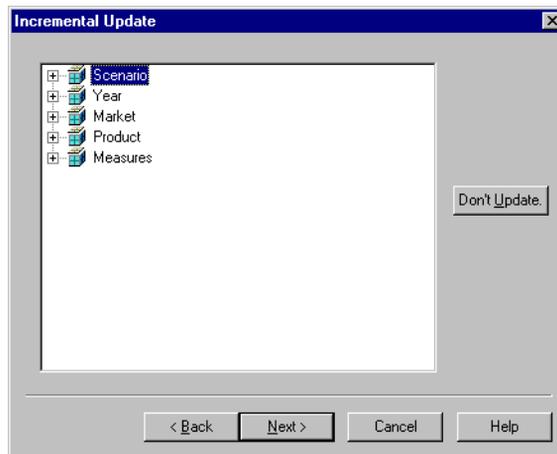


Figure 7-6: Updating Hyperion Essbase Outlines Incrementally

- a. To prevent OLAP Integration Server from updating a dimension or member, select the dimension or member and click Don't Update.
- b. When you have marked all dimensions and members that OLAP Integration Server should not update, click Next.

6. Proceed to “Procedure 3: Loading Members Immediately” on page 7-13 or to “Procedure 4: Scheduling a Member Load” on page 7-16.



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*.

7. Select the calc script to use to calculate the Hyperion Essbase database.

If you select...	OLAP Integration Server...
None	Does not run a calc script on the Hyperion Essbase database.
Default	Runs the default calc script that is specified for the Hyperion Essbase database. If you have not specified a calc script to use as the default, Hyperion Essbase uses standard calculation order to calculate the database.
Any other calc script	Runs the selected calc script on the Hyperion Essbase database.

The calc script that you choose is run after OLAP Integration Server loads data into the Hyperion Essbase database. The calc script determines how Hyperion Essbase calculates the data values in the database.

For more information on calculation and on calc scripts, see the *Hyperion Essbase 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.

➤ 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:

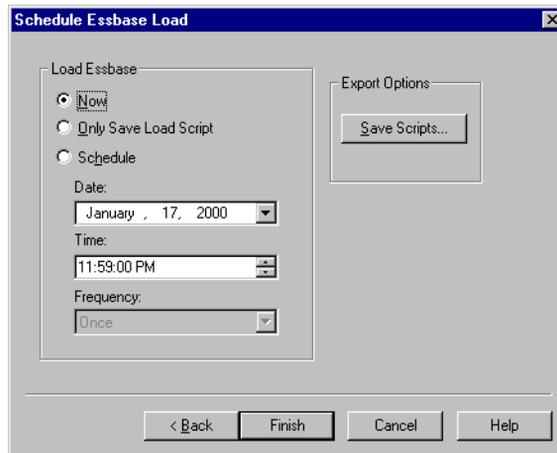


Figure 7-7: Loading Members Immediately

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:

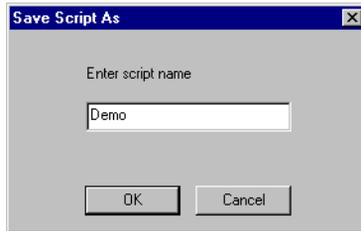


Figure 7-8: Saving a Script

- 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:

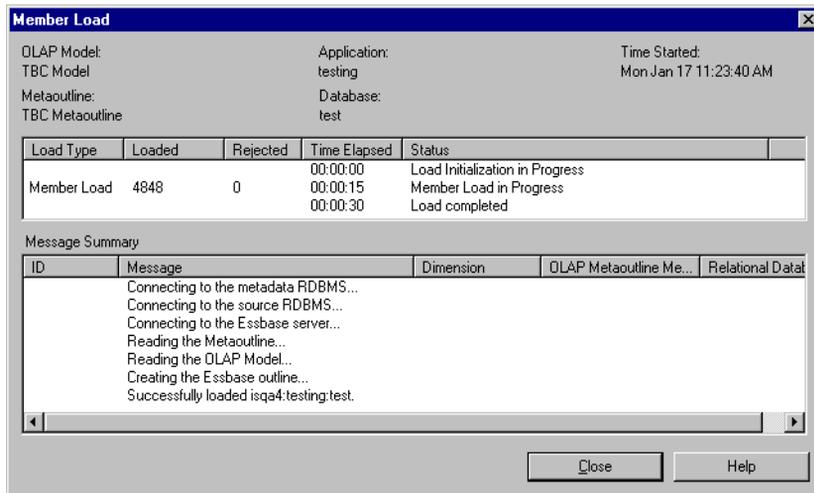


Figure 7-9: Displaying Member Load Information

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.

For example, as shown in Figure 7-10, if the three member levels of the Product dimension do not load, three .txt files are created: PRODUCT_0.txt (PRODUCT.DIET), PRODUCT_1.txt (PRODUCT.PRODUCT_ID), and PRODUCT_2.txt (PRODUCT.DESC).

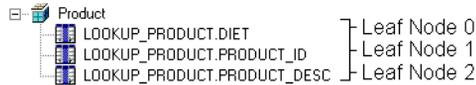


Figure 7-10: Leaf Node Example

For each reject file, a corresponding rules file is created. The rules files created for the rejected member levels of Figure 7-10 are PRODUCT_0.rul (PRODUCT.DIET), PRODUCT_1.rul (PRODUCT.PRODUCT_ID), and PRODUCT_2.rul (PRODUCT.DESC). Both .txt and .rul files are located in the I\$HOME\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 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 7-17 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, January 17, 2000.
 - b. In the **Time** text box, enter the time. Use the hh:mm:ss format, for example, 23:59. To schedule a load to run at midnight, enter 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 7-18. To delete a scheduled member load, see “Deleting Scheduled Member and Data Loads” on page 7-21.

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,  Services, to open the **Services** dialog box.

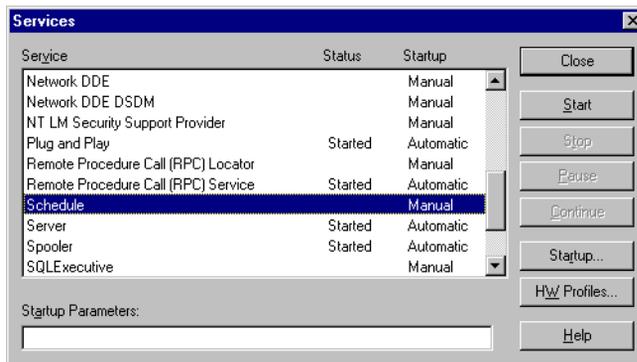


Figure 7-12: Starting the Windows NT Schedule Service

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. 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 7-17.

➤ 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. 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.

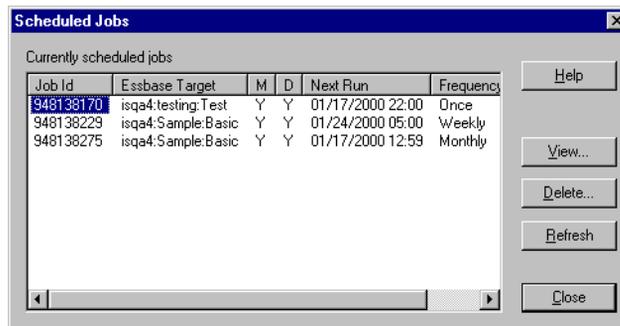


Figure 7-13: Viewing a List of Scheduled Loads

Note: To update the list of scheduled loads, click Refresh.

- 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.

The screenshot shows a dialog box titled "View existing job." with the following sections:

- Schedule Parameters:**
 - Date: 01/17/2000
 - Time: 22:00
 - Frequency: Once
- OLAP Catalog:**
 - Connection String: "DSN=tbc_md;UID=tbc;PWD=password;"
 - Metaoutline: TBC Metaoutline
- Essbase:**
 - Server: isqe4
 - User: hyperion
 - Password: [masked]
 - Application: testing
 - Database: Test
 - Member Load
 - Data Load
 - Parameters: "APP=testing;DBN=Test;OTL=TBC Metaoutline;FLT_ID=1;OTL_C"
- Source Database:**
 - Connection String: "DSN=tbc;UID=tbc;PWD=password;"

Buttons for "OK" and "Help" are located on the right side of the dialog.

Figure 7-14: Viewing a Description of the Selected Load

- When you finish viewing the load properties, click OK.
- 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 7-17.

► 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:

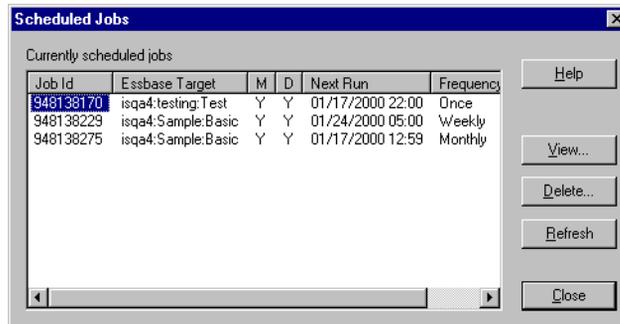


Figure 7-15: Viewing a List of Scheduled Loads

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 calc 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 8-2
- “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 8-11

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 for 1-2-3* or *Hyperion Essbase Spreadsheet Add-in User’s Guide for Excel*.

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 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.

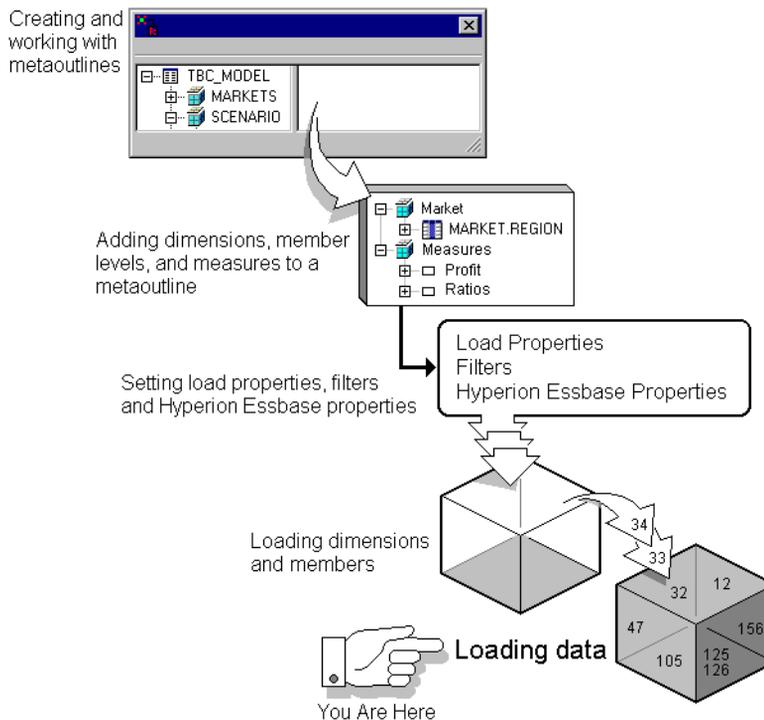


Figure 8-1: Metaoutline Workflow

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 where 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 the load.
See “Viewing the Contents of a Metaoutline Filter” on page 5-11.
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.

