

ORACLE® DATA INTEGRATOR KNOWLEDGE MODULE
FOR ESSBASE

RELEASE 10.1.3.5.5

USER'S GUIDE

ORACLE®
ENTERPRISE PERFORMANCE
MANAGEMENT SYSTEM

ODI Knowledge Module for Essbase User's Guide, 10.1.3.5.5

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

Authors: Data Integration Management writing team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited. The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS:

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Chapter 1. Introduction to Oracle Data Integrator Adapter for Essbase	5
Purpose	5
Integration Process	5
Working with the Adapter	6
Chapter 2. Setting Up Environments	7
Defining Data Servers	7
Defining Physical and Logical Schemas and a Context	9
Chapter 3. Reverse-Engineering Essbase Applications	13
What Reverse-Engineering Does	13
Using the Essbase Reverse-engineering Knowledge Module	13
Chapter 4. Load and Extract	21
Data Integration Tasks	21
Data Extraction Using Essbase-supported Querying and Scripting	21
Data Extraction using Report Scripts	22
Data Extraction Using MDX Queries	22
Data Extraction Using Calculation Scripts	23
Creating Interfaces	23
Loading Metadata	23
IKM SQL to Hyperion Essbase (METADATA)	24
Loading Data	26
IKM SQL to Hyperion Essbase (DATA)	26
Extracting Data	30
LKM Hyperion Essbase DATA to SQL	31
Extracting Members from Metadata	33
LKM Hyperion Essbase METADATA to SQL	33

1

Introduction to Oracle Data Integrator Adapter for Essbase

In This Chapter

Purpose	5
Integration Process	5
Working with the Adapter	6

Purpose

Oracle® Data Integrator Adapter for Oracle Essbase enables you to connect and integrate Essbase with virtually any source or target using Oracle Data Integrator. The adapter provides a set of Oracle Data Integrator Knowledge Modules (KMs) for loading and extracting metadata and data and calculating data in Essbase applications.

The documentation for the Oracle® Data Integrator Adapter for Hyperion Essbase is based on the assumption that users have previous experience with Oracle Data Integrator and have full access to the documentation. The latest documentation for Oracle Data Integrator is available for download from the Documentation area of the Oracle Technical Network (OTN) Web site (<http://www.oracle.com/technology/index.html>).

Integration Process

Note: See OTN for the most up-to-date documentation for Oracle Data Integrator.

You can use Oracle Data Integrator Adapter for Essbase to perform these data integration tasks on an Essbase application:

- Load metadata and data
- Extract metadata and data

Using the adapter to load or extract metadata or data involves these tasks:

- Setting up an environment: Importing the Essbase technology and defining data servers and schemas

See [Chapter 2, “Setting Up Environments.”](#)

- Reverse-engineering an Essbase application using the Reverse-engineering Knowledge Module (RKM)

See [Chapter 3, “Reverse-Engineering Essbase Applications.”](#)

- Extracting metadata and data using Load Knowledge Modules (LKM)
- Integrating the metadata and data into the Essbase application using the Integration Knowledge Modules (IKM)

Working with the Adapter

Using Oracle Data Integrator Adapter for Essbase involves these Oracle Data Integrator features:

- Topology Manager—For defining connections to the Essbase applications
See [Chapter 2, “Setting Up Environments.”](#)
- Designer—For these tasks:
 - Loading metadata and data into datastores, which are target tables that represent Essbase dimensions and data tables
 - Extracting metadata and data from datastores, which are source tables that represent Essbase metadata and data

Oracle Data Integrator Adapter for Essbase includes the Essbase RKM, which creates the datastores.

In Designer, you use the Essbase RKM to create the datastores.

The adapter includes these other knowledge modules (KMs) for loading and extracting data:

- KM_IKM SQL to Hyperion Essbase (METADATA)—Loads metadata to an application from the staging area. See [“IKM SQL to Hyperion Essbase \(METADATA\)” on page 24](#) and [“Loading Metadata” on page 23](#).
- KM_IKM SQL to Hyperion Essbase (DATA)—Loads data to an application from the staging area. See [“IKM SQL to Hyperion Essbase \(DATA\)” on page 26](#) and [“Loading Data” on page 26](#).
- KM_LKM Hyperion Essbase DATA to SQL—Extracts data from an Essbase application to the staging area. See [“LKM Hyperion Essbase DATA to SQL” on page 31](#) and [“Extracting Data” on page 30](#).
- KM_LKM Hyperion Essbase METADATA to SQL—Extracts dimension of an Essbase application to the staging area. See [“LKM Hyperion Essbase METADATA to SQL” on page 33](#) and [“Extracting Members from Metadata” on page 33](#).

