Oracle® Hyperion Financial Data Quality Management, Enterprise Edition

Administrator's Guide

Release 11.1.2.3.530 Updated: May 2015



FDMEE Administrator's Guide, 11.1.2.3.530

Copyright © 2009, 2015, Oracle and/or its affiliates. All rights reserved.

Authors: EPM Information Development 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 is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware 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 that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

Document	ation Accessibility	11
Document	ation Feedback	13
Chapter 1	. Using FDMEE	13
	Supported Source Systems	15
	File-based Data Loads	16
	Supported EPM Target Applications	17
	Upgrading to FDMEE	17
	Post Upgrade	18
	FDMEE Application Architecture	18
	Architecture	20
	Drilling Through	22
	Drilling Through to the FDMEE Landing Page	23
	Configuring Oracle Data Integrator With FDMEE	23
	Set up the Data Server Based on the ERP Source System	24
	Setting up Physical Schemas	24
	Setting up the ODI Context Code	25
	Setting up ODI to Integrate with JD Edwards	25
	Security	27
	Integration Process Overview	28
	Extracting General Ledger Data	28
	Writing Back Data from EPM Applications	29
	Extracting Human Resource Data	29
	Navigating FDMEE	30
	Toolbars	30
	Help Menu	30
	Task Pane Options	30
	Working with Data in Grids	32
	FDMEE User Interface Elements	32
	Advanced Search Options	33
	Using the POV Bar	34

	Administration Tasks	35
	Predefining a List of Profiles	35
	Setting up Source Systems	48
	Working with Source Adapters	52
	Registering Target Applications	63
	Selecting Source Accounting Entities	80
	Assigning General Ledger Responsibility	81
	Working with Source Accounting Entity Groups	82
	Working with Import Formats	83
	Defining Locations	98
	Defining Period Mappings	00
	Defining Category Mappings	.06
Chapter 2.	Loading Metadata	09
	Understanding General Ledger Integration	09
	Requirements	09
	Required Dimensions	10
	Member Properties Sourced from the ERP System	11
	Account	11
	Entity and Intercompany	12
	Scenario	13
	Version	13
	View 1	13
	Year and Period	13
	Alias 1	14
	Custom	14
	How Dimensions are Processed	14
	How Languages are Processed	15
	How Currencies are Processed	16
	Loading Source System Hierarchies into EPM Dimensions	17
	Defining Metadata Rules	18
	Defining the Metadata Rule Details	19
	Defining Dimension Attributes	22
	Managing Metadata Rules	23
Chapter 3.	Loading Data	25
	Creating Member Mappings	26
	Creating Mappings Using the Explicit Method	28
	Creating Mappings Using the Between Method	29
	Creating Mappings Using the In Method	30

Creating Mappings Using the Like Method	30
Using Special Characters in the Source Value Expression for Like Mappings 13	31
Creating Mappings Using Multi-Dimensions	33
Automap Wildcarding	34
Using Special Characters in Multi-Dimensional Mapping	35
Using Special Characters in the Target Value Expression	35
Format Mask Mapping for Target Values	36
Ignoring Member Mappings 13	38
Importing Member Mappings	39
Downloading an Excel Template (Mapping Template) 14	40
Importing Excel Mappings	41
Exporting Member Mappings	42
Deleting Member Mappings	43
Restoring Member Mappings	43
Defining Data Load Rules to Extract Data	43
Defining Data Load Rule Details	44
Defining Data Load Rule Details for a file-based Source System	61
Using a Multiple Period Data Load with a Data Load Rule	62
Managing Data Load Rules	62
Editing Data Load Rules	63
Running Data Load Rules	63
Checking the Data Load Rule Status	67
Deleting Data Load Rules	68
Creating Integration Information	68
Using the Data Load Workbench	68
Workflow Grid	69
Processing Data	70
Using the Workbench Data Grid	74
Viewing Process Details	80
Using Excel Trial Balance Files to Import Data	82
Text Trial Balance Files Versus Excel Trial Balance Files	82
Downloading an Excel Trial Balance Template	82
Defining Excel Trial Balance Templates 18	83
Adding a Multiple Period Data Load using Excel	84
Using Journal Templates to Import Data	84
Integrating Financial Management Journals	85
Downloading a Journal Template	86
Defining Journal Templates	86
Processing Journal Files	89

	Loading Journals	
Chapter 4. L	Logic Accounts	
	Overview of Logic Accounts	
	Creating a Logic Group	
	Creating Accounts In A Simple Logic Group	
	Logic Group Fields	
	Creating Complex Logic Accounts	
	Criteria Value	
	Dimension	
	Criteria Type	
	Criteria Value	
	Group By	
	Group Level	
	Include Calc	
	Complex Logic Example 1: CashTx	
	Complex Logic Example 2: CashTx	
Chapter 5. (Check Rules	
	Overview of Check Rules	
	Creating Check Rule Groups	
	Creating a Check Rule	
	Display Value	
	Description	
	Rule Name	
	Rule Text	
	Category	
	Sequence	
	Rule Logic	
	Using the Rule Logic Editor to Create Check Rules	
	Creating Check Entity Groups	
Chapter 6. [Data Write-Back	
	Creating Mapping Definitions	
	Defining Data Write-Back Rules	
	Defining the Source Filter Options	
	Defining Target Options	
	Defining Write-Back for PeopleSoft Commitment Control	
	Deleting Write Back Member Mappings	
	Managing Write-Back Rules	