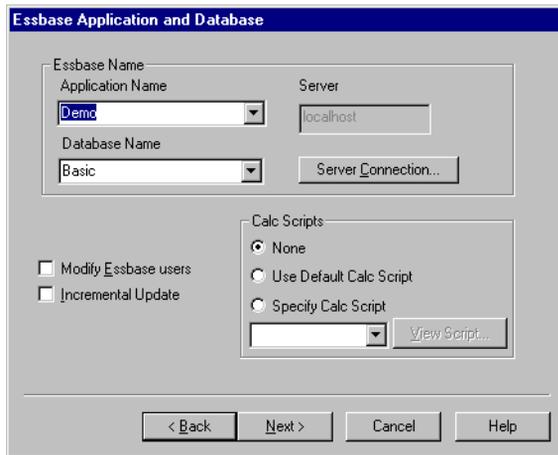


Figure 8-2: Choosing the Hyperion Essbase Application and Database

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:

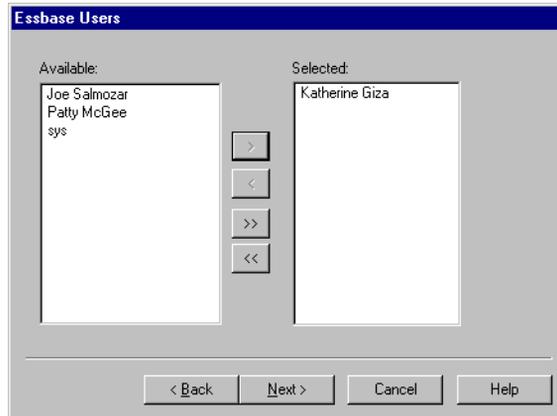


Figure 8-3: Selecting Hyperion Essbase Users

OLAP Integration Server retrieves the list of available users from the following places:

For a(n)	OLAP Integration Server retrieves the users from the...
New application	Hyperion Essbase server. All users can create new applications and databases on the Hyperion Essbase server, for example, <i>sys</i> .
Existing application	Hyperion Essbase application
New database	Hyperion Essbase application
Existing database	Hyperion Essbase database

6. To allow a user to access the Hyperion Essbase application and database, do the following:
 - 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 dimensions 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.

During a data load, you can select to update or not to update only members of the dimension tagged as accounts.

If you choose to update the Hyperion Essbase outline incrementally, the **Incremental Update** dialog box is displayed:

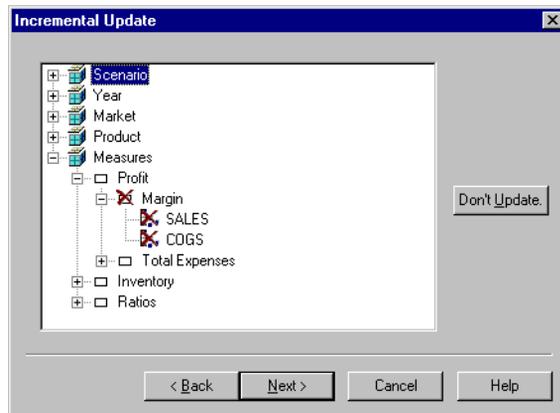


Figure 8-4: Updating a Hyperion Essbase Outline Incrementally

- a. To prevent OLAP Integration Server from updating a dimension or member, select the dimension or member and click Don't Update.

Note: You must update at least one member from each dimension when performing a data load. If more than one hierarchy exists in the Hyperion Essbase outline, you can perform an incremental update.

- b. After you mark all dimensions or 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*.

8. Select the calc script to use to calculate the Hyperion Essbase database.

If you select...	OLAP Integration Server...
None	Does not run a calc script on the Hyperion Essbase database.
Default	Runs the default calc script that is specified for the Hyperion Essbase database. If you have not specified a calc script to use as the default, Hyperion Essbase uses standard calculation order to calculate the database.
Any other calc script	Runs the selected calc script on the Hyperion Essbase database.

The calc script that you choose is run after OLAP Integration Server loads data into the Hyperion Essbase database. The calc script determines how Hyperion Essbase calculates the data values in the database.

For more information on calculation and on calc scripts, see the *Hyperion Essbase Database Administrator's Guide*.

9. 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.

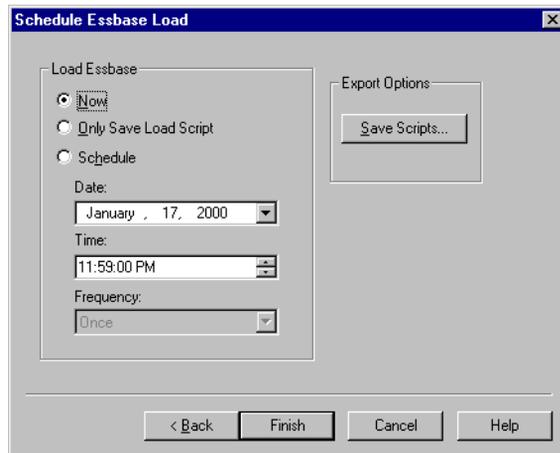


Figure 8-5: Loading Data Immediately

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.
3. In the **Schedule Essbase Load** dialog box, select **Now**.
 4. Click Finish.

The **Data Load** dialog box is displayed:

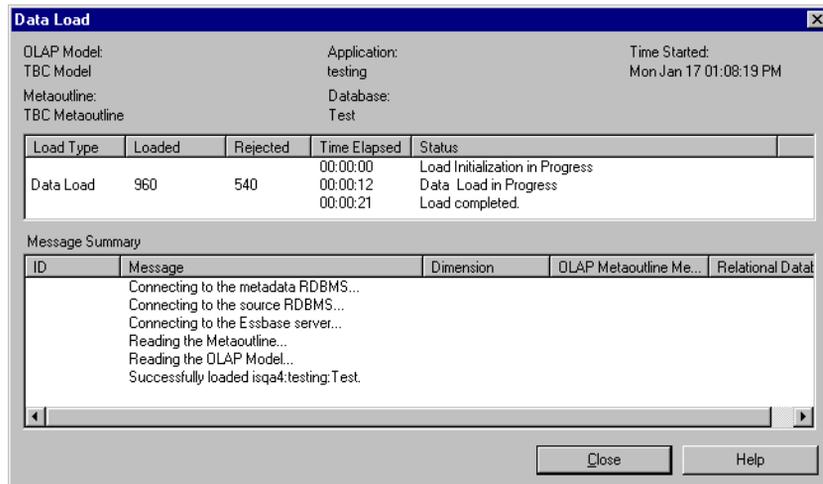


Figure 8-7: Displaying Data Load Information

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, as shown in Figure 8-8, if the three member levels of the Product dimension do not load, three `.txt` files are created; for example, `PRODUCT_0.txt` (PRODUCT.DIET), `PRODUCT_1.txt` (PRODUCT_ID), and `PRODUCT_2.txt` (PRODUCT_DESC).



Figure 8-8: Leaf Node Example

For each rejected file, a corresponding rules file is created. The rules file created for the rejected member levels in Figure 8-8 are `PRODUCT_0.rul` (PRODUCT.DIET), `PRODUCT_1.rul` (PRODUCT_ID), and `PRODUCT_2.rul` (PRODUCT.DESC).

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 3: Loading Data Immediately” on page 8-8.

The **Load Essbase When** dialog box is displayed:

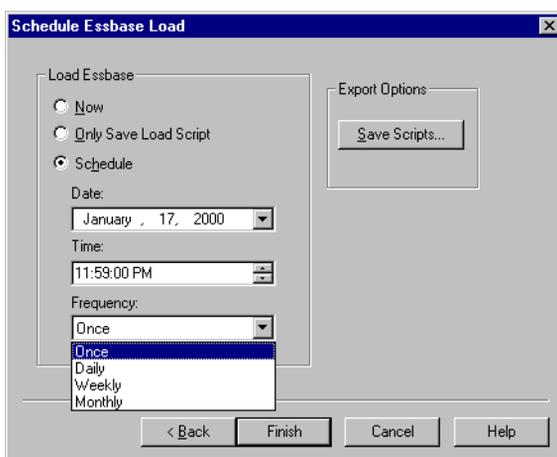


Figure 8-9: Scheduling a Data Load

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 7-17.

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, January 17, 2000.
 - b. In the **Time** text box, enter the time. Use the hh:mm:ss format, for example, 23:59:00. 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 7-21. To view a scheduled data load, see “Viewing Scheduled Member and Data Loads” on page 7-18.

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 9-15
- “Specifying Spreadsheet User Actions” on page 9-20
- “Working with Reports” on page 9-25

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?

Some spreadsheet users may need more detailed information to create a strategy to increase root beer sales. They need to answer questions such as What are the ten lowest performing stores in California? What are the addresses of the stores? What are the store managers' names? 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.

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:

- “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.

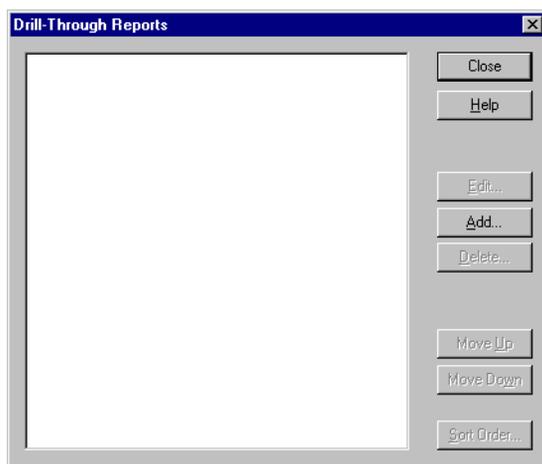


Figure 9-1: Drill-Through Reports Dialog Box

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 Report1.

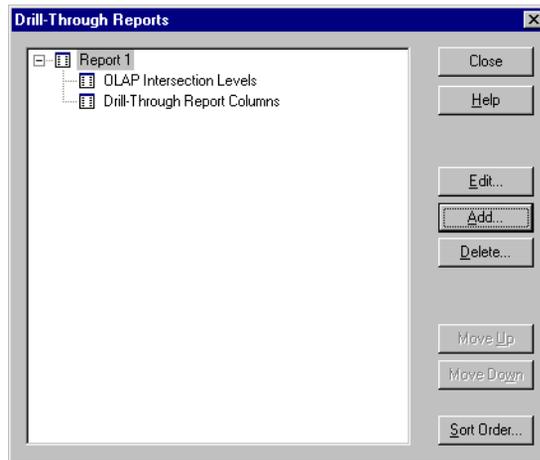


Figure 9-2: New Drill-Through Report

4. If you want to rename the report, see “Renaming a Report in the Drill-Through Reports Dialog Box” on page 9-25.
5. If you want to add intersection levels and columns to the report, see “Adding Intersection Levels” on page 9-10 and “Adding Drill-Through Columns” on page 9-12.

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 9-25.
6. If you want to add or change intersection levels and columns in the report, see “Adding Intersection Levels” on page 9-10 and “Adding Drill-Through Columns” on page 9-12.

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 9-10
- “Adding Drill-Through Columns” on page 9-12
- “Setting the Size of a Query” on page 9-14

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, you can define a report for the intersections of the Quarter, SALES, REGION, PRODUCT_ID, and Month member levels of TBC Metaoutline.

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 level described in the previous paragraph, a spreadsheet user can select any of various intersections, such as the intersection of Qtr1, Profit, Retail, Connecticut, Bottles, 100-20, 1997 or the intersection of Qtr2, Sales, Retail, Florida, Cans, 100-30, 1996.

An intersection level affects all members of the specified dimension at the specified level. For example, if you set an intersection level on `PRODUCT_GROUP_ID` in the Product 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 Product dimension at level 1 and lower can access the Drill-Through report. Thus, both the members under `PRODUCT_GROUP_ID` and the members under Diet (because Diet is also a level 1 member of the Product dimension) can access the report.

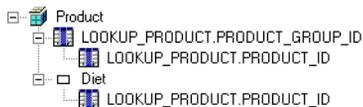


Figure 9-3: Intersection Levels

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, in a metaoutline, you define an intersection level, 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 accounts.
- **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 upon the designated metaoutline member level. For example, if you set an intersection level at PRODUCT_ID, spreadsheet users can double-click any data value that involves a member based upon PRODUCT_ID, 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

This topic uses a simple metaoutline of four dimensions and illustrates how the setting of various intersection levels affects the reports that spreadsheet users see.