2

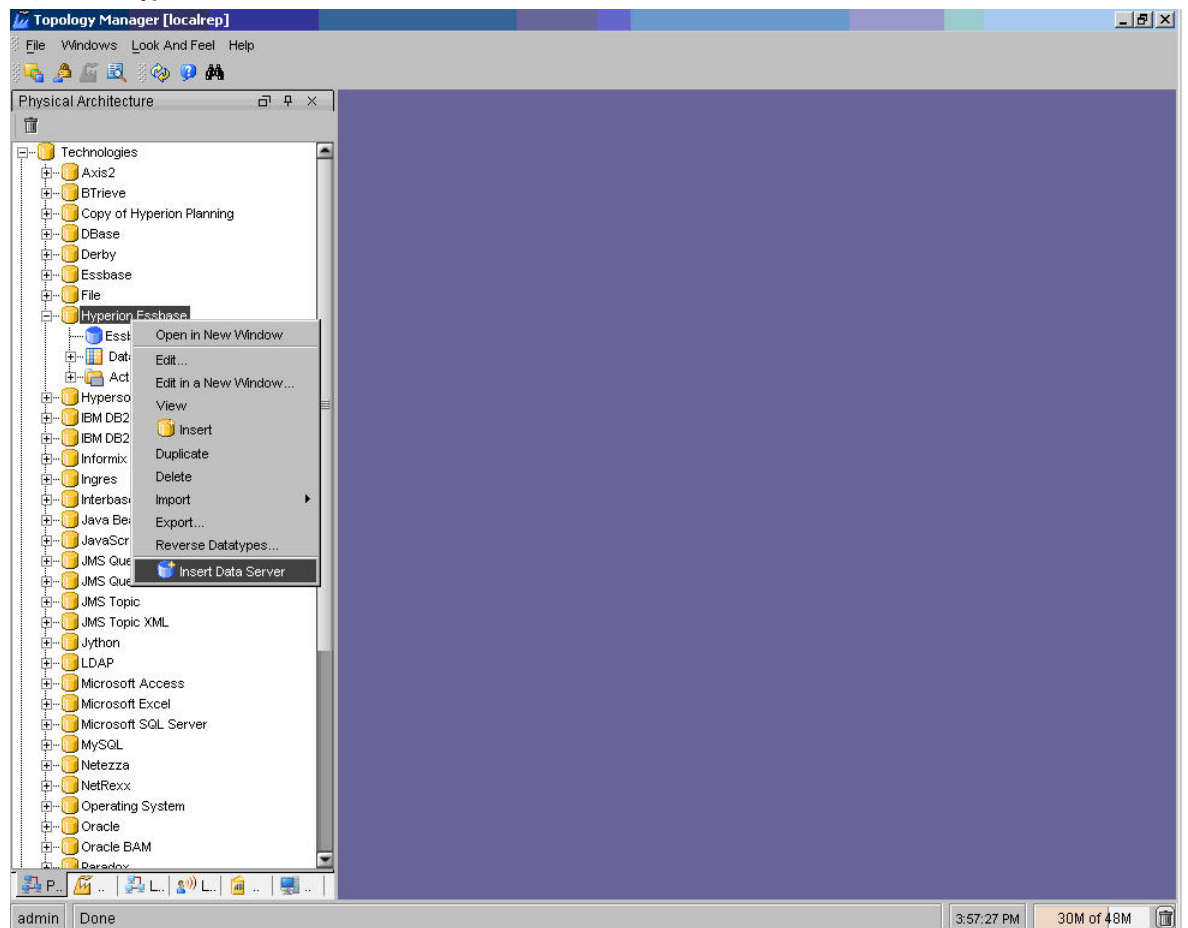
Setting Up Environments

In This Chapter

Defining Data Servers	7
Defining Physical and Logical Schemas and a Context	9

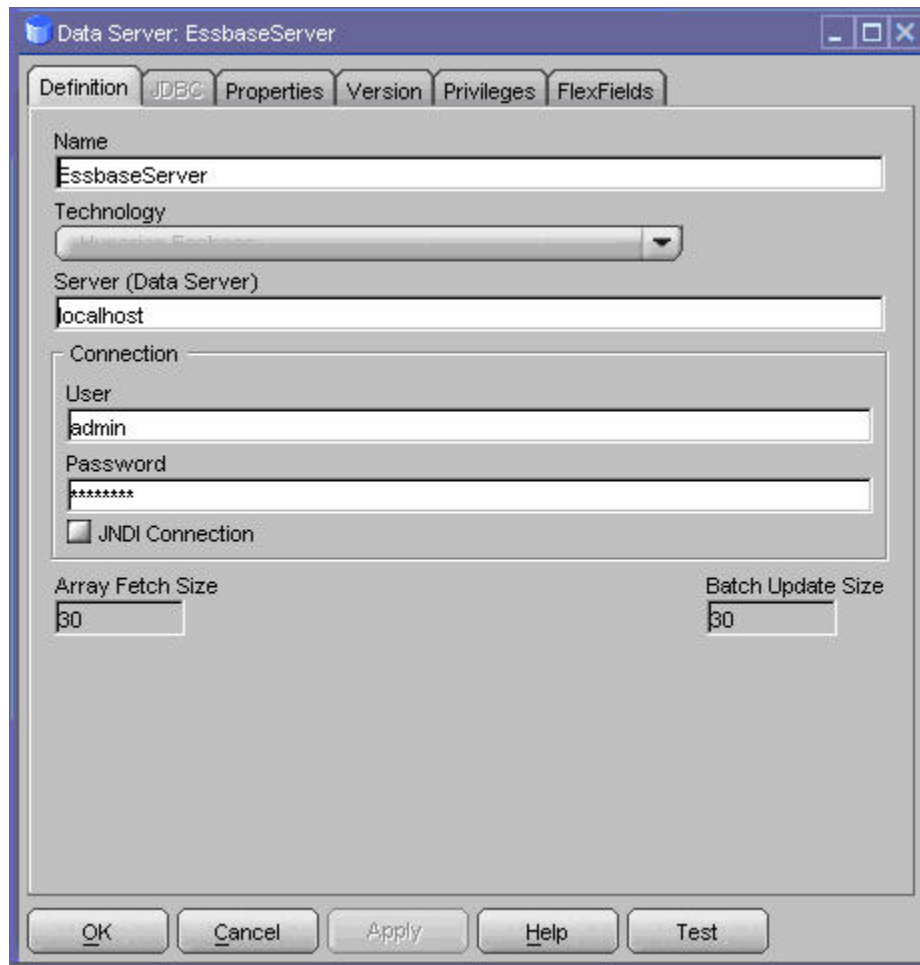
Defining Data Servers

- ▶ To define a data server for connecting to an Essbase server:
 - 1 In **Topology Manager**, expand **Technologies**.
 - 2 Right-click **Hyperion Essbase**, and select **Insert Data Server**.



Note: If the Hyperion Essbase technology is not defined in your master repository, you can import it from the `impexp` folder.

Data Server is displayed.



3 OnDefinition:

- a. Under **Name**, enter a name for the data server definition.
- b. Under **Server (Data Server)**, enter the Essbase server name.

Note: If the Essbase server is running on a port other than the default port (1423), then provide the Essbase server details in this format, `<Essbase Server hostname> : <port>`.

- c. Under **Connection**, enter a user name and password for connecting to the Essbase server.
- d. Click **OK**.

Note: The **Test** button does not work for an Essbase data server connection. This button works only for relational technologies that have a JDBC Driver.

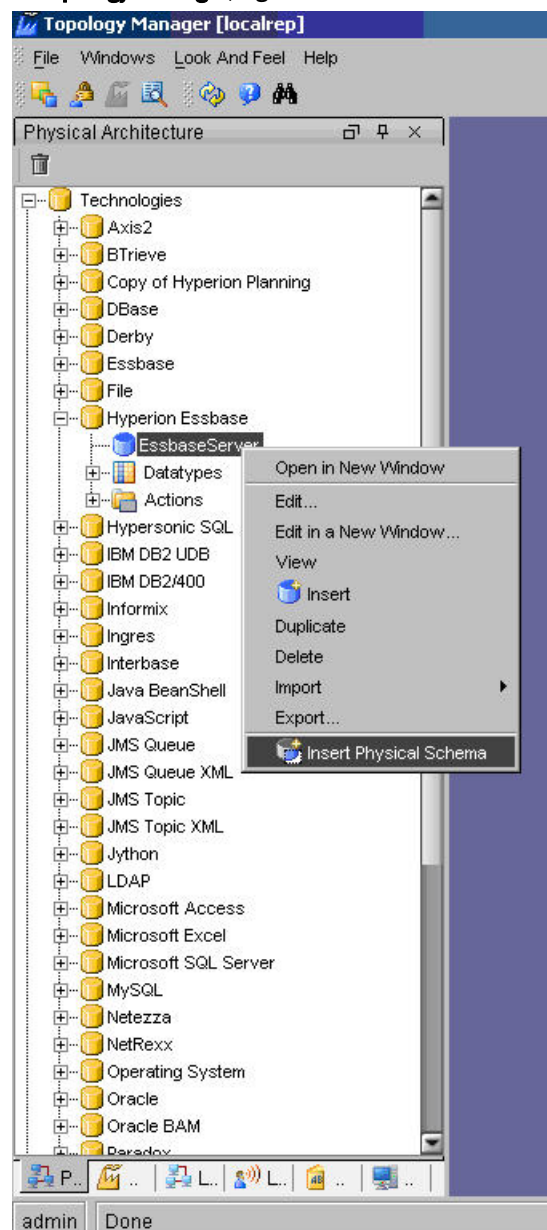
A page for defining a physical schema is displayed. See “[Defining Physical and Logical Schemas and a Context](#)” on page 9.