	Editing Write-Back Rules
	Running Write-Back Rules
	Using the Write-Back Workbench
	Checking the Data Write-Back Rule Status
	Deleting Data Write-Back Rules
	Loading Data into Supported Source Systems
	Loading Data into Oracle E-Business Suite
	Loading Data into PeopleSoft Enterprise Financial Management
Chapter 7.	Loading Human Resources Data
	Understanding Human Resource Data Integration
	Requirements
	Running Processes in PeopleSoft Human Capital Management
	Defining the Data Load Settings in Planning
	Public Sector Planning and Budgeting Dimensions
	Smart Lists
	Using Loaded Data in Public Sector Planning and Budgeting
	Defining Human Resource Data Load Rules
	Defining the Rule Details
	Creating Rule Line Definitions by Copying
	Creating Rule Line Definitions by a Template Mapping
	Creating Rule Line Definitions
	Creating Mapping Definitions
	Managing Human Resource Data Load Rules
	Editing Human Resource Data Load Rules
	Running Human Resource Data Load Rules
	Deleting Human Resource Data Load Rules
	Checking the Human Resource Data Load Rule Status
Chapter 8.	Batch Processing
	Working with Batch Definitions
	Adding a Batch Group
	Executing Batches
	Using Open Batches
	Creating Open Batches
	Scheduling Jobs
	Canceling a Scheduled Job
	Working with Batch Scripts
	Using Password Encryption
	Executing the Batch Script for Data Load Rules

	Executing the Batch Script for Metadata Rules	53
	Executing the Batch Script for HR Data Rules	53
	Executing the Batch Script for Write-Back Rules	54
	Running a Batch	55
Chapter 9. Cro	eating and Using Scripts	57
	Overview	57
	Key Jython Concepts	57
	Using the Script Editor	68
	Overview	<u>5</u> 8
	Launching the Script Editor	<u>5</u> 8
	Script Editor Layout	68
	Using Import Scripts	<u>59</u>
	Overview	<u>59</u>
	Creating Import Scripts	<u>59</u>
	Import Script Parameters	['] 0
	Assigning Import Scripts to Import Formats	1
	Using Import Script Functions	13
	Storing and Retrieving Temporary Variables	7
	Import Script Sample	⁷ 9
	Using Mapping Scripts	30
	Overview	30
	Creating Mapping Scripts	30
	Using Jython Objects with Mapping Scripts	32
	Mapping Script Sample	32
	Using Event Scripts	33
	Overview	3
	FDMEE Supported Event Scripts	3
	Creating Event Scripts	34
	Dynamically Changing Import Formats	35
	Using the File System Object in Event Scripts	35
	Events Script Sample	36
	Using Custom Scripts	37
	Overview	37
	Creating a Custom Script	37
	Working with Custom Scripts	38
	Executing a Custom Script	90
	Custom Script Sample using Jython	90
	Working with Drill Through Scripts	2

	Using the JAVA API
	Overview
	JAVA API List
	Working with UNICODE Characters in Jython Scripts
	Using JAVA IDE to Develop Scripts
	Using Visual Basic API and Samples
	Overview
	Visual Basic API List
	Visual Basic Sample Scripts
Chapter 10	D. FDMEE Reports
	FDMEE Reports
	Working with Query Definitions
	Working with Report Definitions
	Adding Report Groups 325
	Working with Report Definitions
	Creating a Report Template
	Running Reports
	FDMEE Detail Reports
	Audit Reports
	Check Reports
	Base Trial Balance Reports
	Listing Reports
	Location Analysis
	Process Monitor Reports
	Variance Reports
Appendix <i>A</i>	A. FDMEE Web Service
	executeDataRule
	executeMetaDataRule
	getDataRuleNames
	getLocationNames
	getPeriodNames
Appendix E	3. Source System Tables Used by FDMEE
	Fusion and E-Business Suite Source System Tables
	PeopleSoft Enterprise Financial Management Source System Tables
	PeopleSoft Human Capital Management Source System Tables
	Creating Synonyms for FRS Tables

Appendix C	Creating an Alternate Schema in an ERP Source System	355
Appendix D	D. Staging Tables 3	357
	FDMEE Staging Tables	357
	Staging Tables Used for Import from Source	357
	Data Tables Used for Drill Through	358
	Views Used for Export to Target 3	358
	PeopleSoft Human Capital Management Staging Tables 3	358
	PS90HCM	358
	CHARTFIELD	377
Appendix E	. Working with Open Interface Adapters	379
	Setting up the Open Interface Adapter	379
	Using the Open Interface Table 3	379
	Loading Source Data using Custom Loaders	380
	Customizing ODI to Load Data from a Custom Source	38]
Appendix F	PeopleSoft Commitment Control	383
Appendix G	G. Integrating FDM	385
	Enabling FDM Applications	385
	Drilling Through to the FDM Landing Page	386
Appendix H	I. Report Query Table Reference	387
	TDATASEG Table Reference	387
	TLOGPROCESS Table Reference	39]
Appendix I.	. Setting up Jython, Eclipse and Python	395
Glossarv		an.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Documentation Feedback

Send feedback on this documentation to: epmdoc_ww@oracle.com

Follow EPM Information Development on these social media sites:

LinkedIn - http://www.linkedin.com/groups?gid=3127051&goback=.gmp_3127051

Twitter - http://twitter.com/hyperionepminfo

Facebook - http://www.facebook.com/pages/Hyperion-EPM-Info/102682103112642

Google + - https://plus.google.com/106915048672979407731/#106915048672979407731/posts

YouTube - http://www.youtube.com/user/OracleEPMWebcasts

Using FDMEE

1

In This Chapter

Supported Source Systems	15
File-based Data Loads	16
Supported EPM Target Applications	17
Upgrading to FDMEE	17
Architecture	20
Drilling Through	22
Configuring Oracle Data Integrator With FDMEE	23
Security	27
Integration Process Overview	28
Extracting General Ledger Data	28
Writing Back Data from EPM Applications	29
Extracting Human Resource Data	29
Navigating FDMEE	30
Administration Tasks	35

Oracle Hyperion Financial Data Quality Management, Enterprise Edition enables you to perform the following tasks:

- Integrate metadata and data from an Enterprise Resource Planning (ERP) source system into an Enterprise Performance Management (EPM) target application.
- Drill through from the EPM target application and view data in the ERP source system
- Write-back budget data to the source system from any Oracle Hyperion Planning, Oracle Essbase aggregate storage, or Essbase block storage application.