The sample metaoutline contains the following dimensions and member levels:

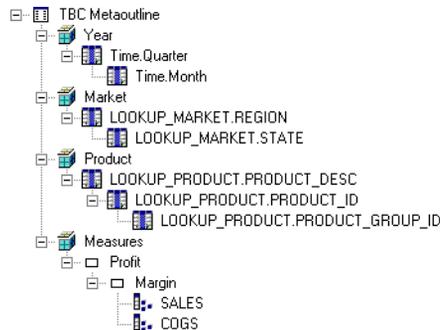


Figure 9-4: Simple Drill-Through Metaoutline

Now, suppose that you define the following reports:

- A demographics report on the intersection level of MONTH, PRODUCTS, CHANNEL, and ACCOUNTS. To view the demographics report, spreadsheet users drill down to any month under any quarter. Because the demographics report is defined at the dimension level for all dimensions other than TIME, spreadsheet users can view the report at any level of the Products, Channel and Accounts dimensions. The worksheet excerpt in Figure 9-5 shows a spreadsheet at the highest level from which spreadsheet users can view the

demographics report. If users drill down on one of the dimensions, they can still view the demographics report. If they drill up from one of the MONTH members, they cannot view the demographics report.

			ACCOUNTS
CHANNEL	PRODUCTS	QTR_1Feb	107573.2871
		QTR_1Jan	306131.7676
		QTR_1Mar	127575.3608

Figure 9-5: Simple Demographics Report

- An advertising report on the intersection level of MONTH and PRODUCTS. If a report is not defined on a dimension or on any member of the dimension, spreadsheet users can view the report from any member of the dimension. Therefore, spreadsheet users can view the advertising report from any member of CHANNEL or Accounts as well as from any member of PRODUCTS and any member of the TIME dimension that is located at or below the month level.
- A financial report on the intersection level of MONTH and COGS. To view the financial report, spreadsheet users drill down to any month under any quarter and to the COGS member of the ACCOUNTS dimension. If a report is not defined on a dimension or on any member of the dimension, spreadsheet users can view the report from any member in the dimension. The worksheet excerpt in Figure 9-6 shows a spreadsheet at the highest level from which spreadsheet users can view the financial report. If spreadsheet users drill down on one of the dimensions, they can still view the financial report. If they drill up from COGS or from one of the MONTH members, they cannot view the financial report.

COGS	CHANNEL	QTR_1Feb	25887.34
		QTR_1Jan	81171.76
		QTR_1Mar	30823.72

Figure 9-6: Simple Financial 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.

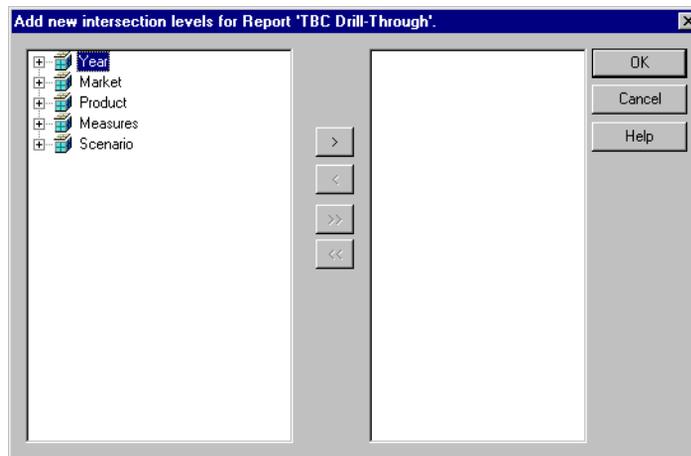


Figure 9-7: Adding Intersection Levels

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 .

Figure 9-8 shows the set of intersection levels defined in the TBC Drill-Through report.

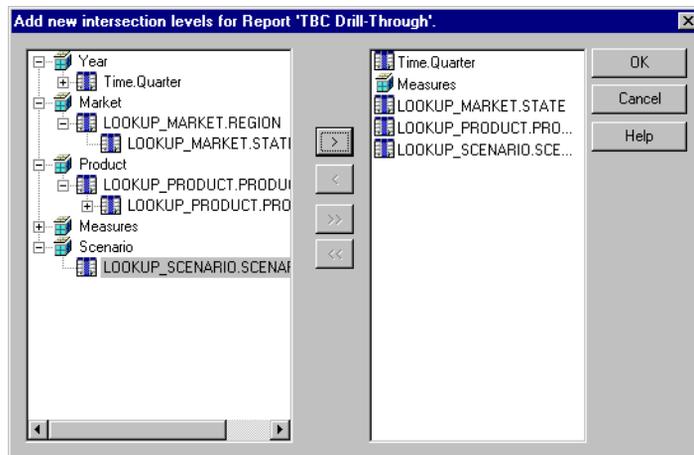


Figure 9-8: Intersection Levels

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 9-10.
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.

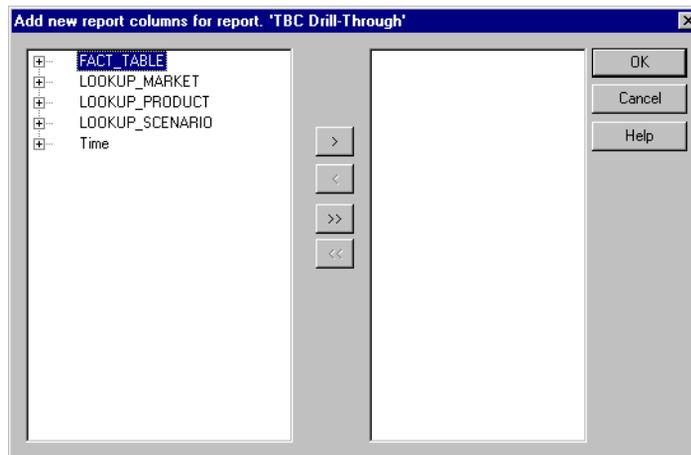


Figure 9-9: Adding Drill-Through Columns

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 .

Figure 9-10 contains the set of columns for the TBC Drill-Through report. These columns provide spreadsheet users a view of details on store demographics and store managers.

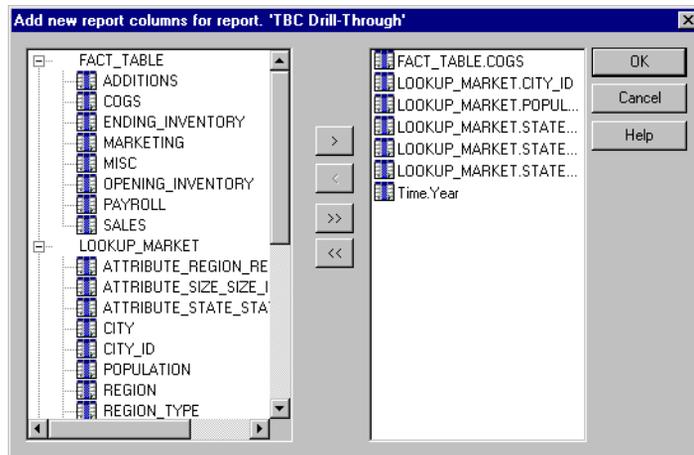


Figure 9-10: Sample Drill-Through Report Columns

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.

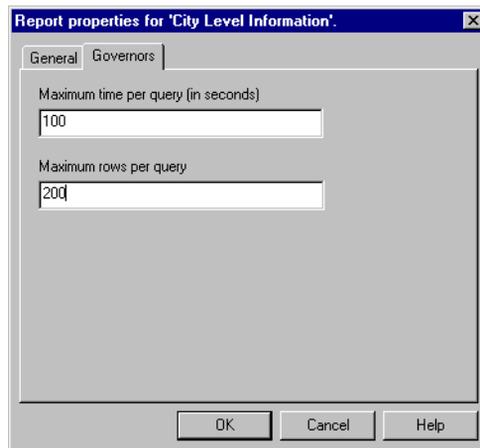


Figure 9-11: Setting a Query Governor

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 9-16
- “Specifying Row Ordering” on page 9-18

You can, if you want, allow spreadsheet users to specify the order of columns and rows in a Drill-Through report. See “Customizing a Report” on page 9-21.

Specifying Column Ordering

Columns are displayed in a report left to right in the order in which they are listed 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 9-12, CITY is the first column, MKTTYPE the second, and DEMOMIX the third.

	A	B	C	D	E	F	G	H	I
1	CITY	MKTTYPE	DEMOMIX	INCOME	PRODDES	MANAGE	PHONE	YEAR	MONTH
2	New Londc	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be:	408-555-12	1997	Feb
3	New Londc	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be:	408-555-12	1997	Mar
4	New Londc	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be:	408-555-12	1997	Jan
5	New Cana:	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Feb
6	New Cana:	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Mar
7	New Cana:	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Jan
8	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Feb
9	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Mar
10	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be:	408-555-12	1997	Jan
11									
12									
13									
14									

Figure 9-12: Column Order in TBC Drill-Through Report

- 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, MARKET.CITY.

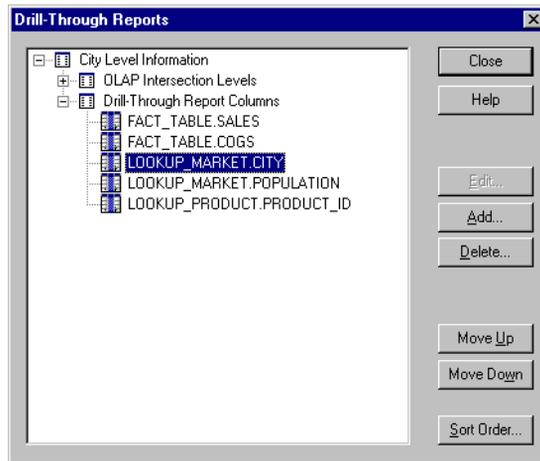
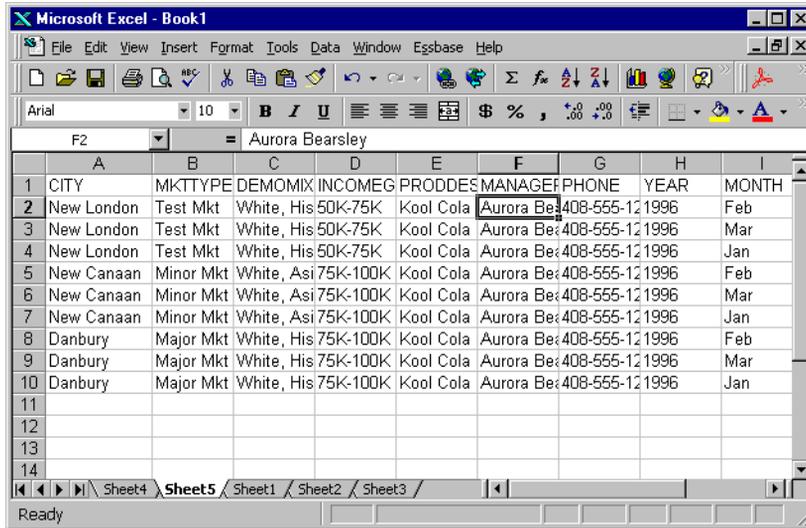


Figure 9-13: Positioning Columns

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 MARKET.CITY column in descending order (from highest to lowest value). This sort presents the cities in reverse alphabetical order, from New London to New Canaan to Danbury and produces the following report:



	A	B	C	D	E	F	G	H	I
1	CITY	MKTTYPE	DEMOMIX	INCOME	PRODDES	MANAGER	PHONE	YEAR	MONTH
2	New London	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be	408-555-12	1996	Feb
3	New London	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be	408-555-12	1996	Mar
4	New London	Test Mkt	White, His	50K-75K	Kool Cola	Aurora Be	408-555-12	1996	Jan
5	New Canaan	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Feb
6	New Canaan	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Mar
7	New Canaan	Minor Mkt	White, Asi	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Jan
8	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Feb
9	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Mar
10	Danbury	Major Mkt	White, His	75K-100K	Kool Cola	Aurora Be	408-555-12	1996	Jan
11									
12									
13									
14									

Figure 9-14: 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.

- 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.