Defining Physical and Logical Schemas and a Context

Under a data server, you define a physical schema corresponding to an application and the logical schemas on which models are based. You work with Oracle Data Integrator and Adapter for Essbase through a logical schema. A context is used to link the logical schemas and the physical schemas.

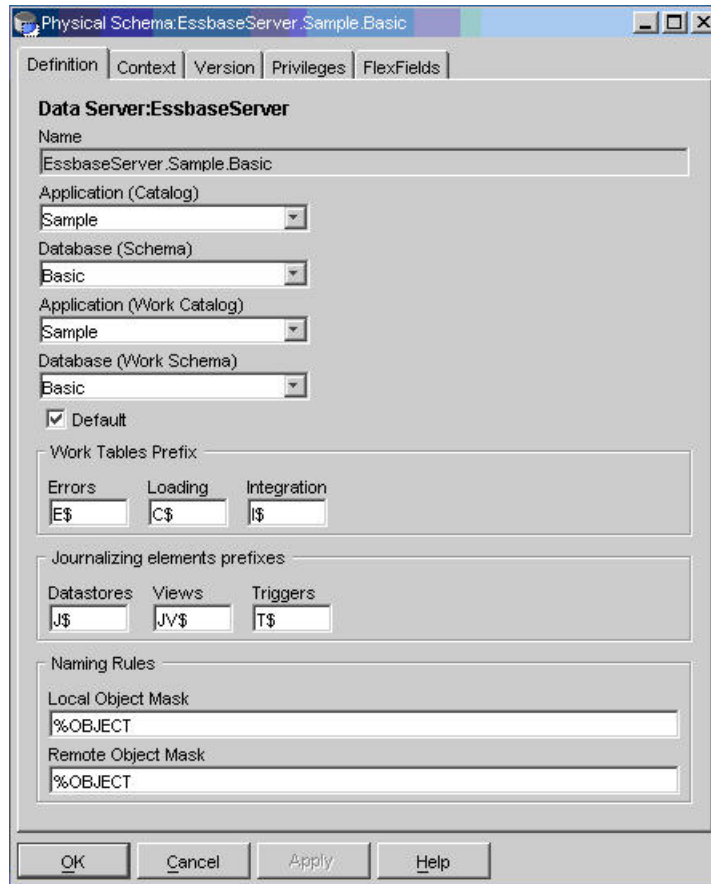
► To create a physical schema:

- 1 In **Topology Manager**, right-click the Essbase data server, and select **Insert Physical Schema**.



The Physical Schema window is displayed.

Figure 1 Physical Schema Definition




- 2 On the **Definition** tab of **Physical Schema** (Figure 1), under **Application (Catalog)** and **Application (Work Catalog)**, specify an Essbase application and under **Database (Schema)** and **Database (Work Schema)**, specify an Essbase database associated with the application you selected.

In Figure 1, the specified application is Sample and database is Basic.

- To specify a logical schema and a context for a new physical schema:

- 1 In the Physical Schema window, on **Context**:

- If one or more Essbase logical schemas exist, select a context and a logical schema.
- If no Essbase logical schemas exist:
 - a. Click .
 - b. Select a context from the left column.
 - c. Enter a name for a logical schema in the right column.

- 2 Click **OK**.

The logical schema that you selected or created is associated with the physical schema in the selected context.

See the *Oracle Data Integrator User's Guide* for more information about physical schemas, logical schemas, and contexts.

3

Reverse-Engineering Essbase Applications

In This Chapter

What Reverse-Engineering Does.....	13
Using the Essbase Reverse-engineering Knowledge Module	13

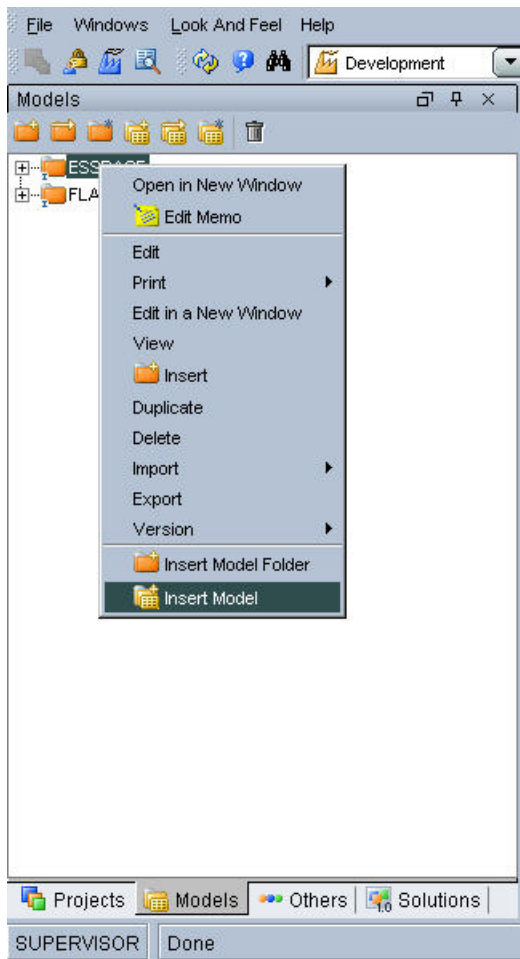
What Reverse-Engineering Does

Reverse-engineering an Essbase application creates an Oracle Data Integrator model that includes a datastore for each dimension in the application, a datastore for data. For more information about reverse-engineering, models, and datastores, see the *Oracle Data Integrator User's Guide*.