For information on supported EPM System versions, see the *Oracle Hyperion Enterprise Performance Management System Certification Matrix*.

For FDMEE issues and workarounds, see the *Oracle Hyperion Financial Data Quality Management Readme*.

Supported Source Systems

FDMEE supports general ledger data for:

- File-based data loads that enable users to import balances from delimited or fixed-width text files.
- Oracle E-Business Suite 11i
- Oracle E-Business Suite 12
- Oracle Fusion Financials
- PeopleSoft Enterprise Financial Management 9
- PeopleSoft Commitment Control
- SAP ERP Financial
- SAP BW (Business Warehouse)
- JD Edwards General Ledger system

FDMEE supports human resource data load for:

- PeopleSoft Human Capital Management 9.0
- PeopleSoft Human Capital Management 9.1

Integration includes data load and drill through. The integration is supported for the following SAP ERP Financial modules:

- General Ledger
- Profit Center
- Cost Center
- Vendor Balances
- Customer Balances

In addition FDMEE provides support for Open Interface Adapter. The Open Interface Adapter allows you to import data from any source system using the interface table.

For information on supported technologies for each source system, see the *Oracle Hyperion Enterprise Performance Management System Certification Matrix*.

File-based Data Loads

File-based imports are supported for those users who do not have a direct connection to their ERP source data, but have data available from their source in a text file. Any file, whether it is a fixed width file or a delimited file, may be easily imported into the target EPM application. For example, you can take a trial balance report generated from their source system, and map it into FDMEE by way of the import format feature. You can instruct the system where the account, entity, data values and so on reside in the file, as well as which rows to skip during the data import. This feature allows a business user to easily import data from any source, and requires limited technical help, if any, when loading into an EPM application like Planning, Oracle Hyperion Financial Management, or Essbase.

Supported EPM Target Applications

Supported EPM target applications (Oracle Hyperion EPM Architect and Classic) are:

- Planning
- The Planning module, Oracle Hyperion Public Sector Planning and Budgeting
- Financial Management
- Essbase aggregate storage and Essbase block storage—If Essbase has been deployed in standalone mode, it must be registered with Shared Services since it does not work directly with FDMEE. Planning applications that rely on the standalone Essbase application cannot work directly with FDMEE either.
- Account Reconciliation Manager (ARM)

Upgrading to FDMEE

In Release 11.1.2.3, FDMEE is offered as the full-featured successor to Oracle® Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications (ERP Integrator) and Oracle Hyperion Financial Data Quality Management (FDM). FDMEE represents the integration of both products. It captures key functionality from FDM, and preserves the ERP Integrator framework that enables users to load metadata and data, and to drill-through and write-back. (For an online introduction to ERP Integrator (now called FDMEE (Release 11.1.2.2.) see http://www.youtube.com/watch?v=i5cxkaUw90Mhttp://www.youtube.com/watch?v=i5cxkaUw90M.)

Note: For an online introduction to FDMEE (Release 11.1.2.2.) see http://www.youtube.com/watch?v=i5cxkaUw90Mhttp://www.youtube.com/watch?v=i5cxkaUw90M.

Key benefits supported in FDMEE include:

- Improved User Experience—Integrated in Oracle Enterprise Performance Management System. The FDMEE user interface is consistent with the Planning and Financial Management user interfaces.
- Improved performance—Improved user interface and data load performance
- Close Oracle Hyperion Shared Services integration—All native Shared Services features are supported (for example, user groups)
- Support for multiple browsers—Internet Explorer, Firefox
- Support for Oracle Hyperion Enterprise Performance Management System Lifecycle
 Management—Consistent Lifecycle Management support like other EPM System products
- Support for multiple platforms—All supported platforms. If integrating with Financial Management only Windows operating system is supported
- Consistent Scaling and Load Balancing—All scale out configuration and load balancing steps are consistent with other EPM System products

Post Upgrade

After installing or upgrading to FDMEE, configuration settings must be executed by the administrator. The administrator needs to create an FDMEE application folder on the server, and then specify this folder in the system settings (see "Setting System Level Profiles" on page 36). This folder functions as the root folder for all data files, log files, report templates, and any other user generated files. Once specified, the user selects the "Create Application Folders" button to create the required sub folders under the top level root folder on the server.

- To set up the application root directory:
- 1 From the Setup tab, and then under Configure, select System Settings.
- 2 In System Settings, from Profile Type, select File.

When you select the **File** profile type, the System Setting screen displays the "Create Application Folders" button.

3 In Application Root Folder, specify the root directory of the application.

For example, specify C:\APPDATA\FDMEE.

Note: Use a Universal Naming Convention (UNC) path for the application root folder when Financial Management and FDMEE are on separate servers. Share permissions on the folder must allow access to the DCOM user for read/write operations. Contact your server administrator to define the required UNC definition.

- 4 Click Create Application Folders.
- 5 Click Save.

FDMEE Application Architecture

An FDMEE application consists of a relational database management system (RDBMS) database and directories that contain the transactional data, metadata, reports, and other files that are used to integrate FDMEE with target applications. One FDMEE application can load to multiple target applications of multiple systems. For example, one FDMEE application can load to two Financial Management applications, and three Planning applications.