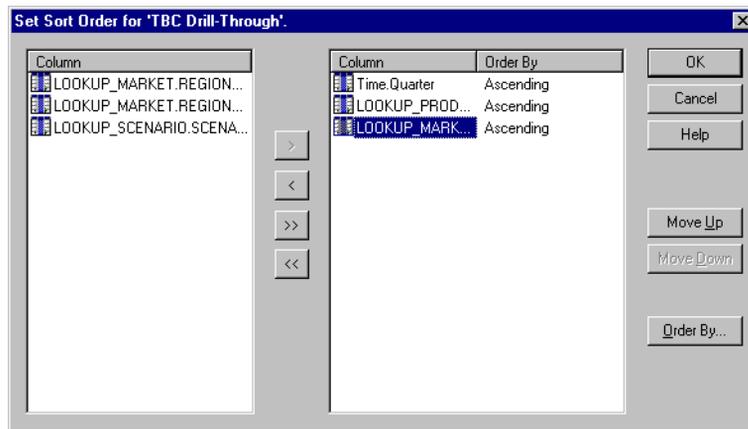


Figure 9-15: Setting Sort Order

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.

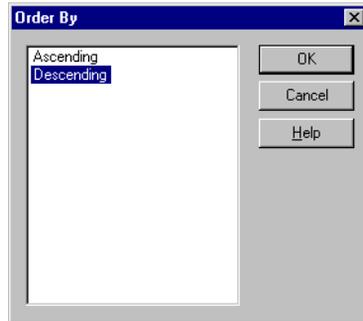


Figure 9-16: Order By Dialog Box

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, use a specific server running OLAP Integration Server, and 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 9-21
- “Specifying the OLAP Integration Server Name” on page 9-22
- “Specifying a Database Username and Password” on page 9-23

Customizing a Report

You can allow 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 allow spreadsheet users to customize a report, they see all columns that you specify and use the row and column ordering that you specify.

- To allow spreadsheet users to customize a report:
 1. In the **Drill-Through Reports** dialog box, select the report that you want to allow spreadsheet users to customize.
 2. Click Edit to open the **Report properties** dialog box.



Figure 9-17: Allowing 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 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 the ODBC data source name 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 use OLAP_TBC as the ODBC data source name for the external data source and OLAP_MD as the ODBC data source name for the 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 names for the 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.

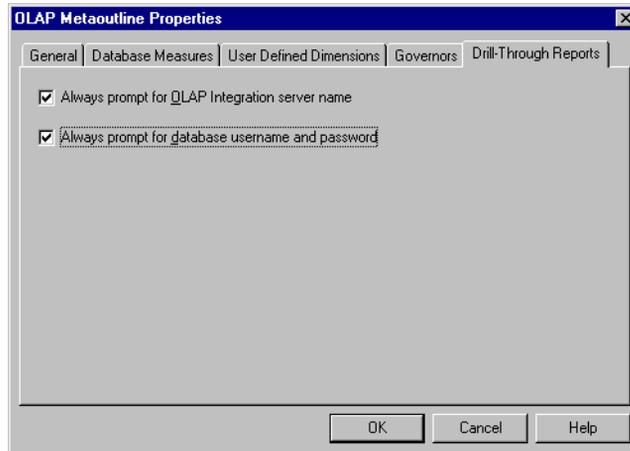


Figure 9-18: Prompting for the Name of the Server Running Hyperion Integration Server

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.

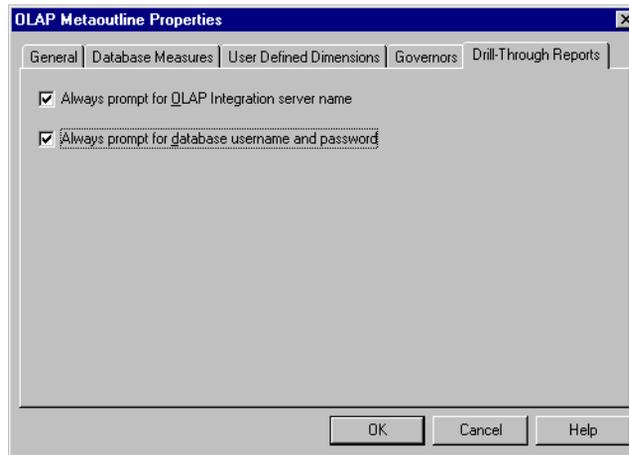


Figure 9-19: Prompting for a Username and Password

4. Select **Always prompt for database username and password**.

To allow 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 9-25
- “Renaming a Report in the Report Properties Dialog Box” on page 9-26
- “Deleting an Individual Report, an Intersection Level, or a Drill-Through Column” on page 9-27
- “Deleting All Intersection Levels or Drill-Through Columns in a Report” on page 9-27

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, .
 3. Type the new report name.
See “About Naming” on page 3-9.
 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 9-20: Renaming a Report

4. In the **Name** text box, type the new name.
See “About Naming” on page 3-9.
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:

- 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 3-30.)

- 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 do any of the following:
 - 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:

- 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 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 Database Administrator’s Guide*.

Problems with Filters

If member names, dimension names, or data are not being filtered correctly, consider the following:

- 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 5-13.
- 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 3-30.
- 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 3-30) or check the SQL statements in the Hyperion Integration Server log file (see the *Hyperion Essbase Database Administrator’s Guide*).

Many filter errors do not show up until you perform a member 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:

- 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 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 3-9.

- 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 Database Administrator’s Guide*.
- Is a scheduled load not appearing 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 7-17.
- 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 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 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:

- 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 3-9.
- Did you perform the load correctly, but some members or data are missing? Check the following:
 - 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 3-30.
 - Did you specify that OLAP Integration Server should create duplicate members as shared? See “Handling Duplicate Member Names” on page 4-6.
 - 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 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 5-13.

- Did you remember to calculate the data before viewing it in Hyperion Essbase Spreadsheet Add-in? See the *Hyperion Essbase 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 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 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.

#MISSING. *See* missing data.

Accounts dimension. In Essbase, a dimension type that makes accounting intelligence available. You can tag only one dimension in a database as Accounts. Essbase does not require an Accounts dimension. The Accounts dimension in Hyperion Integration Server becomes the Accounts dimension in the Essbase database that you create with Hyperion Integration Server.

In Hyperion Integration Server, an optional dimension you create as a container for measures data values. *See also* Measure dimension, measures.

Add Joins mode. In Hyperion Integration Server, an OLAP Architect state in which you can draw lines to define joins between objects in the OLAP model. *See also* Move mode.

administrator. An individual who installs and maintains the system, including setting up user accounts and security. *See also* database administrator, system administrator.

aggregate. *See* consolidate.

alias. In Essbase, an alternate name for a dimension or member.

alias column. In Hyperion Integration Server, a column in the data source that contains the aliases for a member in the metaoutline.

alias table. In Essbase, a table that stores alias names for members and dimensions.

alternate name. *See* alias.

ancestor. In Essbase, a member or dimension that has members below it. For example, in Sample Basic, Qtr2 and Year are ancestors of April.

application. In Essbase, a container for one or more Essbase databases and files, such as calc scripts and data load rules that are related to the databases. *See also* Essbase database.

ARBORPATH. In Essbase, an environment variable that specifies the Essbase root directory.

Architect. *See* OLAP Architect.

attribute. In Essbase, a dimension or member property that determines functionality. For example, a member that has the two-pass attribute is recalculated on a second calculation pass through the outline.

block. In Essbase, the primary storage unit. A data block is a multidimensional array representing the cells of all dense dimensions.

Builder. *See* OLAP Builder.

calc script. In Essbase, a text file containing a set of instructions telling Essbase how to calculate a database.

Catalog. *See* OLAP Catalog

cell. In Essbase, a unit of data representing the intersection of each dimension combination in a multidimensional database. Also, the intersection of a row and a column in a spreadsheet. *See also* intersection level.

child. In Essbase, a member that has a parent above it in the database outline. *See also* sibling, parent.

cleansing. *See* data cleansing.

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. For example, a column may contain the employee name or employee number for each employee. Every column has a name and a particular datatype. *Contrast with* row.

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 YorkLibrary.

condition. In relational databases, a data extraction criterion. For example, you can apply a condition to extract only the data that starts with the letter A. *See also* filter.

consolidate. In Essbase, a method of computing the data relationships for all parent/child combinations within a dimension. A consolidation is typically additive, but can be any type, including multiplication or subtraction. For example, in Sample Basic, the Year dimension has four children: Qtr1, Qtr2, Qtr3, and Qtr4. Essbase consolidates these members by adding them to calculate Year. The term “consolidate” is also known as “aggregate” and “roll up.”

consolidation level. In Essbase, the level in the database outline up to which Essbase consolidates the database.

database. A collection of related information organized in such a way that a computer program can quickly select desired pieces of data. *See also* Essbase database, relational database.

database administrator. An individual who administers databases, and may also design, maintain, and create databases. *See also* administrator, *See also* system administrator.

database outline. *See* outline.

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 not of the correct datatype.

data file. In Essbase, a file containing data blocks. Essbase generates data files upon loading data and stores them on disk. Data files are also known as page files.

data load. In Essbase, the process of populating an Essbase database with data. Loading data establishes actual values for the cells defined by the database’s structural outline. *Contrast with* member load.

data load rules. In Essbase, a set of operations that Essbase performs on data from an external data source file as the data is loaded.

data point. *See* cell.

data source. External data, such as a text file, spreadsheet file, or relational database, that is loaded into an Essbase database.

data value. *See* cell, metadata.

DateTime transformation. A set of instructions that defines how to change or reformat a relational database DateTime datatype to your choice of date format.

DBMS. *See* RDBMS.

dense dimension. In Essbase, a dimension likely to contain data for every combination of dimension members, for example, the Time dimension in Sample Basic. *Contrast with* sparse dimension.

descendant. In Essbase, any member below a parent.

detail member. *See* leaf member.

dimension. In Essbase, a data category. Sample Basic includes the dimensions Time, Accounts, Product, Market, and Scenario. In an Essbase database outline, the dimensions represent the highest consolidation level. *See also* outline.

dimension member. In Essbase, the highest member in a dimension branch. Dimension members define the dimension. For example, in Sample Basic, Time, Accounts, Product, Market, and Scenario are dimension members. *See also* leaf member.

dimension branch. In Hyperion Integration Server, a collection of dimension tables organized in a hierarchical structure, with one of the dimension tables joining directly to the fact table. A dimension branch defines a single, potential Essbase dimension in the metaoutline. *See also* dimension table.

dimension build rules. In Essbase, a set of operations similar to data load rules. But, instead of loading data, the dimension build rules modify the outline based on data in the external data source file. *See also* member load.

dimension table. In Hyperion Integration Server, a container in the OLAP model for one or more relational tables that define a potential Essbase dimension. When a dimension table joins to another dimension table, forming a dimension branch, the dimension is composed of the columns of all of the dimension tables in the dimension branch. *See also* dimension branch.

dimension type. In Essbase, a dimension tag that allows a dimension to use predefined functionality. For example, dimensions of the type Time have predefined calendar functionality for accounting.

In Hyperion Integration Server, attributes in the OLAP model and metaoutline that define which tables form the Essbase Accounts, Time and standard (default) dimensions.

duplicate member. In Hyperion Integration Server, the second occurrence of a member name in a data source. You can determine whether OLAP Builder ignores duplicate members or adds them as shared members. *See also* shared member.

Dynamic Calc And Store members. In Essbase, members that Essbase calculates when you first retrieve the values. Essbase then stores the calculated values in the database. Subsequent retrievals of Dynamic Calc And Store members do not require calculating. *See also* Dynamic Calc members.

Dynamic Calc members. In Essbase, members that Essbase calculates at retrieval time. Essbase discards calculated values after the retrieval request is complete. *See also* Dynamic Calc And Store members.

dynamic calculation. In Essbase, a calculation that occurs when you retrieve data on a member that has been tagged as Dynamic Calc or Dynamic Calc And Store. The member's values are calculated at retrieval time instead of being precalculated during batch calculation.

Essbase database. A repository of data within 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 calc scripts or data load rules. *See also* application.

Essbase outline. *See* outline.

Essbase server. A database server that locates and accesses multidimensional data.

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. In Essbase, a value or item in a data source file that is loaded into an Essbase database.

In relational databases, a space allocated for a particular item of information. Fields are the smallest units of information 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. In 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.

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 only products in the Product A product family. A filter contains one or more conditions. *See also* condition.