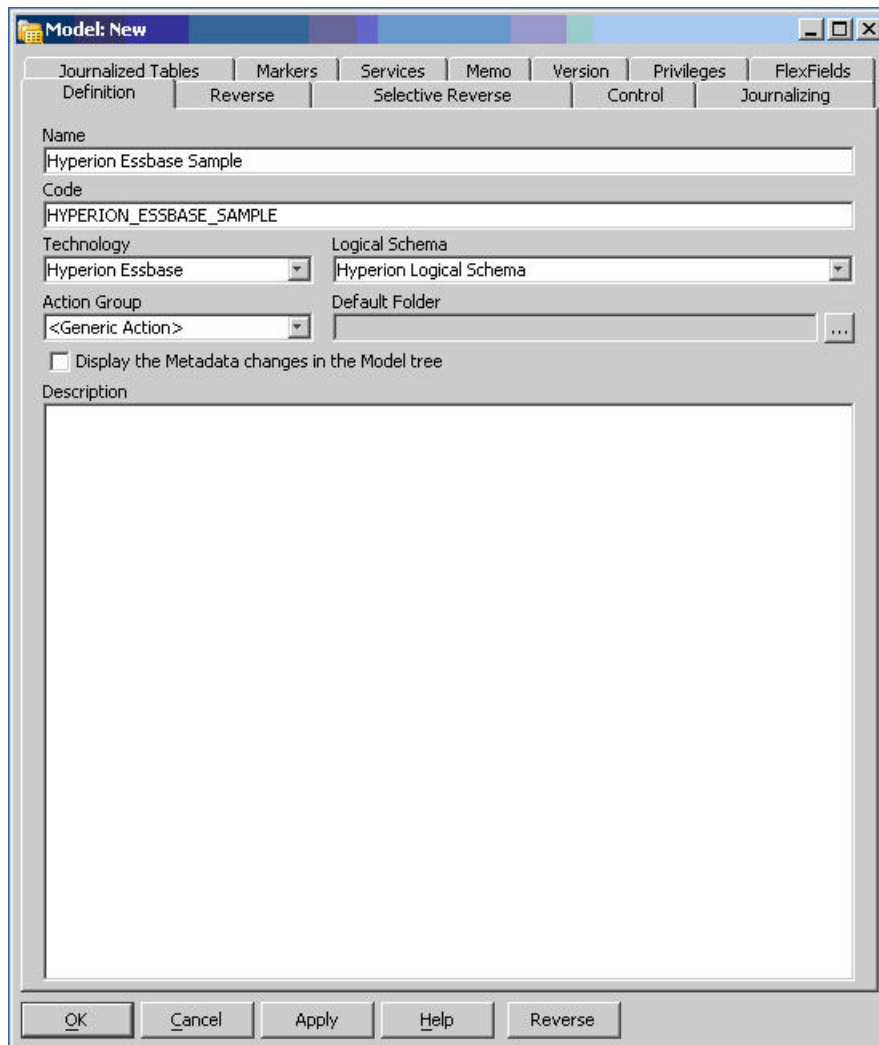
Using the Essbase Reverse-engineering Knowledge Module

Use Oracle Data Integrator Designer to reverse-engineer applications. For more information about Designer, see the *Oracle Data Integrator User's Guide*.

- ▶ To reverse-engineer an Essbase application:
 - 1 In **Designer**, create a model folder—**Essbase**.
 - 2 Right-click **Essbase**, and select **Insert Model**.

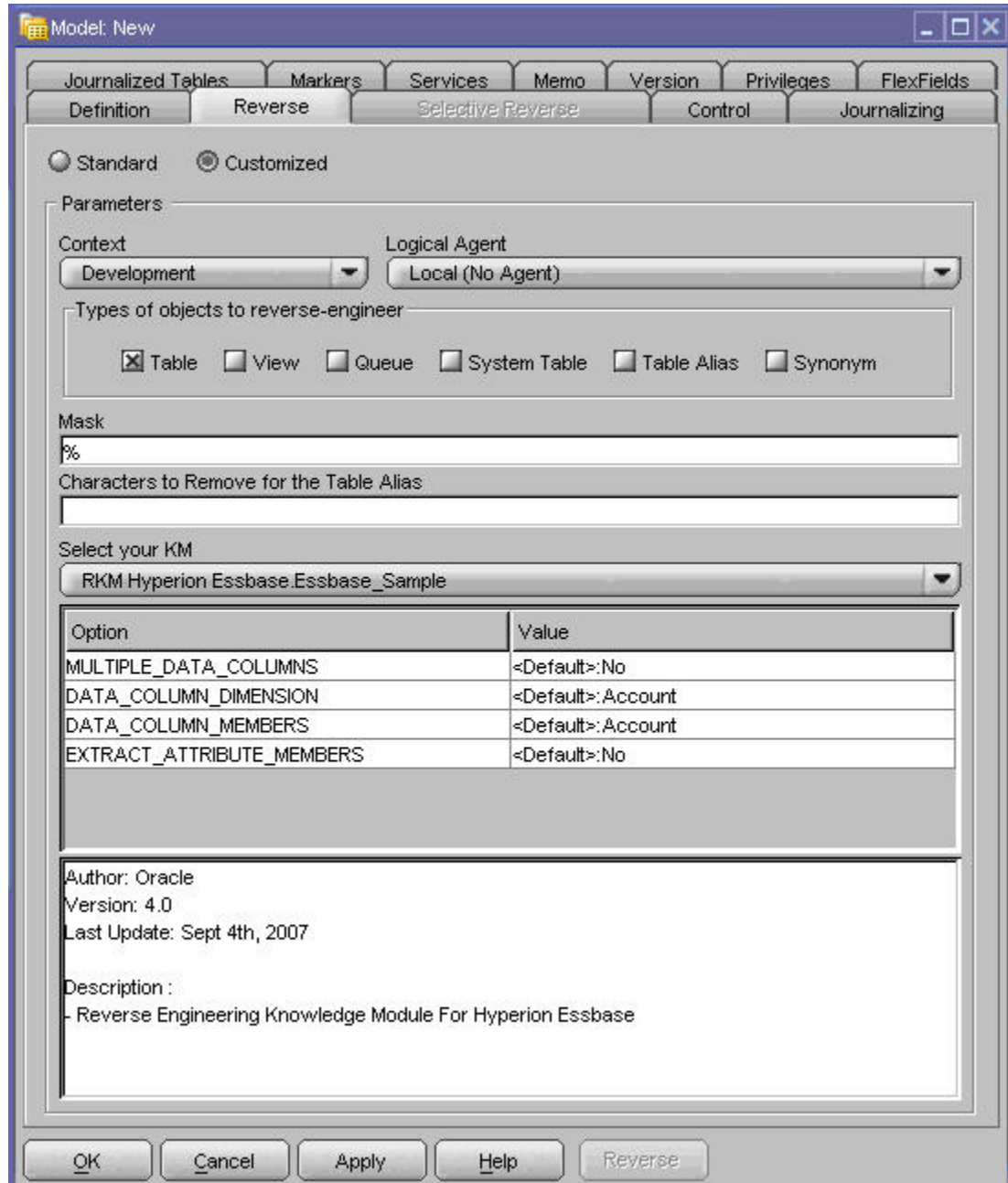


The Model window is displayed.



- 3 Enter a name for the new model, select the **Hyperion Essbase** technology, and (if not already set) select a logical schema on which to base the model.