The FDMEE application uses the following directories:

data

In data, FDMEE archives data files that are imported to and exported from FDMEE (imported source files, import logs, journal entries, multiload files, target system load files, and any attached memo documents). Source files, journals, multiload files, logs, output data files, and attachments are also archived here. Each file in data is assigned a unique name, can be retrieved from the Import and Export screens, and provides an audit trail.

Supported Encodings

Encoding refers to mapping *bit* combinations to characters for creating, storing, and displaying text

Encoding is selected from the File Character Set in "Setting System Level Profiles" on page 36.

You should convert the encoding to UNICODE if your source file is not in one of the supported formats.

scripts

data contains a scipts subdirectory, where FDMEE scripts are stored. Scripts contains subdirectories, each of which contains scripts of a specific type:

- event—Scripts that are executed when a particular event runs. The scripts are named for the events with which they are associated.
- import—Import scripts that are created when import formats are defined (Import scripts are executed during the source file import process).
- custom—Custom scripts executed on an as needed basis.

inbox

You can use inbox, the default directory from which to import source files, as a central repository for all ledger extract files. Because source files can be retrieved from any accessible directory, you are not required to place import files in this directory. In addition, the inbox is where location based folders are created when you select the option to create a folder for every location created by the user.

Inbox includes the batches directory.

batches

Files used for batch loading are stored in batches, standard batch files in the openbatch subdirectory, and multiple load batch files in the OpenBatchML directory.

outbox

outbox provides a central location for all FDMEE export files. Outbox also contains four subdirectories: excel Files, logs, templates, and archive.

logs

logs contain the log files that are created when source files are imported. The logs contain the data lines that FDMEE did not import and an explanation of why each line was not imported. logs also contains error logs, which are named per the following convention

(username.err where username is the user that is logged on to FDMEE and .err is the extension used to identify error logs. Logs can be purged to reclaim disk space.

reports

reports stores report output generated by the batch process in PDF, HTML or XLS.

Architecture

FDMEE is the key application for integrating ERP systems with Oracle's Hyperion EPM applications. FDMEE fits within the Oracle Enterprise Performance Management System architecture and is accessed through Oracle Hyperion Enterprise Performance Management Workspace, which uses Shared Services to authenticate users. The key to its integration lies within its underlying engine, which is Oracle Data Integrator.

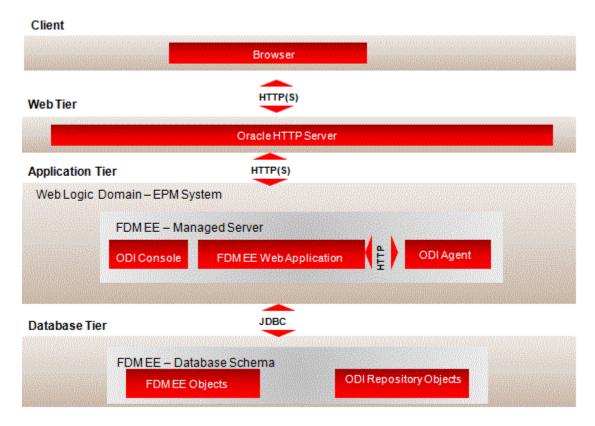
FDMEE sits on top of Oracle Data Integrator and orchestrates the movement of metadata and data into Hyperion EPM applications. The application server can be deployed on multiple platforms (See the *Oracle Hyperion Enterprise Performance Management System Certification Matrix*) and connects with Hyperion EPM applications such as Financial Management, Planning, and Essbase.

The following diagram displays the FDMEE high level architecture.

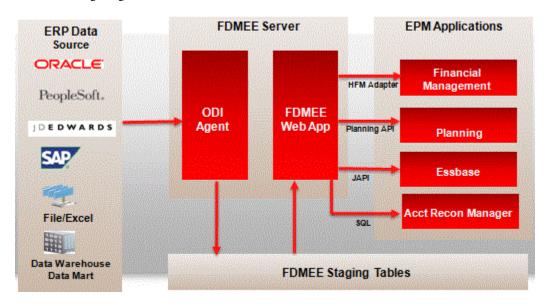
DATA SOURCES ORACLE EPM SYSTEM EPM Applications EBS GL FDM, Enterprise Edition Planning Fusion GL PeopleSoft Financials and HCM Financial Close SmartView Management (HFM, ARM) SAP FI / CO and BW Oracle Data Integrator JDE EnterpriseOne Financial Reports Open Interface Table Files **EPM Architect** Drill Through Metadata and Data Load Write Back

FDM, Enterprise Edition High Level Architecture

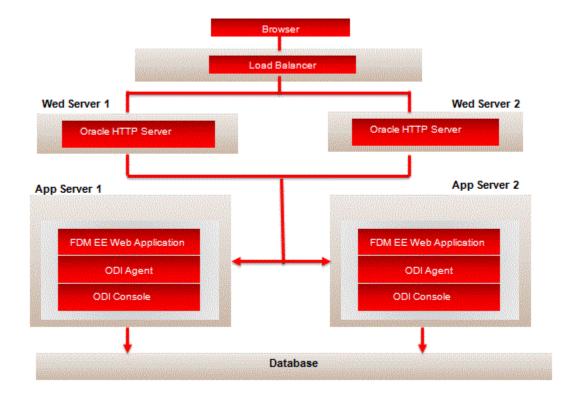
The following diagram shows the tier structure of FDMEE:



The following diagram shows the flow of data in FDMEE:



The following diagram shows FDMEE High Availability:



Drilling Through

FDMEE provides the framework to drill through from the EPM applications back to the general ledger source. Drill through is not supported for human resource data. Users can drill through to detail in the source system through FDMEE *from* the following products:

Note: If the source system is EBS/Peoplesoft and you have metadata rules, then the drill region is created based on the metadata rule. Otherwise, it is created based on the target members in the data load mappings. For Year, Period, and Scenario, FDMEE uses audit information to create the drill region.