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, primary key.

formula. In Essbase, a combination of operators and functions as well as dimension names, member names, and numeric constants. Formulas are used to calculate relationships between any members of an Essbase database. @VAR(Actual, Budget); is an example of a formula.

generation. In Essbase, a layer of a hierarchical tree structure that defines member relationships in an Essbase database. Essbase numbers generations incrementally from the dimension member (generation 1) towards the leaf members.

generation name. In Essbase, a unique name describing a generation.

header record. One or more records at the top of a data source. Header records describe the contents of the data source.

hierarchy. In Essbase, a tree structure relationship between members in the database outline, beginning with the dimension member and going down to the leaf members.

In Hyperion Integration Server, a predefined tree-structure relationship for some types of dimensions in the OLAP model and all dimensions in the metaoutline. For example, you can create a Time dimension hierarchy that contains YEAR, QTR, and MONTH.

index. In Essbase, a method of retrieving Essbase data based on sparse dimensions. Also refers to the index files, collectively.

In relational databases, pointers that are logically arranged by the values of a key. Indexes optimize access to relational data.

index cache. In Essbase, a buffer in memory that holds index pages.

index entry. In Essbase, a pointer to an intersection of sparse dimensions. Each index entry points to a block on disk and locates a particular cell within the block by means of an offset.

index file. In Essbase, a file Essbase uses to store data retrieval information. It resides on disk and contains index pages.

index page. In Essbase, a subdivision of an index file containing entries that point to data blocks.

input block. In Essbase, a type of data block that has at least one loaded data value.

input data. In Essbase, any data that is loaded from a data source and is not generated by calculating the database.

Integration Server. The server component of the Hyperion Integration Server product family. The Integration Server uses the information stored in the OLAP Catalog to extract from the data source, the dimension names and member names needed to build the Essbase outline. When the Essbase outline is complete, Hyperion Integration Server extracts data from the data source, performs the operations specified in the metaoutline, and loads the data into the Essbase database.

integrity constraint. In relational databases, a rule stating that each row should have an entry for each required key column.

intersection level. In the Essbase Spreadsheet Add-in, an 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. *See also* cell.

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 using, for example, the Table Properties dialog box Physical Joins tab.

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. *See also* foreign key, primary key.

leaf member. In Essbase, a member that has no children. Also referred to as detail member, level 0 member, and leaf node. *See also* dimension member.

left pane. In the OLAP Builder main window, the area on the left that lets you view a list of dimensions previously defined in the OLAP model.

In the OLAP Architect main window, the area on the left that displays a list of the tables and views available in your source relational database. *See also* right pane.

level. In Essbase, a layer of a hierarchical tree structure that defines member relationships in an Essbase database. Essbase numbers levels incrementally from the leaf member (level 0) towards the dimension member.

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. In Essbase, a data block created for sparse member combinations in which all sparse members are leaf members.

level 0 member. *See* leaf member.

level name. In Essbase, a unique name describing a level, for example, Lev2.

linked reporting object (LRO). In Essbase, an external file linked to a data cell in an Essbase database. LROs can be cell notes or files.

load attributes. In Hyperion Integration Server, a set of rules that determine what actions OLAP Builder performs on member names and data as they are loaded.

load data. *See* data load.

load member. In Hyperion Integration Server, a member in a user-defined dimension into which Integration Server loads data. Only user-defined dimensions need load members. For all non user-defined dimensions, Integration Server knows how to load members and data into the Essbase database. *See also* member load.

logical column. In Hyperion Integration Server, a column created by manipulating the data in one or more physical columns. *See also* column. *Contrast with* *See also* 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.

MDDB. A multidimensional database, such as an Essbase database. *See also* multidimensional.

Measure dimension. In Hyperion Integration Server, a dimension that, by default, maps to the Accounts dimension in the Essbase outline. *See also* Accounts dimension, measures.

measures. In Hyperion Integration Server, the data values that you want to track, such as UNIT_PRICE or DISCOUNT. By default, 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 Essbase outline. *See also* Accounts dimension, Measure dimension.

member. In Essbase, a discrete component within a dimension. For example, in Sample Basic, Year, Profit, and Diet are members. Members can also be dimensions; for example, Year is a member as well as a dimension.

In Hyperion Integration Server, members in the metaoutline map to members in the Essbase outline. However, one member in the metaoutline can represent multiple members in the Essbase outline.

member combination. In Essbase, a list of member names used to specify a set of data at the intersection of two or more dimensions. Use the cross-dimensional operator -> (a hyphen followed by a right-angle bracket) to specify a member combination. For example, the actual sales data for the month of January in Sample Basic is Sales->Jan->Actual.

member consolidation. *See* consolidate.

member load. The process of adding new dimensions or members (without data) to an Essbase outline. *See also* load member. *Contrast with* data load.

metadata. The data that describes the data values within a database. For example, in Essbase, dimension names and member names are metadata.

metaoutline. In Hyperion Integration Server, a template containing the structure and rules for creating an Essbase outline from an OLAP model.

missing data. In Essbase, a data label (#MISSING) indicating that no data exists in the labeled location. The value labeled #MISSING is similar to a null value but not the same as zero.

model. *See* OLAP model.

Move mode. In Hyperion Integration Server, an OLAP Architect state in which you can pick up, move, and drop objects in the OLAP model. *See also* Add Joins mode.

multidimensional. A method of referencing data through three or more dimensions. An individual data value is the intersection of one member from each dimension. *See also* MDDB.

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.

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. Open Database Connectivity: standardized application programming interface (API) technology that allows applications to access multiple third-party databases.

OLAP. *See* Online Analytical Processing (OLAP).

OLAP Architect. In Hyperion Integration Server, a tool used to design and create an OLAP model based on a relational data source.

OLAP Builder. In Hyperion Integration Server, a tool used to create one or more metaoutlines from an OLAP model. The metaoutline determines the contents and structure of your final Essbase database. Using the metaoutline, you can create an Essbase outline and load data into it.

OLAP Catalog. In Hyperion Integration Server, a relational database containing metadata describing the nature, source, location, and type of data that you are pulling from the relational data source. The OLAP Catalog also generates the SQL statements and the information required to generate an Essbase database outline.

OLAP Command Interface. In Hyperion Integration Server, a command-line tool that you can use to perform common OLAP Builder operations on the Essbase outline and the data in the Essbase database. For example, you can use the `LOADDATA` command to load data.

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 method of multidimensional, multi-user, client-server computing for users who need to analyze consolidated enterprise data in real time. OLAP systems feature zooming, data pivoting, complex calculations, trend analyses, and modeling.

Online Transaction Processing (OLTP). OLTP applications are also commonly referred to as data capture, data entry, or data collection applications. OLTP applications allow an organization to capture the large amounts of data resulting from its daily activities but provide limited capability for reporting on the data.

outline. The structure that defines all elements of a database within Essbase. For example, an outline contains definitions of dimensions, members, and formulas.

page file. *See* data file.

parent. In Essbase, a member that has at least one child. *See also* child.

pass-through transformations. In Hyperion Integration Server, a feature that allows 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.

permissions. A user’s access rights to data.

physical column. In relational databases, a column that is stored in a relational database. *See also* column. *Contrast with* logical column.

physical table. In relational databases, 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.

predicate. *See* condition.

primary key. In relational databases, a column (key) that uniquely identifies a row, for example, Employee_ID. *See also* foreign key, key column.

query governor. In Hyperion Integration Server, a parameter that you set to control the time and size of the queries that OLAP Builder makes to the data source.

RDBMS. relational database management system: a system for accessing data in a relational database.

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 an 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* MDDB. *See also* database.

restructure. In Essbase, an operation to regenerate or rebuild the database index and, in some cases, the data files.

right pane. In the OLAP Builder main window, the area on the right, in which you build your metaoutline.

In the OLAP Architect main window, the area on the right, in which you build your OLAP model. *See also* left pane.

roll up. *See* consolidate.

root member. *See* dimension member.

row. *See* record. *Contrast with* column.

schema. In relational databases, a logical model that represents your relational data. *See also* star schema, metadata.

server. *See* Essbase server, Integration Server.

shared member. In Essbase, a member that shares storage space with another member of the same name. This member has an attribute that designates it as shared. Shared members prevent Essbase from making extra calculations on a member that appears in more than one location in the outline. *See also* duplicate member.

sibling. In Essbase, a child member that shares a common parent with another child member (for example, in Sample Basic East and West are siblings). *See also* child, parent.

sparse dimension. In Essbase, a dimension with a low probability that data exists for every combination of dimension members. For example, in Sample Basic, the Product and Market dimensions are sparse. *Contrast with* dense dimension.

SQL. Structured Query Language. A computer language used to access data in relational databases.

staging area. A database that you create to meet the needs of a specific application. A staging area is a snapshot of one or more RDBMSs.

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. In Essbase, a layer of the Essbase server that handles mapping of Essbase data to physical disk storage. The Storage Manager controls lock, index, data, allocation, and transaction management functions.

string. A sequence of characters treated as a unit.

supervisor. In Essbase, a 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. *See also* administrator, database administrator.

table. In relational databases, a form of data storage in which data is stored in rows and columns.

Time dimension. In Essbase, a dimension type that defines how often you collect and update data. You may tag only one dimension as Time. Essbase does not require a Time dimension.

In Hyperion Integration Server, an optional dimension you create as a container for time-related and date-related data.

transformation rules. In Hyperion Integration Server, a set of instructions that define how to change or reformat the member names and data you extract from the source relational database.

Two-Pass calculation. An Essbase attribute, used for recalculating an Accounts member, a Dynamic Calc member, or a Dynamic Calc And Store member, after an initial calculation.

unary operator. In Essbase, one of a group of mathematical indicators (+, -, *, /, %) that defines how Essbase consolidates 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. In Essbase, a type of data block created for sparse dimension member combinations, when at least one of the sparse dimension members is a parent.

user-defined attribute (UDA). In Essbase, a method you use to return, within calc scripts and reports, lists of members with particular characteristics. You can report on subsets of data values predefined in the database outline.

In Hyperion Integration Server, attributes you specify to create Essbase UDAs. You can explicitly add UDAs to the metaoutline, or you can derive UDAs from the source relational table. For example, a PRODUCT_ID table may contain a COLOR column from which you can obtain UDAs.

user-defined dimension. In Hyperion Integration Server, dimensions that you explicitly create in OLAP Builder instead of obtaining and building them from the relational data source.

user-defined member. In Hyperion Integration Server, members that you explicitly create in the OLAP Builder instead of obtaining and building them from the relational data source.

validation. In Essbase, a process of checking a rules file against the outline to make sure the rules file is valid.

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. *See also* logical table.