Figure 2 Model Window Reverse Tab



4 On Reverse (Figure 2):

- a. Select Customized.
- b. Select a context.
- c. Select RKM for Hyperion Essbase from the list of available KMs.
- d. Specify Reverse-engineering Knowledge Module (RKM) options:

Option	Possible Values	Description
MULTIPLE_DATA_COLUMNS	<ul style="list-style-type: none"> ● No (Default) 	If this option is set to No, then the datastore created for the data

Option	Possible Values	Description
	<ul style="list-style-type: none"> ● Yes 	<p>extract / load model contains one column for each of the standard dimensions and a single data column.</p> <p>If this option is set to Yes, then the datastore created for the data extract / load model contains one column for each of the standard dimensions excluding the dimension specified by the DATA_COLUMN_DIMENSION option and as many data columns as specified by the comma separated list for the DATA_COLUMN_MEMBERS option.</p>
DATA_COLUMN_DIMENSION	Account	<p>This option is only applicable if MULTIPLE_DATA_COLUMNS is set to Yes.</p> <p>Specify the data column dimension name.</p> <p>For example, data columns are spread across the dimension Account or Time, and so on.</p>
DATA_COLUMN_MEMBERS	Account	<p>This option is only applicable if MULTIPLE_DATA_COLUMNS is set to Yes.</p> <p>Separate the required data column members with , (Comma).</p> <p>For example, if the data column dimension is set to Account and members are set to "Sales,COGS", then the datastore for data extract/load contains one column for each of the dimension except the data column dimension and one column for each of the data column member specified in the comma separated value. For example. Assuming that the dimensions in the Essbase application are Account, Scenario, Product, Market, and Year and the data column dimension is specified as Account and Data Column Members as "Sales, COGS", the datastore will have the following columns:</p> <ul style="list-style-type: none"> ● Scenario (String) ● Product (String) ● Market (String) ● Year (String)

Option	Possible Values	Description
		<ul style="list-style-type: none"> ● Sales (Numeric) ● COGS (Numeric)
EXTRACT_ATTRIBUTE_MEMBERS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>If this option is set to No, then the datastore created for the data extract / load model contains one column for each of the standard dimensions and a single data column. Attribute dimensions are not included.</p> <p>If this option is set to Yes, then the data model contains these columns.</p> <ul style="list-style-type: none"> ● One column is created for each of the standard dimensions ● One or more Data column(s) are created depending upon the value of the MULTIPLE_DATA_COLUMN option ● One column is created for each of the associated attribute dimension

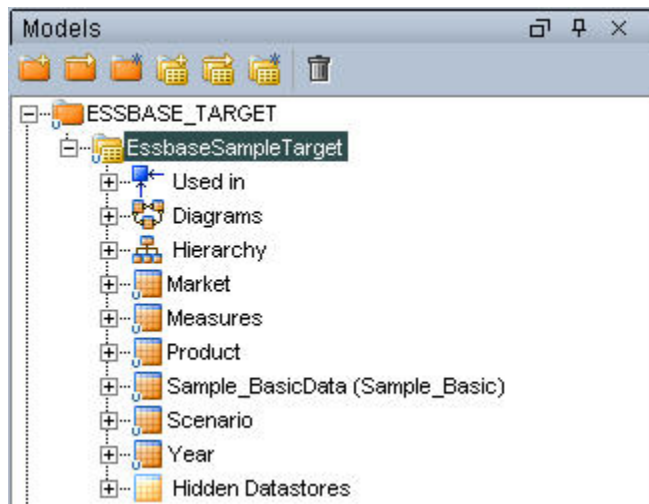
- e. Click **Reverse**.
- f. In **Execution**, click **OK**.
- g. In **Information**, click **OK**.

5 Click Yes when prompted to confirm your entries.

The RKM connects to the application (which is determined by the logical schema and the context) and imports some or all of these datastores, according to the dimensions in the application, The following datastores list is for the Essbase application—Sample and database—Basic:

- Market—For loading the Market dimension
- Measures—For loading the Measures dimension
- Product—For loading the Product dimension
- Scenario—For loading the Scenario dimension
- Year - For loading the Year dimension
- Sample_BasicData (Sample_Basic)—For loading data from the application—Sample and database—Basic

Any errors that occur in the reverse-engineering are listed in the Oracle Data Integrator Operator. For information about Operator, see the *Oracle Data Integrator User's Guide*.



4

Load and Extract

In This Chapter

Data Integration Tasks.....	21
Data Extraction Using Essbase–supported Querying and Scripting	21
Creating Interfaces.....	23
Loading Metadata.....	23
IKM SQL to Hyperion Essbase (METADATA)	24
Loading Data	26
IKM SQL to Hyperion Essbase (DATA)	26
Extracting Data	30
LKM Hyperion Essbase DATA to SQL.....	31
Extracting Members from Metadata	33
LKM Hyperion Essbase METADATA to SQL.....	33

Data Integration Tasks

In Oracle Data Integrator, loading or extracting Essbase application metadata or data involves these tasks:

- Creating interfaces for datastores
- (Optional) Chaining interfaces into packages so that you can run the interfaces in a single process
- Using interfaces

See “[Creating Interfaces](#)” on page 23.

For instructions on creating interfaces and chaining them into packages, see the *Oracle Data Integrator User's Guide*.

Data Extraction Using Essbase–supported Querying and Scripting

The Oracle Data Integrator Adapter for Essbase supports the following Essbase–supported querying and scripting for data extraction. To extract data, as a general process, create an extraction query and provide the extraction query to the adapter. Before the adapter parses the output of the extraction query and populates the staging area, a column validation is done. The

adapter executes the extraction query based on the results of the metadata output query during the validation. The adapter does the actual parsing of the output query only when the results of the column validation are successful.

After the extraction is complete, validate the results—make sure that the extraction query has extracted data for all the output columns.

You can extract data with these Essbase-supported query and scripts:

- Report Script
- MDX Query
- Calculation Script

Data Extraction using Report Scripts

Data can be extracted by parsing the reports generated by report scripts. The report scripts can exist on the client computer as well as server, where Oracle Data Integrator is running on the client computer and Essbase is running on the server. The column validation is not performed when extracting data using report scripts. So, the output columns of a report script is directly mapped to the corresponding connected column in the source model. However, before you begin data extract using report scripts, you must complete these tasks:

- Suppress all formatting in the report script. Include this line as the first line in the report script—`{ROWREPEAT SUPHEADING SUPFORMAT SUPBRACKETS SUPFEED SUPCOMMAS NOINDENTGEN TABDELIMIT DECIMAL 15}`.
- The number of columns produced by a report script must be greater than or equal to the connected columns from the source model.
- The column delimiter value must be set in the LKM option.

Data Extraction Using MDX Queries

An MDX query is an XML-based data-extraction mechanism. You can specify the MDX query to extract data from an Essbase application. However, before you begin data extract using MDX queries, you must complete these tasks:

- The names of the dimension columns must match with the dimensions in the Essbase cube.
- For Type 1 data extraction, all the names of data columns must be valid members of a single standard dimension.
- For Type 1 data extraction, it is recommended that the data dimension exists in the lower level axis, that is, axis (0) of columns. If it is not specified in the lowest level axis then the memory consumption would be high.
- If columns are connected with the associated attribute dimension from the source model, then, the same attribute dimension must be selected in the MDX query.
- The script of the MDX query can be present on the client computer or the server.