Note: Drill through is not supported for Financial Management journals and intercompany transactions.

- Planning
- Financial Management
- Oracle Smart View for Office
- Oracle Hyperion Financial Reporting

Note: In Smart View and Financial Reporting, you can drill through only when the data source is Financial Management, Planning, or Essbase.

When you drill through, if data was loaded by FDMEE, a landing page is displayed in a new EPM Workspace tab or a new window. The landing page is a gateway to the data in the source system. See "Drilling Through to the FDMEE Landing Page" on page 23.

Drilling Through to the FDMEE Landing Page

The FDMEE landing page displays general ledger accounts and the hyperlinked balances that were used to populate the cells in the EPM application. When you click a hyperlinked data value, you can drill through to the source system and view the associated journal entries for the selected general ledger account.

You can drill through to balances to display data loaded from your source system. When you navigate to the Oracle General Ledger Balances page after login validation, you can view a table listing all the general ledger accounts that contributed to the drilled value shown in the Hyperion EPM application for the specific period.

This table includes a breakdown of all general ledger accounts values with hyperlinks, enabling users to further drill into the Journal Lines page in Oracle General Ledger. Users can then view the associated journal entries for the selected Oracle General Ledger account.

When you navigate to PeopleSoft Enterprise Financial Management, the Ledger Inquiry page is displayed after login validation. Users can then view information on the Journal Inquiry page. See *PeopleSoft Enterprise General Ledger 9.1 PeopleBook* for additional information on drill through capabilities.

Configuring Oracle Data Integrator With FDMEE

FDMEE relies on Oracle Data Integrator as the engine that extracts data and metadata from the defined sources and then either populates the Performance Management Architect interface tables or populates the Classic applications with the extracted artifacts. The Oracle Hyperion Enterprise Performance Management System Oracle Hyperion Enterprise Performance Management System Installer installs ODI when FDMEE is installed. The installer also configures the ODI Master and Work Repository and ODI J2EE Agent as part of install process. In addition some manual configuration steps must be performed in Oracle Data Integrator before FDMEE can be used.

Note: You only need to configure ODI when you load data from a source other than a file. File-based data loads work out of the box without any additional ODI configuration.

- 1. Set up the data server based on the ERP Source System.
- 2. Set up the Physical Schemas.
- 3. Set up the ODI Context Code.

Set up the Data Server Based on the ERP Source System

You need to set up the appropriate data servers based on the ERP source system used to source metadata and/or data.

When importing from ERP sources (EBS/PeopleSoft/Fusion/JDE/SAP), set up the applicable data servers listed below:

- EBS_DATA_SERVER—For Oracle E-Business Suite (EBS) General Ledger
- PSFT_FMS_DATA_SERVER—For PeopleSoft General Ledger and Commitment Control
- PSFT_HCM_DATA_SERVER—For PeopleSoft Human Capital Management (HCM)
- FUSION_DATA_SERVER—For Fusion General Ledger
- JDE_DATA_SERVER—JD Edwards Enterprise (JDE) General Ledger
- SAP_SERVER—SAP FICO
- ➤ To update server connections information:
- 1 Access the Oracle Data Integrator Console.
- 2 Select the **Browse** tab.
- 3 Expand Topology.
- 4 Select the operating data server to update, and click Edit.

For example, select **EBS_DATA_SERVER** or **PSFT_FMS_DATA_SERVER**.

5 In Edit Data Servers, and then under JDBC Details, enter the JDBC driver in JDBC Driver.

For example, enter oracle.jdbc.OracleDriver

6 In JDBC URL, enter the JDBC URL address.

For example, enter jdbc:oracle:thin:@<host>:<port>:<sid>

- 7 In **User**, enter the user name.
- 8 In JDBC Password, enter the password.
- 9 Click Save.

Setting up Physical Schemas

- To update a physical schema:
- 1 Access the Oracle Data Integrator Console.
- 2 Select the Browse tab.
- 3 Expand Schemas.
- 4 Expand Physical Schemas.
- 5 Select the schema to update, and click Edit.

For example, select **EBS_DATA_SERVER** or **PSFT_FMS_DATA_SERVER**.

- 6 In Schema Name, enter the schema name in uppercase characters.
- 7 Click Save.

Caution!

Extensive problems can occur if you switch the ERP system connection information for the Physical Schema in the Oracle Data Integrator Topology Manager after you have completed initial configuration. For example, extensive problems can occur if you start using one physical schema (ERPTEST) pointing to ERP Test Instance1 in the Oracle Data Integrator Topology Manager and then change to a connection information in this physical schema to point to ERP Test Instance2 without first creating a new context in Oracle Data Integrator. The correct procedure is to create two physical schemas (ERPTEST1 and ERPTEST2) each pointing to a different ERP instance. Then, create two contexts and associate the appropriate physical schema to the logical schema in the context.

Setting up the ODI Context Code

You can set up the default "GLOBAL" ODI context code. The ODI context code refers to the context defined in Oracle Data Integrator. A context groups the source and target connection information.

- ➤ To set up the default "GLOBAL" ODI context code:
- 1 Access the Oracle Data Integrator Console.
- 2 Select the Browse tab.
- 3 Expand Contexts.
- 4 Select Global update, and then click Edit.
- In Edit Context Global, and then in Context Name, enter GLOBAL.
 Enter the word "GLOBAL" in uppercase characters only, that is, enter GLOBAL.
- 6 Select Default Context.
- 7 Click Save.