Index

- (dashes)
 - consolidation property, 6-14
- (hyphens)
 - in names in scripts and formulas, 3-10, 4-17
- (minus signs), reversing, 4-28

Symbols

- ! (exclamation points), in names in scripts and formulas, 3-10
- " (double quotation marks)
 - for terms in scripts and formulas, 3-11
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in scripts and formulas, 3-10
- #MISSING
 - aggregate filters and, 5-7
 - in Hyperion Essbase databases, 10-6
 - properties, time balance and, 6-19
 - reserved word, 10-4
 - skipping during calculations, 6-19
 - zeros, skip properties, 6-19
- \$\$\$UNIVERSE\$\$\$, reserved word, 10-4
- % (percent signs)
 - consolidation property, 6-14
 - Hyperion Essbase formulas and, 6-26
 - in names in scripts and formulas, 3-10
- () (parentheses)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- * (asterisks)
 - consolidation property, 6-14
 - in names in scripts and formulas, 3-10
- + (plus signs)
 - consolidation property, 6-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
 - reversing, 4-28
- . (periods)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- / (slashes)
 - consolidation property, 6-14
 - 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, 5-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- <= (less than or equal to), filter operator, 5-14
- <>(not equal to) filter operator, 5-14
- = (equal signs)
 - filter operator, 5-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- > (greater than signs)
 - filter operator, 5-14
 - in names in scripts and formulas, 3-10
- >= (greater than or equal to), filter operator, 5-14
- @ (at signs)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10

- @VAR function, 6-21
- @VARPER function, 6-21
- [] (brackets), in names in scripts and formulas, 3-10
- \ (backslashes)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- _ (underscores)
 - converting spaces to, 4-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- { } (braces)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- | (vertical bars)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- ~ (tilde), consolidation property, 6-14
- (dashes)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- ' (single quotation marks)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- , (commas)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - 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, xii
- accounts dimensions
 - dense dimension, 6-6
 - dimension type, 6-4
 - measure tag, 3-24
 - properties, shared members and, 4-6
 - two-pass calculation and, 6-13, 10-2
- Add New Intersection Levels dialog box, 9-10

- Add New Report Columns dialog box, 9-12
- After, 2-11
- aggregate filters, problems with, 10-6
- aggregation options, defined, 1-3
- aliases
 - assigning, 6-10
 - defined, 6-10
 - dimensions, setting, 6-10
 - members, setting, 6-10
 - reports and, 6-10
 - shared members and, 4-6
 - transformations and, 10-4
- alphabetizing members, 4-23
- applications
 - sample, xi
 - specifying, 7-9, 8-4
- ascending sort, Drill-Through reports and, 9-18
- assigning, aliases, 6-10
- asterisks (*)
 - consolidation property, 6-14
 - in names in scripts and formulas, 3-10
- at signs (@)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- at.exe, 10-5
- automatic loads, scheduling, 7-16, 8-11
- average, time balance property, 6-18

B

- backslashes (\)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- balance
 - beginning, 6-18
 - beginning and ending, 6-18
- Batch directory, scheduled loads and, 10-5
- batch processing, OLAP Command Interface, 1-9
- beginning balance, 6-18
- braces ({ })
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- brackets ([]), in names in scripts and formulas, 3-10
- budgeting data, tracking, 3-7

- build level
 - problems with, 10-6
 - setting, 4-25
- C**
- calc scripts
 - names with special characters, 3-10, 3-11
 - setting default, 7-12, 8-7
 - UDAs and, 6-23
- calculations
 - order of, 6-12
 - problems, 10-7
- case
 - case-sensitive names, 3-9
 - case-sensitive searches, 4-14
 - changing, 4-21
- characters
 - forbidden in a name, 3-9, 4-14, 4-17, 7-4
 - maximum in dimension names, 3-9
 - maximum in member names, 3-9
 - quoting in scripts and formulas, 3-10, 3-11
- children
 - attaching a name to a parent, 4-19
 - shared members and, 4-6
- chronological sorting, 4-15
- closing balance, 6-18
- collapsing, member branches, 2-12
- colons (:), in names in names in scripts and formulas, 3-10
- columns
 - adding, 9-12
 - capitalization, xi
 - changing for measures, 4-29
 - creating dimensions from, 3-12
 - creating metaoutlines from, 3-12
 - deleting, 9-27
 - Drill-Through and, 9-2
 - example of, 9-16
 - ordering, 9-16
 - valid for Drill-Through reports, 9-12
- command line, loading members from, 7-12
- commas (,)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- conditions
 - extra, problems, 10-3
 - measure filters and, 5-20
 - multiple, measure filters and, 5-20
 - multiple, member filters and, 5-15
 - selecting multiple, 5-15, 5-20
- 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, 2-30
- consolidation
 - defined, 1-3
 - drill down, 1-4
 - example, 1-3
 - members, 1-3
 - properties, list of, 6-14
 - properties, shared members and, 4-6
- container filters, 5-3
- converting, spaces to underscores, 4-16
- copying
 - Drill-Through reports, 9-5
 - metaoutlines, 2-27
- creating
 - calc scripts, 7-12, 8-7
 - dimensions from columns, 3-12
 - hierarchies, 3-11
 - 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, 3-24
 - measure filters, 5-7
 - measures, 3-11
 - measures from columns, 3-12
 - member filters, 5-6
 - member levels, 3-11
 - members, 3-7
 - metaoutline filters, 5-10
 - partitions, 6-1
 - row filters, 5-13, 5-17, 5-18, 5-22
 - UDAs, 6-24

- user-defined dimensions and members, 3-19
- user-defined members, 3-19
- curly braces. *See* braces
- customer comparisons, 1-2
- customizing, Drill-Through reports, 9-21

D

daily, member and data load, 7-17

dashes (-)

- consolidation property, 6-14
- in dimension and member level names, 3-9
- in dimension and member names, 4-14, 4-17
- in names in scripts and formulas, 3-10

data

- budget, tracking, 3-7
- checking, 3-31
- drill down, 1-4
- drill up, 1-4
- loading. *See* data loading
- missing, in sample databases, xi
- pivoting to change view, 1-4
- storage properties, 6-8

Data Load dialog box, 8-10

data loading

- build level, setting, 4-25
- calculation and, 10-7
- daily, 7-16, 8-11
- defined, 8-1
- deleting scheduled loads, 7-18, 7-21
- diagram, 7-3
- failing, 10-4
- Hyperion Essbase outlines, 7-1, 8-1
- immediately, 8-8
- load member and, 3-22
- members, new, 4-8
- metaoutline filters and, 5-4
- OLAP Command Interface and, 7-12
- other sources, 3-7
- partial load, 10-5, 10-7
- prerequisites for, 8-3
- problems, 10-3, 10-6
- rules file, 7-15, 8-10
- scheduling, 8-11
- scheduling problems, 10-5
- starting, 8-4

- stopping, 8-10, 10-4
- text file, 7-15, 8-10
- viewing scheduled loads, 7-18, 7-21
- weekly, 7-16, 8-11

data sources

- accessing, 2-4
- filtering data from, 5-1
- Hyperion Essbase, specifying, 7-9, 8-4
- names, OLAP Integration Server and, 9-22
- OLTP databases, 1-5
- sample, xi
- security for Drill-Through reports, 9-23
- spreadsheet files, 1-5
- SQL, 1-5
- text files, 1-5
- tuning, 3-31
- username, Drill-Through reports and, 9-23
- workflow for transferring, 1-9

data values

- See also* measures
- defined, 3-6
- extracting from a relational data source, 7-3
- filtering, 5-7
- manipulating, 4-27
- multiple in measure filters, 5-20
- reversing signs, 4-28
- scaling, 4-27
- selecting multiple in filters, 5-15

databases. *See* data sources

dataload.txt, 8-10, 10-5, 10-7

date format, 7-17, 8-12

decimal points, moving, 4-27

default calc script, setting, 7-12, 8-7

Default metaoutline filter, 5-4

Delete an OLAP Model/Metaoutline dialog box, 2-29

Delete Locks command, 2-30

deleting

- dimensions, 3-29
- dimensions during loading, 7-9
- measures, 3-29, 3-33
- member levels, 3-29
- member levels from hierarchies, 3-17
- members during loading, 7-9
- metaoutline filters, 5-12
- metaoutlines, confirming, 2-29
- metaoutlines, open by other users, 2-30

- metaoutlines, procedures, 2-28
- OLAP models, open by other users, 2-26
- problems with, 10-2
- scheduled loads, 7-18, 7-21
- dense dimensions
 - accounts, 6-6
 - defined, 6-6
 - time, 6-6
- descending sort, Drill-Through reports and, 9-18
- descriptions, of metaoutlines, 2-25
- detail data
 - Drill-Through, 9-2
 - retrieving, 1-4
- dialog boxes
 - Add New Intersection Levels, 9-10
 - Add New Report Columns, 9-12
 - Data Load, 8-10
 - Delete an OLAP Model/Metaoutline, 2-29
 - Dimension Properties, 4-7, 4-9, 6-6, 6-7, 6-9, 6-17, 6-24
 - Drill-Through Reports, 9-3
 - Edit Hierarchy, 3-5
 - Edit OLAP Metaoutline Transformations, 4-13, 4-18, 4-21
 - Enter the new dimension's name, 3-20
 - Enter the new filter's name, 5-10
 - Essbase Application and Database, 7-8, 8-4
 - Essbase Users, 7-10, 8-5
 - Incremental Update, 7-11, 8-6
 - Measure Properties, 3-28, 4-27, 4-28, 4-30, 5-19, 5-22
 - Member Load, 7-14
 - Member Load Filtering, 4-24, 5-13, 5-17
 - Member Properties, 3-26, 6-11, 6-13, 6-15, 6-17, 6-20, 6-22, 6-24, 6-26
 - OLAP Metaoutline Properties, 3-22, 3-32, 7-6, 9-14
 - OLAP Model Properties, 2-26
 - Order By, 9-20
 - Preview Filter Results, 3-31
 - Preview Transformation Results, 3-31, 4-15
 - Report Properties, 9-14, 9-26
 - Sample Outline, 3-31
 - Save New Metaoutline, 2-24
 - Schedule Essbase Load, 7-13, 7-16, 8-8, 8-11
 - Scheduled Jobs, 7-19, 7-21
 - Select a Member, 3-23
 - Select Values From, 5-15, 5-20
 - Services, 7-18
 - Set Sort Order, 9-19
 - View Existing Job, 7-20
 - View Filter, 5-12
- dimension building. *See* member loading
- dimension names, maximum length, 3-9
- Dimension Properties dialog box
 - Account Info tab, 6-17
 - Member Info tab, 6-6, 6-7, 6-9
 - Outline Build tab, 4-7, 4-9
 - UDA tab, 6-24
- dimension storage, specifying, 6-6
- dimension types
 - accounts, 6-4
 - standard, 6-4
 - time, 6-5
- dimensions
 - adding, 3-1
 - aliases, setting, 6-10
 - creating, 3-12
 - creating extra, 3-7
 - defined, 1-3, 1-11, 3-3
 - deleting, 3-29
 - deleting during load, 7-9
 - dense, 6-6
 - Drill-Through reports, 9-7
 - examples, 1-3
 - extracting from a relational data source, 7-3
 - formulas on, 6-25
 - measures, creating, 3-24
 - members and, 1-3
 - minimum number in metaoutline, 2-20
 - naming rules, 3-9
 - OLAP model, not associated with, 3-7
 - ordering, 3-12
 - rejected, 7-15, 8-10, 10-5, 10-7
 - renaming, 3-25
 - sparse, 6-6
 - TBC Metaoutline and, 1-12
 - two-pass calculations, specifying, 6-12
 - types of, 6-3
 - use in consolidation, 1-3
 - user-defined, 3-7, 3-19
 - viewing sample, 3-30

directories
 default, 7-15, 8-10
 default metaoutline, 2-3
 is, 2-3
 ISADMIN, 7-15, 8-10

dividing, values while loading, 4-27

documentation, xi, xii

double quotation marks ("")
 in dimension and member level names, 3-9
 in dimension and member names, 4-14, 4-17
 terms in scripts and formulas, 3-10, 3-11

Drill, 9-12

drill down, 1-4

drill up, 1-4

Drill-Through columns
 adding, 9-12
 deleting, 9-27
 deleting all, 9-27
 example of, 9-16
 ordering, 9-16

Drill-Through reports
 copying, 9-5
 creating, 9-1, 9-3
 customizing, 9-21
 database username and, 9-23
 deleting, 9-27
 deleting metaoutlines and, 2-29, 10-2
 load balancing, 9-22
 missing, 10-2
 OLAP Integration Server name, 9-22
 renaming, 9-25, 9-26
 rows, sort order of, 9-18
 security and, 9-23
 valid columns for, 9-12

Drill-Through Reports dialog box, 9-3

Drill-Through, defined, 9-1

Drop leading/trailing spaces option, 4-16

dummy members, creating, 4-8

duplicate members
 defined, 4-3
 handling, 4-6
 ignoring, 4-3
 problems with, 10-6

dynamic calc and store, property, 6-8

dynamic calc, property, 6-8

Dynamic Time Series

setting, 6-16

dynamic time series
 problems with, 10-2

E

Edit Hierarchy dialog box, 3-5

Edit menu
 Add Member as Child menu command, 3-20
 Add Member as Sibling menu command, 3-21
 Drill-Through Reports command, 9-3
 Filter command, 4-23, 5-13, 5-17, 5-18, 5-22
 Transformation command, 4-13