Data Extraction Using Calculation Scripts

Calculation scripts provide a faster option to extract data from an Essbase application. However, before you extract data using the calculation scripts, take note of these restrictions:

- Data extraction using calculation scripts is supported ONLY for BSO applications.
- Data extraction using calculation scripts is supported ONLY for the Essbase Release 9.3 and later.
- Set the DataExportDimHeader option to ON.
- (If used) Match the DataExportColHeader setting to the data column dimension (in case of multiple data columns extraction).
- The Oracle Data Integrator Agent, which is used to extract data, must be running on the same machine as the Essbase server.
- When accessing calculation scripts present on the client computer, a fully qualified path to the file must be provided, for example, `C:\Essbase_Samples\Calc_Scripts\calcall.csc`, where as, to access calculation scripts present on the server, only the file name is sufficient.

Creating Interfaces

After reverse-engineering an Essbase application as a model, you can use the datastores in this model in these ways:

- As targets of interfaces for loading data and metadata into the application
- As sources of interfaces for extracting metadata and data from the application.

Loading Metadata

Metadata consists of dimension members. You must load members, or metadata, before you load data values for the members.

You can load members only to dimensions that exist in Essbase. You must use a separate interface for each dimension that you load. You can chain interfaces to load metadata into several dimensions at once.

► To load metadata into an Essbase application:

1 Create an interface for loading metadata.

You can give the interface any name. See the *Oracle Data Integrator Users Guide* for instructions on creating interfaces.

2 On **Diagram, drag and drop the target dimension datastore from an Essbase model.**

3 Define the source datastores.

4 Define the mapping between source and target data.

5 On Flow:

- a. Ensure that IKM SQL to Hyperion Essbase (METADATA) is selected.
- b. Specify load options. See “IKM SQL to Hyperion Essbase (METADATA)” on page 24.

6 Click Execute.

The metadata is loaded into the application.

7 Check the Operator to verify that the interface ran successfully.

Note: The metadata datastore can also be modified by adding or delete columns to match the dimension build rule that will be used to perform the metadata load. For example, the default datastore would have columns for ParentName and ChildName, if the rules is a generational dimension build rule, and you can modify the metadata datastore to match the columns within your generational dimension build rule. The loadMarkets interface within the samples is an example of performing a metadata load using a generational dimension build rule.

IKM SQL to Hyperion Essbase (METADATA)

IKM SQL to Hyperion Essbase (METADATA) dimension supports these options for defining how the adapter loads metadata into an Essbase application:

Option	Possible Values	Description
RULES_FILE	Blank (Default)	Specify the rules file for loading or building metadata. If the rules file is present on the Essbase server, then, only specify the file name, otherwise, specify the fully qualified file name with respect to the Oracle Data Integrator Agent.
RULE_SEPARATOR	, (Default)	(Optional) Specify a rule separator in the rules file. These are the valid values: <ul style="list-style-type: none">● Comma● Tab● Space● Custom character; for example, @, #, ^
RESTRUCTURE_DATABASE	<ul style="list-style-type: none">● KEEP_ALL_DATA (Default)● KEEP_INPUT_DATA● KEEP_LEVEL0_DATA● DISCARD_ALL_DATA	Restructure database after loading metadata in the Essbasecube. These are the valid values: <ul style="list-style-type: none">● KEEP_ALL_DATA— Keep all the data● KEEP_INPUT_DATA—Keep only input data

Option	Possible Values	Description
		<ul style="list-style-type: none"> ● KEEP_LEVEL0_DATA—Keep only level 0 data ● DISCARD_ALL_DATA—Discard all data <p>Note: This option is applicable for the Essbase Release 9.3 and later. For the Essbase releases prior to 9.3, this option is ignored.</p>
PRE_LOAD_MAXL_SCRIPT	Blank (Default)	<p>Enable this option to execute a MAXL script before loading metadata to the Essbase cube.</p> <p>Specify a fully qualified path name for the MAXL script file.</p> <p>Note: To successfully execute this option, the Essbase client must be installed and configured on the machine where the Oracle Data Integrator Agent is running.</p>
POST_LOAD_MAXL_SCRIPT	Blank (Default)	<p>Enable this option to execute a MAXL script after loading metadata to the Essbase cube.</p> <p>Specify a fully qualified path name for the MAXL script file.</p> <p>Note: To successfully execute this option, the Essbase client must be installed and configured on the machine where the Oracle Data Integrator Agent is running.</p>
ABORT_ON_PRE_MAXL_ERROR	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>This option is only applicable if you are enabling the PRE_LOAD_MAXL_SCRIPT option.</p> <p>If you set the ABORT_ON_PRE_MAXL_ERROR option to Yes, then the load process is aborted on encountering any error while executing the pre-MAXL script.</p>
LOG_ENABLED	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>If this option is set to Yes, during the IKM process, logging is done to the file specified in the LOG_FILE_NAME option.</p>
LOG_FILE_NAME	<pre><? =java.lang.System.getProperty("java .io.tmpdir")?>/Extract_< %=snpRef.getFrom()%>.log (Default)</pre>	<p>Specify a file name to log events of the IKM process.</p>
ERROR_LOG_FILENAME	<pre><? =java.lang.System.getProperty("java</pre>	<p>Specify a file name to log the error-records of the IKM process.</p>

Option	Possible Values	Description
	.io.tmpdir"?)?/Extract_<%=snpRef.getFrom()%>.err (Default)	

Loading Data

You can load data into selected dimension members that are already created in Essbase. For a successful data load, all the standard dimension members are required and they should be valid members. You must set up the Essbase application before you can load data into it.

You can also create a custom target to match a load rule.

Before loading data, ensure that the members (metadata) exist in the Essbase dimension. The data load fails for records that have missing members and this information is logged (if logging is enabled) as an error record and the data load process will continue until the maximum error threshold is reached.

► To load data into an Essbase application:

1 Create an interface for loading data.

You can give the interface any name. See the *Oracle Data Integrator Users Guide* for instructions on creating interfaces.

2 On Diagram, drag and drop the target datastore from an Essbase model.

3 Define the source datastores.

4 Define the mapping between source and target data.

5 On Flow:

- a. Ensure that **IKM SQL to Hyperion Essbase (DATA)** is selected.
- b. Specify load options. See [“IKM SQL to Hyperion Essbase \(DATA\)” on page 26](#).

6 Click **Execute.**

The data is loaded into the application.

7 Check the Operator to verify that the interface ran successfully.

Note: The data datastore can also be modified by adding or delete columns to match the data load rule that will be used to perform the data load.