Setting up ODI to Integrate with JD Edwards

- To set up ODI to integrate with JD Edwards:
- 1 Download the JD Edwards EnterpriseOne Data Access Driver (DAD) driver:
 - a. Select the Patches and Updates tab, and then select JD Edwards Patches.
 - b. From JDEdwards, enter: EnterpriseOne Tools Releases.
 - c. In Release, select All Releases.
 - d. In **Platform**, select **Multiplatform**.
 - e. In Description (*text*), select *Data Access Driver*.

- f. Select the **License Agreement** check box.
- g. Click Search.
- h. Click the plus sign (+) to add **Tools 9.1 Data Access Driver** to the Download Basket.
- i. Click the Item(s) hyperlink and download the tools release specific item.

2 To copy the DAD Drive:

- a. Unzip the archive file to a temporary directory.
- b. Extract the JAR file DADriver EAR.jar file.
- c. Copy the extracted contents of **DADriver_EAR.jar** to the following directory: EPM MIDDLEWARE HOME\odi\odi misc.

Typically, EPM_MIDDLEWARE_HOME is C:\Oracle\Middleware

- 3 Request that the JD Edwards system administrator to provide you with the following files:
 - jdbj.ini
 - jas.ini
 - jdelog.properties

The three files above can be generated when you register the JD Edwards EnterpriseOne Data Access Driver using the JD Edwards Server Manager.

4 Copy the jas.ini file, jdbj.ini file, and jdelog.properties file from the JDE Application Server to:

```
EPM_MIDDLEWARE_HOME\odi\odi_misc directory.
```

Typically, EPM_MIDDLEWARE_HOME is C:\Oracle\Middleware

- Ensure that the **security server** of the jas.ini file is set correctly.
- 6 Review and edit the jdbj.ini file as follows:
 - a. If the JD Edwards application database is Oracle, update the location to the **tnsnames.ora** file.
 - b. Encrypt the [JDBj-BOOTSTRAP SESSION] stanza, and then the password value.
 - c. Modifications to the .INI file password encryption can only be accomplished through the use of Server Manager.
 - d. Set the value of following parameters under [JDBj-RUNTIME PROPERTIES] stanza to:

```
resultSetTimeout=-1
transactionTimeout=-1
usageExecutionThreshold=20000
usageResultSetOpenThreshold=120000
usageTracking=false
msSQLQueryTimeout=1800000
```

7 Edit the jdelog.properties file and set the log level as needed.

In a production environment, set the log level to SEVERE so that only severe errors are reported in the log file.

- 8 Ensure you can access the server specified in the jdbj.ini file from the ODI Server.
- 9 Run the script updateODIEAR.bat / sh from the following directory: EPM_ORACLE_HOME \products\FinancialDataQuality\bin

Before running the script, make sure that environment variables JAVA_HOME and TEMP are set correctly.

Typically the EPM_ORACLE_HOME directory is C:\Oracle\Middleware\EPMSystem11R1

10 If you are using JDE Tools 9.1.2 or 9.1.3, then run the script <code>copyfilesJDE912.bat</code> / sh from the following directory:

EPM_ORACLE_HOME \products\FinancialDataQuality\bin

If you are using JDE Tools 9.1.4 or higher, then run the script copyfilesJDE914.bat / sh from the following directory:

EPM_ORACLE_HOME \products\FinancialDataQuality\bin

11 If JD Edwards is using Microsoft SQL Server or IBM databases, then download the JDBC driver and copy it to the: <DOMAIN HOME>/lib directory.

Typically this is the $C:\Dracle\Middleware\user_projects\domains\EPMSystem \lib directory.$

Refer to the JD Edwards Tools documentation for the supported versions of the JDBC driver for the database platform.

Security

FDMEE supports the following roles:

FDMEE Roles	Tasks per Role
Administrator	Manages applications and performs any action in FDMEE. When you login with the Administrator role, all links are visible in the Tasks pane.
Create Integration	Creates FDMEE metadata and data rules. If you have the Create Integration role, you can: Create, delete and edit metadata rules and data rules View process details Perform period mappings Note: You cannot run rules or view, create, edit, or delete source system registrations, target system registrations or source accounting entities. When you login with the Create Integration role, these links are visible in the Tasks pane: Data Load, Member Mapping, HR Data Load, Metadata, and Process Detail.

FDMEE Roles	Tasks per Role
Drill Through	Controls the ability to drill through to the source system.
	In FDMEE, this role controls whether you can drill to the FDMEE landing page, which controls drilling to the source system.
	Note: Drill through is not supported for Financial Management journals and intercompany transactions.
Run Integration	Runs existing FDMEE metadata and data rules. If you have the Run Integration role, you can:
	Run metadata rules or data rules
	View process details
	Note: You cannot view, create, edit, or delete source system registrations, target system registrations or source accounting entities.
	FDMEE users who need to extract data from Oracle or PeopleSoft Enterprise Financial Management must be granted this role to enable them to run data rules.
	When you login with the Run Integration role, these links are visible in the Tasks pane: Data Load Workbench, Data Load, Member Mapping, HR Data Load, Metadata, and Process Detail.
GL Write Back	Enables data write-back to the ERP source system.
	When you login with the GL Write Back role, only the Write-Back Workbench, Write-Back Rule and Write-Back Mapping links are visible in the Tasks pane.
Intermediate 2-9	Roles 2-9 for intermediate levels are defined by the administrator.

Note: FDMEE users are allowed to define mappings for only target applications in which they have access.