Edit OLAP Metaoutline Transformations dialog box,
 4-13
 Prefix/Suffix tab, 4-18
 Replace tab, 4-21

ending balance, 6-18

Enter the new dimension's name dialog box, 3-20

Enter the new filter's name dialog box, 5-10

equal signs (=)
 filter operator, 5-14
 in dimension and member level names, 3-9
 in dimension and member names, 4-14, 4-17
 in names 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 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, 6-21
 variance reporting property, 6-21

extra, dimensions and members, 3-7

F

fields, adding prefixes or suffixes to, 4-17

File menu

- Delete command, 2-29
- New command, 2-4
- Open command, 2-4
- Print command, 2-28
- Save As command, 2-23, 2-27
- Validate command, 2-19
- Filter command, 5-13, 5-17, 5-18, 5-22
- filtering
 - data sources, 5-1
 - data values, 5-7
 - members, 5-6
- filters
 - See also* metaoutline filters
 - conditions, creating multiple, 5-10
 - conditions, extra, 10-3
 - defined, 5-1
 - grouping, 5-4
 - Hyperion Essbase databases, creating multiple, 5-4
 - measure, example, 5-7
 - member, example, 5-3
 - metaoutline and data loading, 5-3
 - naming rules, 3-9
 - operators in, 5-14
 - problems with, 10-6
 - row, creating, 5-13
 - row, defined, 5-6
 - shared members, problems and, 10-6
 - SQL, 5-3
 - viewing current metaoutline, 5-8
 - viewing in a metaoutline, 5-11
- flipping, plus and minus signs, 4-28
- forecast data, tracking, 3-7
- forecasting, 1-4
- formulas
 - dimensions and, 6-25
 - names with special characters, 3-10, 3-11
 - problems with, 10-2
 - setting, 6-25
 - shared members and, 4-6
 - user-defined members and, 6-25
- From Database Column, consolidation property, 6-14

G

- generated SQL, viewing, 3-31

- greater than or equal to (\geq), filter operator, 5-14
- greater than signs ($>$)
 - filter operator, 5-14
 - in names in names in scripts and formulas, 3-10

H

- Help menu
 - Help Topics command, xii
- hierarchies
 - consolidation and, 1-3
 - creating dimensions from, 3-12
 - creating, example, 3-15
 - defined, 1-11
 - deleting member levels from, 3-17, 3-29
- 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, 5-4
 - databases, specifying, 7-9, 8-4
 - databases, workflow for creating, 1-9
 - dimensions, defined, 3-3
 - 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, 6-27
 - viewing properties in columnar format, 2-14, 6-27, 6-28
 - viewing properties in condensed format, 6-27, 6-29
 - what if scenarios, 1-4
- Hyperion Essbase outlines
 - building, x, 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, 3-30
 - Hyperion Essbase server, uses, 1-3
 - Hyperion Integration Server
 - components, 1-8
 - defined, 1-5
 - OLAP Command Interface, defined, 1-9
 - OLAP Metadata Catalog, defined, 1-9
 - OLAP Model, defined, 1-8
 - product family, defined, 1-5
 - product family, diagram, 1-7
 - 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, 4-14, 4-17
 - 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, 5-14
 - incremental member load, 7-11, 8-6
 - Incremental Update dialog box, 7-11, 8-6
 - intersection levels
 - adding, 9-6, 9-10
 - defined, 9-2
 - deleting, 9-27
 - deleting all, 9-27
 - 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, 7-19, 7-21
- L**
- label only property, 6-8
 - last time balance property, 6-18
 - leaf members, setting, 4-26
 - leaf node, example, rejected dimensions or members, 7-15, 8-10
 - left frame
 - defined, 2-12
 - selecting multiple objects, 2-15
 - less than or equal to (<=), filter operator>, 5-14
 - less than signs (<)
 - filter operator>, 5-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
 - levels
 - Drill-Through reports and, 9-7
 - level 0 members, setting, 4-26
 - setting build, 4-25
 - linked partitions, 6-1, 6-3
 - load balancing, Drill-Through reports and, 9-22
 - load members
 - setting, 3-22
 - user-defined dimensions, 3-8
 - validation and, 2-21
 - load properties
 - build level, 4-25
 - columns, changing for measures, 4-29
 - defined, 4-1
 - dimensions, setting, 4-3
 - duplicate member names, 4-3
 - measures, manipulating, 4-27
 - measures, setting, 4-26
 - member names
 - converting spaces to underscores, 4-16
 - determining capitalization, 4-21
 - dropping spaces, 4-16
 - prefixes and suffixes, 4-17

- replacing, 4-13
- setting, 4-11
- members, new, 4-8
- reversing plus and minus signs, 4-28
- shared members, 4-6
- sorting members, 4-23
- LOADDATA command, 8-7
- loading
 - build level, setting, 4-25
 - daily, 7-16, 8-11
 - defined, 8-1
 - deleting scheduled loads, 7-18, 7-21
 - Hyperion Essbase outlines, 8-1
 - immediate data load, 8-8
 - immediate member load, 7-13
 - load members and, 3-22
 - metaoutline filters and, 5-4
 - OLAP Command Interface and, 7-12
 - other sources, 3-7
 - scheduling, 7-16
 - scheduling data loads, 8-11
 - viewing scheduled loads, 7-21
 - weekly, 7-16, 8-11
- loading process, 7-3
- LOADMEMBER command
 - example of, 7-12
- loads, viewing scheduled, 7-19
- locking, Hyperion Essbase outlines and, 10-4

M

- Match Whole Word option, in a replace operation, 4-14
- maximum rows per query, setting, 7-6, 9-15
- maximum time per query, setting, 7-6, 9-15
- measure dimension, creating, 3-24
- measure filters
 - See also* filters
 - conditions and, 5-20
 - creating, 5-7
 - metaoutline filters and, 5-4
 - operators in, 5-14
 - typing directly, 5-23
- Measure Properties dialog box
 - Filters tab, 5-19, 5-22
 - General tab, 3-28, 4-30
- Transformations tab, 4-27, 4-28
- measure tag, 2-20, 3-24
- measures
 - adding, 3-1
 - column, changing, 4-29
 - creating dimensions from, 3-12
 - defined, 1-11, 3-6
 - deleting, 3-29, 3-33
 - Drill-Through reports and, 9-7
 - editing, 3-33
 - filtering, 5-7
 - maximum allowed, 2-20
 - metaoutlines, 3-14
 - ordering, 3-16
 - renaming, 3-25
 - renaming a single, 3-27
 - required number in metaoutline, 2-20
 - user-defined members and, 3-7
 - viewing, 3-32
- member filters
 - See also* filters
 - creating, 5-6
 - metaoutline filters and, 5-4
 - operators in, 5-14
 - typing directly, 5-18
- member levels
 - adding, 3-1
 - defined, 1-11, 3-5
 - example of, 3-5
 - ordering, 3-16
 - renaming, 3-25
- Member Load dialog box, 7-14
- Member Load Filtering dialog box
 - Filters tab, 5-13, 5-17
 - Ordering tab, 4-24
- member load, stopping, 7-15
- member loading
 - build level, setting, 4-25
 - command line, 7-12
 - defined, 7-1
 - deleting scheduled loads, 7-18, 7-21
 - failing, 10-4
 - Hyperion Essbase outline, 8-1
 - immediately, 7-13
 - improving performance of, 7-11, 8-6
 - incomplete, 10-6

- incrementally, 7-11, 8-6
- metaoutline filters and, 5-4
- new members, 4-8
- partial load, 10-5, 10-7
- prerequisites for, 7-4
- problems, 10-3, 10-6
- rules file, 7-15, 8-10
- scheduling, 7-16
- scheduling problems, 10-5
- sorting, performance and, 4-24
- starting, 7-8
- stopping, 8-10, 10-4
- text file, 7-15, 8-10
- user-defined load member, 3-8
- viewing scheduled loads, 7-18, 7-21
- member names, maximum length, 3-9
- Member Properties dialog box
 - Account Info tab, 6-20, 6-22
 - Formula tab, 6-26
 - General tab, 3-26
 - Member Info tab, 6-11, 6-13, 6-15, 6-17
 - UDAs tab, 6-24
- members
 - aliases, setting, 6-10
 - alphabetizing, 4-23
 - branches, collapsing or expanding, 2-12
 - changing case of, 4-12, 4-21
 - checking before loading, 3-31
 - consolidation, 1-3
 - consolidation, operators, 6-14
 - 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, 3-9
 - OLAP model, not associated with, 3-7
 - ordering, 4-23
 - prefixes and, 4-17
 - rejected, 7-15, 8-10, 10-5, 10-7
 - sample, problems with, 10-4
 - shared. *See* shared members
 - sorting, 4-23
 - suffixes and, 4-17
 - transforming, 4-11
 - two-pass calculation, 6-12
 - user-defined, 3-19
 - viewing sample, 3-30
- menu bar, defined, 2-13
- menus, right-click, 2-13
- metadata repository. *See* OLAP Catalog
- 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, 5-4
 - defined, 5-1, 5-3
 - deleting, 5-12
 - measure filters and, 5-3
 - member filters and, 5-3
 - multiple conditions in, 5-10
 - naming rules, 3-9
 - problems with, 10-3
 - setting current, 5-8
 - viewing the contents of, 5-11
- Metaoutline main window
 - customizing view, 2-13, 6-27
 - 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
- metaoutlines
 - advantages of, 1-10
 - components of, 1-11
 - confirming a deletion, 2-29
 - copying, 2-27
 - creating, 2-11
 - defined, 1-8
 - deleting, 2-28
 - deleting, problems with, 10-2

- description of, 2-25
- filters, viewing, 5-11
- Hyperion Essbase databases, creating multiple, 5-4
- Hyperion Integration Server sample, xi
- introducing, 1-1
- locks, deleting, 2-30
- measures in, viewing, 3-32
- navigating, 2-12
- opening, 2-4
- populating, 3-1
- prerequisites for, 1-1
- printing, 2-28
- renaming, 2-27
- sample, xi
- 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, 6-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10
 - reversing, 4-28
- Miscellaneous members, 4-8
- missing
 - See also #MISSING*
- models. *See* OLAP models
- monthly, member and data load, 7-17
- months, sorting chronologically, 4-15
- 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, 4-27

N

- naming conventions, case sensitivity, 3-9
- naming rules
 - applications and databases, 7-4
 - members, 3-9
 - metaoutlines, 3-9
- navigating metaoutlines, 2-12
- negative data values, reversing signs, 4-28
- never share property, 6-8
- non expense, variance reporting property, 6-21
- not equal to filter operator (<>), 5-14
- not in filter operator, 5-14

O

- ODBC
 - data sources, multiple OLAP Integration Servers and, 9-22
 - 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 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, 9-22
 - ODBC data source names, 9-22
- OLAP intersection levels
 - adding, 9-6, 9-10
 - defined, 9-2
 - deleting, 9-27
 - deleting all, 9-27
 - 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, 2-30
- OLAP Metaoutline Assistant, 1-8, 2-7
- OLAP Metaoutline Properties dialog box
 - Database Measures tab, 3-32
 - Drill-Through Reports tab, 9-23, 9-24
 - Governors tab, 7-6, 9-14
 - User Defined Dimensions tab, 3-22
- OLAP metaoutline. *See* metaoutlines
- OLAP Model
 - defined, 1-8
 - startup programs necessary, 2-3
 - uses, 1-1
- OLAP Model Assistant, 1-7
- OLAP Model Properties dialog box
 - General tab, 2-26
- OLAP models
 - dimensions associated with, 3-12
 - dimensions not associated with, 3-7, 3-19
 - members, 1-3
 - members associated with, 3-15
 - members not associated with, 3-7, 3-19
 - missing, 10-1
 - sample, xi
 - specifying user access permissions, 2-25
- OLAP Server, defined, 1-2
- olapblldr program, 2-3
- OLTP data, defined, 1-5
- Online Analytical Processing (OLAP), defined, 1-2
- online guide, xii
- online help, xii
- Online Transaction Processing (OLTP), examples of
 - contents, 1-5
- Open Database Connectivity. *See* ODBC
- opening balance, 6-18
- operational data, 1-5
- operations, order of, 4-12
- operators, 6-14
 - consolidation, 6-14
 - member and measure filters, 5-14
- order
 - calculations and, 6-12
 - transformations performed, 4-12
- Order By dialog box, 9-20
- ordering
 - columns in Drill-Through reports, 9-16
 - dimensions, 3-12