IKM SQL to Hyperion Essbase (DATA)

IKM SQL to Hyperion Essbase (DATA) supports these options for defining how the adapter loads and consolidates data in an Essbase application:

Option	Possible Values	Description
RULES_FILE	Blank (Default)	<p>(Optional) Specify a rules file to enhance the performance of data loading.</p> <p>Specify a fully qualified file name if the rules file is not present on the Essbase server.</p> <p>If the rules file option is not specified, then the API-based data load is used. However, you cannot specify the API.</p>
RULE_SEPARATOR	, (Default)	<p>(Optional) Specify a rule separator in the rules file.</p> <p>These are the valid values:</p> <ul style="list-style-type: none"> ● Comma ● Tab ● Space ● Custom character; for example, @, #, ^
GROUP_ID	Integer	<p>When performing multiple data loads in parallel, many interfaces can be set to use the same GROUP_ID. This GROUP_ID is used to manage parallel loads allowing the data load to be committed when the final interface for the GROUP_ID is complete. For more information on loading to parallel ASO cubes, refer to the Essbase Database Administrators guide.</p>
BUFFER_ID	1-1000000	<p>Multiple data load buffers can exist on an aggregate storage database. To save time, you can load data into multiple data load buffers at the same time. Although only one data load commit operation on a database can be active at any time, you can commit multiple data load buffers in the same commit operation, which is faster than committing buffers individually. For more information on loading to parallel ASO cubes, refer to the Essbase Database Administrators guide.</p>
BUFFER_SIZE	0-100	<p>When performing an incremental data load, Essbase uses the aggregate storage cache for sorting data. You can control how much of the cache a data load buffer can use by specifying the percentage (between 0 and 100% inclusive). By default, the resource usage of a data</p>

Option	Possible Values	Description
		load buffer is set to 100, and the total resource usage of all data load buffers created on a database cannot exceed 100. For example, if a buffer of 90 exists, you cannot create another buffer of a size greater than 10. A value of 0 indicates to Essbase to use a self-determined, default load buffer size.
CLEAR_DATABASE	<ul style="list-style-type: none"> ● None (Default) ● All ● Upper Blocks ● Non-Input Blocks 	<p>Enable this option to clear data from the Essbase cube before loading data into it.</p> <p>These are the valid value</p> <ul style="list-style-type: none"> ● None—Clear database will not happen ● All—Clears all data blocks ● Upper Blocks—Clears all consolidated level blocks ● Non-Input Blocks—Clears blocks containing values derived from calculations <p>Note: For ASO applications, the Upper Blocks and Non-Input Blocks options will not be applicable.</p>
CALCULATION_SCRIPT	Blank (Default)	<p>(Optional) Specify the calculation script that you want to run after loading data in the Essbase cube.</p> <p>Provide a fully qualified file name if the calculation script is not present on the Essbase server.</p>
RUN_CALC_SCRIPT_ONLY	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>This option is only applicable if you have specified a calculation script in the CALCULATION_SCRIPT option.</p> <p>If you set the RUN_CALC_SCRIPT_ONLY option to Yes, then only the calculation script is executed without loading the data into the target Essbase cube.</p>
PRE_LOAD_MAXL_SCRIPT	Blank (Default)	<p>Enable this option to execute a MAXL script before loading data to the Essbase cube.</p> <p>Specify a fully qualified path name for the MAXL script file.</p> <p>Note: Essbase client must be installed and configured on the machine where the Oracle Data Integrator Agent is running.</p>

Option	Possible Values	Description
POST_LOAD_MAXL_SCRIPT	Blank (Default)	<p>Enable this option to execute a MAXL script after loading data to the Essbase cube.</p> <p>Specify a fully qualified path name for the MAXL script file.</p> <p>Note: Essbase client must be installed and configured on the machine where the Oracle Data Integrator Agent is running.</p>
ABORT_ON_PRE_MAXL_ERROR	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>This option is only applicable if you are enabling the PRE_LOAD_MAXL_SCRIPT option.</p> <p>If you set the ABORT_ON_PRE_MAXL_ERROR option to Yes, then the load process is aborted on encountering any error while executing pre-MAXL script.</p>
MAXIMUM_ERRORS_ALLOWED	1 (Default)	<p>Enable this option to set the maximum number of errors to be ignored before stopping a data load.</p> <p>The value that you specify here is the threshold limit for error records encountered during a data load process. If the threshold limit is reached, then the data load process is aborted. For example, the default value 1 means that the data load process stops on encountering a single error record. If value 5 is specified, then data load process stops on encountering the fifth error record. If value 0 (== infinity) is specified, then the data load process continues even after error records are encountered.</p>
COMMIT_INTERVAL	1000 (Default)	<p>Commit Interval is the chunk size of records that are loaded in the Essbase cube in a complete batch.</p> <p>Enable this option to set the Commit Interval for the records in the Essbase cube.</p> <p>Changing the Commit Interval can increase performance of data load based on design of the Essbase database.</p>
LOG_ENABLED	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>If this option is set to Yes, during the IKM process, logging is done to the file specified in the LOG_FILENAME option.</p>

Option	Possible Values	Description
LOG_FILENAME	<? =java.lang.System.getProperty("java.io.tmpdir")?>/< %=snpRef.getTargetTable("RES_NAME")%>.log (Default)	Specify a file name to log events of the IKM process.
LOG_ERRORS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, during the IKM process, details of error records are logged to the file specified in the ERROR_LOG_FILENAME option.
ERROR_LOG_FILENAME	<? =java.lang.System.getProperty(java.io.tmpdir)?>/< %=snpRef.getTargetTable("RES_NAME")%>.err	Specify a file name to log error record details of the IKM process.
ERR_LOG_HEADER_ROW	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, then the header row containing the column names are logged to the error records file.
ERR_COL_DELIMITER	, (Default)	Specify the column delimiter to be used for the error records file.
ERR_ROW_DELIMITER	\r\n (Default)	Specify the row delimiter to be used for the error records file.
ERR_TEXT_DELIMITER	' (Default)	Specify the text delimiter to be used for the column data in the error records file. For example, if the text delimiter is set as ' "' (double quote), then all the columns in the error records file will be delimited by double quotes.

Extracting Data

You can extract data for selected dimension members that exist in Essbase. You must set up the Essbase application before you can extract data from it.

► To extract data from an Essbase application, in Oracle Data Integrator:

1 Create an interface for extracting data.

You can give the interface any name. See the *Oracle Data Integrator Users Guide* for instructions on creating interfaces.

2 On Diagram, drag and drop the source datastore from an Essbase model.

3 Define the target datastores.

4 Define the mapping between source and target data.

5 On the Flow tab:

- a. Ensure that LKM Hyperion Essbase (DATA) to SQL is selected.
- b. Specify extract options, see [“LKM Hyperion Essbase DATA to SQL”](#) on page 31.

6 Click **Execute.**

The data is extracted into the staging area.