Integration Process Overview

There are three ways that you can use FDMEE to integrate metadata and data from your ERP source system:

- "Extracting General Ledger Data" on page 28
- "Writing Back Data from EPM Applications" on page 29
- "Extracting Human Resource Data" on page 29

Extracting General Ledger Data

Follow this process to extract general ledger metadata and data and push into target EPM applications:

- 1. Register source systems in FDMEE by adding details for Oracle Data Integrator and FDMEE, which are specific to the source system.
- 2. Register target applications for use with FDMEE.
- 3. Select the Source Accounting Entities.

- 4. Define import formats and locations.
- 5. Create metadata rules.
- 6. Create period mappings for Year and Period dimensions.
- 7. Create category mappings for the Scenario dimension.
- 8. Create the member mappings and data load rules.
- 9. Run the metadata rules to import metadata into Financial Management or Planning applications.
 - If you use Performance Management Architect, you also deploy or redeploy the applications.
- 10. Run data rules to extract data from the source system and push into target applications. Data and/or metadata is staged in the FDMEE staging tables, extracted from the source system, and loaded into the target application.
 - The data loaded is used for multiple purposes by the respective target applications (Planning, Financial Management, or Essbase). In addition, the sourced data can also be used for drill through from web forms in the applications or Smart View and Financial Reporting.

Writing Back Data from EPM Applications

Follow this process to write-back data from EPM applications to your general ledger source system:

- 1. Perform steps 1-5 and step 7–8 in "Extracting General Ledger Data" on page 28.
- 2. Define write-back mappings for required segments or chartfields.
- 3. Create data write-back rules.
- 4. Run data write-back rules to push the data from supported target applications (Planning, Essbase aggregate storage, and Essbase block storage) into your general ledger source system.
 - **Note:** You cannot write-back data to SAP General Ledger and PeopleSoft Human Capital Management source systems.
- 5. Load the data into E-Business Suite or PeopleSoft Enterprise Financial Management by running a process in Oracle General Ledger or PeopleSoft General Ledger.

Extracting Human Resource Data

Follow this process to extract metadata and data from human resource source systems into Public Sector Planning and Budgeting target applications:

- 1. Register Source systems in FDMEE by adding details for Oracle Data Integrator and FDMEE, which are specific to the PeopleSoft Human Capital Management source system.
- 2. Register your target application.
- 3. Select the Source Accounting Entities (Business Units).

- 4. Create the human resource data load rules.
- 5. Run human resource data rules to extract data and/or metadata from the source system and push into target applications.

Data and/or metadata is staged in the FDMEE staging tables, extracted from the PeopleSoft Human Capital Management source system, and loaded into the target application.

Navigating FDMEE

Subtopics

- Toolbars
- Help Menu
- Task Pane Options
- Working with Data in Grids
- FDMEE User Interface Elements
- Advanced Search Options
- Using the POV Bar

From EPM Workspace, you can access FDMEE from the Navigate menu. (Navigate, Administer, Data Management)

Toolbars

The Standard toolbar is used for common Oracle Hyperion Enterprise Performance Management Workspace features. For additional information, see the *Oracle Enterprise Performance Management Workspace User's Guide*.

Help Menu

You use the Help menu to access FDMEE online help, Oracle technical support, the EPM documentation located on the Oracle Technology Network, Oracle website, and information about FDMEE.

Task Pane Options

The Tasks pane is a resizeable window located to the left of FDMEE workspace. It provides easy access to FDMEE options and features. The Tasks pane consists of two tabs:

- Workflow
- Setup

Workflow Tasks

From the Workflow tab, you can integrate metadata and data from an Enterprise Resource Planning (ERP) source system into an Enterprise Performance Management (EPM) target application:

You can also load data from file and other source systems.

- Data Load
 - Data Load Workbench
 - o Data Load Rule
 - Data Load Mapping
 - o Logic Group
 - Check Rule Group
 - Check Entity Group
- Metadata—Metadata Rule
- Write Back
 - Write Back Workbench
 - Write Back Rule
 - Write Back Mapping
- Other
 - Batch Execution
 - Report Execution
- Monitor—Process Details

Setup Tasks

From the Setup tab you can administer source and target systems, specify report and batch definitions, and manage application settings.

Available tasks:

- Configure
 - System Settings
 - Application Settings
 - Security Settings
 - User Settings
- Register
 - Source System
 - Target Application
 - o Source Accounting Entity

- Source Adapter
- Integration Setup
 - o Import Format
 - Location
 - Period Mapping
 - Category Mapping
- Data Load Setup
 - o Logic Group
 - Check Rule Group
 - Check Entity Group
- Scripts
 - Script Editor
 - Script Registration
- Reports
 - Query Definition
 - Report Definition
- Batchr—Batch Definition

Working with Data in Grids

Most screens display data in one or more grids. To manipulate grid data, perform one or more actions:

- To add a record, click **Add**.
- To delete, select a record and click **Delete**.
- To delete all records in a grid, click **Delete All**.
- To edit a record, click within its cell, and start typing. When applicable, you can also select the value to edit, and click ...
- To search items in a column, enter the search value in the blank field above the column of the value and press **Enter**. If the value is matched, it is displayed as the first item.
- To cancel all changes made to a row, select the row and click **Cancel**.
- To save all changes made to a row, select **Save**.

FDMEE User Interface Elements

The following elements are common on FDMEE pages.