- measures, 3-16
- member levels, 3-16
- members, 4-23
 - rows in Drill-Through reports, 9-18
- Other, reserved word, 10-4
- 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, 4-14, 4-17
 - in names in scripts and formulas, 3-10
- parents
 - attaching name to children, 4-19
 - rolling up into different, 4-4
 - two-pass calculation and, 6-12
- partitions, 6-1, 6-3
- percent signs (%)
 - consolidation property, 6-14
 - Hyperion Essbase formulas, 6-26
 - in names in scripts and formulas, 3-10
- performance
 - build level and, 4-25
 - Drill-Through reports and, 9-22
 - sorting and, 4-24
 - tuning, data sources, 3-31
- periods (.)
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- permissions
 - changing, 2-27
 - saving metaoutlines and, 2-25
- pivoting
 - defined, 1-4
 - using to change data view, 1-4
- plus signs (+)
 - consolidation property, 6-14
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - in names in scripts and formulas, 3-10

- reversing, 4-28
- populating, metaoutlines, 3-1, 7-1, 8-1
- pop-up menu, defined, 2-13
- positive data values, reversing sign, 4-28
- prefixes
 - adding to fields, 4-17
 - example, 4-12
 - new member names and, 4-10
 - operations, order of, 4-12
- prerequisites
 - data, loading, 8-3
 - metaoutlines, creating, 7-4
- Preview Filter Results dialog box, 3-31
- Preview Transformation Results dialog box, 3-31, 4-15
- Print button, Metaoutline toolbar, 2-28
- printing
 - metaoutlines, 2-28
 - SQL, 3-31
- 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, 6-17
 - consolidation, 6-14
 - dimension storage, 6-6
 - dynamic calc, 6-8
 - dynamic calc and store, 6-8
 - expense, setting, 6-21
 - first time balance, 6-18
 - Hyperion Essbase, 6-1
 - label only, 6-8
 - last time balance, 6-18
 - load. *See* load properties
 - never share, 6-8
 - skip, 6-19
 - time balance, 6-18
 - two-pass calculation, 6-12
 - variance reporting, 6-21
 - zeros, 6-19

Q

- query governors
 - defined, 7-5, 9-14
 - problems with, 10-4
 - setting, 7-5, 9-14
- query size, controlling, 7-5, 9-14
- query time, controlling, 7-5, 9-14
- quotation marks
 - in dimension and member level names, 3-9
 - in dimension and member names, 3-9, 4-14, 4-17
 - in scripts and formulas, 3-10, 3-11

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
 - dimensions, 3-25
 - Drill-Through reports, 9-25, 9-26
 - measures, 3-25
 - member levels, 3-25
 - metaoutlines, 2-27
 - single measure, 3-27
- repeated member names. *See* duplicate members
- replace operations, order of, 4-12
- replacing case-sensitive strings, 4-14
- replicated partitions, 6-1, 6-3
- Report Properties dialog box
 - General tab, 9-26
 - Governors tab, 9-14
- report scripts
 - names with special characters, 3-10, 3-11
 - user-defined attributes and, 6-23
- reports
 - aliases and, 6-10
 - copying, 9-5
 - creating, 9-1, 9-3
 - customizing, 9-21
 - database username and, 9-23

- deleting, 9-27
 - Drill-Through, missing, 10-2
 - load balancing, 9-22
 - OLAP Integration Server name, 9-22
 - renaming, 9-25, 9-26
 - security and, 9-23
 - valid columns for, 9-12
 - repository. *See* OLAP Metadata Catalog
 - reserved words, 3-9
 - restructure, incremental, 7-11, 8-6
 - result sets, controlling size of, 7-5, 9-14
 - reversing, plus and minus signs, 4-28
 - right frame
 - defined, 2-12
 - selecting multiple objects, 2-16, 2-17
 - right-click menus, xiv, 2-13
 - roll-ups
 - average property and, 6-18
 - defined, 1-3
 - last property and, 6-18
 - row filters
 - creating, 5-13, 5-17, 5-18, 5-22
 - defined, 5-6
 - rows
 - example of, 9-18
 - maximum in query, setting, 7-5, 9-14
 - none returned, 7-6
 - ordering, 9-18
 - rules
 - intersection levels and, 9-7
 - naming, 3-9
 - naming applications and databases, 7-4
 - shared members, 4-6
 - rules file, 7-15, 8-10
- S**
- sample
 - applications, xi
 - Hyperion Essbase outlines, problems with, 10-4
 - Hyperion Essbase outlines, viewing, 3-30
 - members, viewing, 3-30
 - Sample command, 3-30
 - Sample Outline dialog box, 3-31
 - Save New Metaoutline dialog box, 2-24
 - saving
 - metaoutlines, using a different name, 2-27
 - metaoutlines, using File > Save As, 2-27
 - problems with, 10-2
 - scaling, values, 4-27
 - scenarios, 1-4
 - Schedule Essbase Load dialog box, 7-13, 7-16, 8-8, 8-11
 - schedule service, starting, 7-18
 - Scheduled Jobs dialog box, 7-19, 7-21
 - scheduled loads
 - problems with, 10-5
 - unable to schedule, 10-5
 - viewing and deleting, 7-18, 7-21
 - scheduling
 - data loads, 8-11
 - member loads, 7-16
 - problems with, 10-5
 - unable to complete, 10-5
 - Windows NT schedule service, 7-18
 - scripts
 - calc scripts, 7-12, 8-7
 - searches, case-sensitive, 4-14
 - security
 - Drill-Through reports and, 9-23
 - Hyperion Essbase users, creating, 8-5
 - Select a Member dialog box, 3-23
 - Select Values From dialog box, 5-15, 5-20
 - selecting
 - calc scripts, 7-12, 8-7
 - multiple conditions, 5-15, 5-20
 - multiple objects, 2-15, 2-16
 - semicolons (;)
 - in names in scripts and formulas, 3-10
 - servers, Hyperion Essbase users and, 8-5
 - Services dialog box, 7-18
 - Set Sort Order dialog box, 9-19
 - setting
 - aliases, 6-10
 - build level, 4-25
 - calc scripts, 7-12, 8-7
 - expense property, 6-21
 - formulas, 6-25
 - Hyperion Essbase properties, 6-1
 - load member, 3-22
 - metaoutline filters, 5-8
 - OLAP intersection levels, 9-7

- query governors, 7-5, 9-14
 - Settings menu
 - Control Panels command, 7-18
 - 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, 6-23
 - signs, reversing plus and minus, 4-28
 - single measure
 - renaming, 3-27
 - single quotation marks (')
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
 - size, query, controlling, 7-5, 9-14
 - skip property, 6-19
 - slashes (/)
 - consolidation property, 6-14
 - in names in scripts and formulas, 3-10
 - sorting
 - Drill-Through reports and, 9-18
 - members, 4-23
 - months, chronologically, 4-15
 - performance and, 4-24
 - source data, multidimensional databases, 1-5
 - sources. *See* data sources
 - spaces
 - converting to underscores, 4-16
 - dropping, 4-16
 - dropping and converting, order of, 4-12
 - in dimension and member names, 3-9
 - sparse dimensions, defined, 6-6
 - Spreadsheet Add-in, drilling through from, 9-1
 - spreadsheet files, data sources, 1-5
 - spreadsheet users, customizing Drill-Through reports, 9-21
 - SQL
 - checking, 3-31
 - data sources, 1-5
 - filters, 5-1, 5-3
 - printing, 3-31
 - queries, controlling the size of, 7-5, 9-14
 - syntax, filter errors, 10-3
 - versions supported, 1-5
 - viewing for a dimension, 3-31
 - 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, 6-8
 - dimensions, specifying, 6-6
 - store properties, 6-8
 - strategic planning, 1-4
 - SUBSTRING function, 10-6
 - suffixes
 - adding to fields, 4-17
 - new members and, 4-10
 - operations, order of, 4-12
 - syntax
 - filters, errors, 10-3
 - formulas, incorrect, 10-2
- ## T
- tab, in names, 3-9
 - tables
 - capitalization, xi
 - tagging
 - accounts dimension, 6-4
 - measure dimensions, 3-24
 - time dimension, 6-5
 - TBC Metaoutline, xi
 - dimensions in, 3-4
 - measures in, 3-6
 - parts of, 1-12
 - TBC Model, xi
 - templates. *See* metaoutlines
 - text
 - dataload.txt, 8-10, 10-5, 10-7
 - files, data sources, 1-5
 - tilde, consolidation property (~), 6-14
 - time
 - format, 7-17, 8-12
 - query, controlling, 7-5, 9-14

- time-out errors, 10-4
- time balance
 - average, 6-18
 - first, 6-18
 - last, 6-18
 - missing properties, 6-19
 - properties, specifying, 6-18
 - skip, 6-19
- time balance properties, 6-18
- time dimension
 - dense dimension, 6-6
 - dimension type, 6-5
- time-out errors, 10-4
- timing, specifying for loads, 7-16, 8-11
- toolbar, defined, 2-13
- Tools menu, Scheduler command, 7-19, 7-21
- tracking, budget data, 3-7
- Transformation command, 4-13
- transformations
 - aliases and, 10-4
 - order of, 4-12
- transforming member names, 4-11
- transparent partitions, 6-1, 6-3
- tree editor, collapsing or expanding a member
 - branch, 2-12
- troubleshooting
 - 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, 3-31
 - viewing SQL, 3-31
- two-pass calculations, specifying, 6-12
- types of dimensions, 6-3

U

UDAs

- calc scripts, 6-23
- creating, 6-24
- defined, 6-23
- report scripts, 6-23

- shared members and, 4-6, 6-23
- unary operators, 6-14
- underscores (_)
 - converting spaces to, 4-16
 - in dimension and member level names, 3-9
 - in dimension and member names, 4-14, 4-17
- UNIVERSE reserved word, 10-4
- unscheduled data loads, 8-8
- unscheduled member loads, 7-13
- updating
 - data. *See also* data loading
 - Hyperion Essbase outlines incrementally, 7-11, 8-6
 - incrementally, 8-6
 - members. *See also* member loading
- user names, specifying, 7-10, 8-5
- user privileges. *See* permissions
- user-defined dimensions
 - creating, 3-19
 - defined, 3-7
 - example, 3-7
 - load member for, 3-22
 - valid, 2-21
- user-defined members
 - creating, 3-19
 - defined, 3-7
 - formulas on, 6-25
- users, Hyperion Essbase, creating, 7-10, 8-5

V

validation

- metaoutlines, 2-19
- problems, 10-1

values. *See* data values

variance reporting properties, 6-21

vertical bars (|)

- in dimension and member level names, 3-9
- in dimension and member names, 4-14, 4-17

View Existing Job dialog box, 7-20

View Filter dialog box, 5-12

View menu

- Display Essbase Properties command, 2-14, 6-27
- Metaoutline icons command, 2-14
- Sample command, 3-30
- Status bar command, 2-13

- Toolbar command, 2-13
- viewing
 - columnar format, 6-27
 - condensed format, 6-27
 - displaying Hyperion Essbase properties, 6-27
 - filters, 5-11
 - generated SQL, 3-31
 - Hyperion Essbase properties in columnar format, 6-28
 - Hyperion Essbase properties in condensed format, 6-29
 - measures, 3-32
 - sample dimensions, members, outlines, 3-30
 - sample dimensions, members, outlines, problems with, 10-4
 - scheduled loads, 7-18

W

- weekly, member and data load, 7-17
- 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, 7-18
- 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, 5-4

Z

- zeros
 - property, 6-19
 - skipping during calculation, 6-19