7 Check the Operator to verify that the interface ran successfully.

LKM Hyperion Essbase DATA to SQL

LKM Hyperion Essbase Data to SQL supports these options for defining how Oracle Data Integrator Adapter for Essbase extracts data:

Option	Possible Values	Description
PRE_CALCULATION_SCRIPT	Blank (Default)	(Optional) Specify the calculation script that you want to run before extracting data from the Essbase cube.
EXTRACTION_QUERY_TYPE	<ul style="list-style-type: none">● ReportScript (Default)● MDXQuery● CalcScript	<p>Specify an extraction query type—report script, MDX query, or calculation script.</p> <p>Provide a valid extraction query, which fetches all the data to fill the output columns.</p> <p>The first record (first two records in case of calculation script) contains the meta information of the extracted data.</p>
EXTRACTION_QUERY_FILE	Blank (Default)	Specify a fully qualified file name of the extraction query.
EXT_COL_DELIMITER	\t (Default)	<p>Specify the column delimiter for the extraction query.</p> <p>If no value is specified for this option, then space (“ ”) is considered as column delimiter.</p>
EXTRACT_DATA_FILE_IN_CALC_SCRIPT	Blank (Default)	<p>This option is only applicable if the query type in the EXTRACTION_QUERY_TYPE option is specified as CalcScript.</p> <p>Specify a fully qualified file location where the data is extracted through the calculation script.</p>
PRE_EXTRACT_MAXL	Blank (Default)	Enable this option to execute a MAXL script before extracting data from the Essbase cube.

Option	Possible Values	Description
POST_EXTRACT_MAXL	Blank (Default)	Enable this option to execute a MAXL script after extracting data from the Essbase cube.
ABORT_ON_PRE_MAXL_ERROR	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>This option is only applicable if the PRE_EXTRACT_MAXL option is enabled.</p> <p>If the ABORT_ON_PRE_MAXL_ERROR option is set to Yes, while executing pre-MAXL script, the load process is aborted on encountering any error.</p>
LOG_ENABLED	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, during the LKM process, logging is done to the file specified in the LOG_FILE_NAME option.
LOG_FILE_NAME	<? =java.lang.System.getProperty(java.io.tmpdir)?>/Extract_<%=snpRef.getFrom()%>.log (Default)	Specify a file name to log events of the LKM process.
MAXIMUM_ERRORS_ALLOWED	1 (Default)	Enable this option to set the maximum number of errors to be ignored before stopping extract.
LOG_ERRORS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, during the LKM process, details of error records are logged to the file specified in the ERROR_LOG_FILENAME option.
ERROR_LOG_FILENAME	<? =java.lang.System.getProperty(java.io.tmpdir)?>/Extract_<%=snpRef.getFrom()%>.err	Specify a file name to log error record details of the LKM process.
ERR_LOG_HEADER_ROW	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, then the header row containing the column names are logged to the error records file.
ERR_COL_DELIMITER	, (Default)	Specify the column delimiter to be used for the error records file.
ERR_ROW_DELIMITER	\r\n (Default)	Specify the row delimiter to be used for the error records file.
ERR_TEXT_DELIMITER	\ " (Default)	<p>Specify the text delimiter to be used for the column data in the error records file.</p> <p>For example, if the text delimiter is set as ' ' (double quote), then all the columns in the error records file are delimited by double quotes.</p>

Option	Possible Values	Description
DELETE_TEMPORARY_OBJECTS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	<p>Set this option to No, in order to retain temporary objects (tables, files, and scripts) after integration.</p> <p>This option is useful for debugging.</p>

Extracting Members from Metadata

You can extract members from selected dimensions in an Essbase application. You must set up the Essbase application and load metadata into it before you can extract members from a dimension.

Before extracting members from a dimension, ensure that the dimension exists in the Essbase database. No records are extracted if the top member does not exist in the dimension.

► To extract members from a dimension in an Essbase application, in Oracle Data Integrator:

1 Create an interface for loading metadata.

You can give the interface any name. See the Oracle Data Integrator Users Guide for instructions on creating interfaces.

2 On **Diagram, drag and drop the source dimension datastore from an Essbase model.**

3 Define the target datastores.

4 Define the mapping between source and target data.

5 On the **Flow:**

a. Ensure that LKM Hyperion Essbase (METADATA) to SQL is selected.

b. Specify extract options, see [“LKM Hyperion Essbase METADATA to SQL” on page 33](#).

6 Click **Execute.**

The metadata is loaded into the application.

7 Check the Operator to verify that the interface ran successfully.

LKM Hyperion Essbase METADATA to SQL

LKM Hyperion Essbase METADATA to SQL supports these options for defining how Oracle Data Integrator Adapter for Oracle Essbase extracts dimension members:

Option	Possible Values	Description
MEMBER_FILTER_CRITERIA	IDescendants, (Default)	<p>Enable this option to select members from the dimension hierarchy for extraction. You can specify these selection criteria:</p> <ul style="list-style-type: none"> ● IDescendants

Option	Possible Values	Description
		<ul style="list-style-type: none"> ● Descendants ● IChildren ● Children ● Member_Only ● Level0 ● UDA
MEMBER_FILTER_VALUE	Blank (Default)	<p>Enable this option to provide the member name for applying the specified filter criteria. If no member is specified, then the filter criteria is applied on the root dimension member.</p> <p>If the MEMBER_FILTER_CRITERIA value is MEMBER_ONLY or UDA, then the MEMBER_FILTER_VALUE option is mandatory and cannot be an empty string.</p>
LOG_ENABLED	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, during the LKM process, logging is done to the file specified by the LOG_FILE_NAME option.
LOG_FILE_NAME	<? =java.lang.System.getProperty(java.io.tmpdir"?>/Extract_<%=snpRef.getFrom()%>.log	Specify a file name to log events of the LKM process.
MAXIMUM_ERRORS_ALLOWED	1 (Default)	Enable this option to set the maximum number of errors to be ignored before stopping extract.
LOG_ERRORS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, during the LKM process, details of error records are logged to the file specified in the ERROR_LOG_FILENAME option.
ERROR_LOG_FILENAME	<? =java.lang.System.getProperty(java.io.tmpdir"?>/Extract_<%=snpRef.getFrom()%>.err	Specify a file name to log error record details of the LKM process.
ERR_LOG_HEADER_ROW	<ul style="list-style-type: none"> ● No (Default) ● Yes 	If this option is set to Yes, then the header row containing the column names are logged to the error records file.
ERR_COL_DELIMITER	, (Default)	Specify the column delimiter to be used for the error records file.
ERR_ROW_DELIMITER	\r\n (Default)	Specify the row delimiter to be used for the error records file.

Option	Possible Values	Description
ERR_TEXT_DELIMITER	<ul style="list-style-type: none"> ● Blank (Default) ● \" ● \" 	Specify the text delimiter to be used for the data column in the error records file. For example, if the text delimiter is set as ' ' ' (double quote), then all the columns in the error records file are delimited by double quotes.
DELETE_TEMPORARY_OBJECTS	<ul style="list-style-type: none"> ● No (Default) ● Yes 	Set this option to No, in order to retain temporary objects (tables, files, and scripts) after integration. This option is useful for debugging.