 Table 1
 Elements Common on FDMEE Pages

Button	Description
View ▼	Customize your view. Options include:
	Columns—You can choose "Show All" to display all columns or choose individual columns to display.
	Detach—Use to detach the column grid. When you detach the grid, the columns display in their own window. To return to the default view, select View, and then, click Attach or click the Close button.
	Reorder Columns—Use to change the order of the columns that display. You can select a column and then use the buttons on the right to change the column order.
☑ Detach	Use to detach the column grid. When you detach the grid, the columns display in their own window. To return to the default view, select View , and then, click Attach or click the Close button.
Refresh	Refreshes the data. For example if you submit a rule, you can refresh to check when status changes from Running to Complete.
	Note: The Refresh button does not display on the FDMEE setup screens.
	Use to toggle the filter row. You can use the filter row to enter text to filter the rows that display for a specific column.
	You can enter text to filter on, if available, for a specific column, and then press [Enter]. For example, on the Process Details page, to view only processes that failed, you can enter "FAILED" in the Status text box.
	The Query by Example button displays on the following FDMEE screens: Target Application, Source Accounting Entities, Import Format, Location, Data Load Workbench, Write-Back Workbench, Process Details.
	To clear a filter, remove the text to filter by in the text box, then press [Enter].
	All text you enter is case sensitive.
Q	Use to select an artifact on a page, such as a target application, member, or general ledger responsibility. When you click the Search button, the Search and Select dialog box is displayed. In some cases you have advanced search options that enable you to enter additional search conditions. See "Advanced Search Options" on page 33.

Advanced Search Options

The Search button is common to many FDMEE pages. When you select the Search button, if the Advanced Search button is available, you can enter additional search conditions. The fields that display in the advanced search options differ depending on what artifact you are selecting. The following operators are supported:

- Starts with
- Ends with
- Equals
- Does not equal
- Less than
- Greater than
- Less than or equal to
- Greater than or equal to
- Between

- Not between
- Contains
- Does not contain
- Is blank
- Is not blank

Using the POV Bar

For the Data Load Workbench, the POV bar shows the current:

- Location
- Period
- Category
- Data Rule

Location KS7DIM_EBSTB Period Jan-08 Category Actual Rule KS7DIM_EBSTB Source File Target KS7DIM

By default only the data rule assigned to the Category POV is displayed.

The Source System and Target Application are displayed as context information.

Selecting the Location POV

When you log on, the location POV is set to the default location of the user. Users can view only locations that they have permission to access.

- To select another Location POV:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 Double click Location field.
- 3 In Select Point of View, and then in Location, enter a full or partial string for the new location, and then, click OK.
- 4 **Optional**: To search on another location, from the **Location** drop-down, click **More**, navigate to the location on the **Search and Select: Location** screen, and then, click **OK**.
- **Optional**: In **Select Point of View**, select **Set as Default** to use the new location as the default location. When a POV selection is set as a default, the user profile is updated with the default selection.
- 6 Click OK.

Setting the Period POV

The FDMEE administrator controls which accounting period is active for all users. This feature prevents users from inadvertently loading data into incorrect periods. When you log on to

FDMEE, the application identifies the global period value and automatically sets the POV to the current value.

- To select another Period POV:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 Double click Location field.
- 3 In Select Point of View, and then in Period, enter a full or partial string for the new period, and then, click OK.
- 4 **Optional**: To search on another period, from the **Period** drop-down, click **More**, navigate to the period on the **Search and Select**: **period** screen, and then, click **OK**.
- Optional: In Select Point of View, select Set as Default to use the new period as the default period.
 When a new POV selection is set as a default, the user profile is updated with the default selection.
- 6 Click OK.

Setting the Category POV

The FDMEE administrator controls the data category active for all users. This feature prevents users from inadvertently loading data to incorrect categories. .

- ➤ To select another Category POV:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 In Data Load Workbench, double click the Location field.
- 3 In Select Point of View, and then in Category, select the new category, and then, click OK.
- 4 **Optional**: In **Rule**, select the rule assigned to the Category POV.
- 5 Select Set as Default to use the new category as the default category.
 When a POV is set as a default, the user profile is updated with the default selection.
- 6 Click OK.

Administration Tasks

Predefining a List of Profiles

FDMEE uses a predefined list of profiles. You can define values for these profiles to accommodate various business needs. Profiles can be set at the following levels:

- System (applies to the entire system)
- Application (applies to specific target application)
- User (applies to a specific user)

• Security (Role, Report, Batch, Custom Script, and Location)

Setting System Level Profiles

Use the System Settings Option to update or clear System level profiles that apply to entire system. System settings can be defined only by users with "Admin" role privileges.

- To define system settings:
- 1 From the **Setup** tab, and then under **Configure**, select **System Settings**.
- 2 In System Settings, from Profile Type, select the specific profile to list on the System Settings screen.

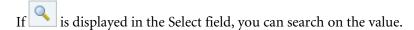
Available profile types:

- All
- File (In addition to file specific system settings, selecting the File profile type displays the "Create Application Folders" button. This feature instructs the system to create a folder structure in the path specified in the Application Root Directory setting).
- Other (Use to set profiles associated with an EPMA data source, User Language, User Interface Theme, and Default Check Report.)
- Point-of-View

Available profile types:

The profile type that you select determines the settings that you can add or modify on the screen.

3 Select the option and add the new value in Value.



System setting options are described in Table 2.

Note: When you install and configure FDMEE, Oracle Data Integrator is automatically installed and configured for you. The database for Oracle Data Integrator is in the same database as FDMEE and the Oracle Data Integrator agent deployed to the FDMEE Managed Server. You should be familiar with Oracle Data Integrator and review the Oracle Data Integrator documentation set before specifying the ODI detail.

4 Click Save.

Table 2 System Setting Profile Options

Profile Type	Profile Option	Profile Description
All	Includes all profile types	

Profile Type	Profile Option	Profile Description
File	Application Root Folder	Note: The Application Root Directory must be setup on the server, specified in this field, and the Create Application Folder options must be executed as first steps when using FDMEE.
		The Application Root folder identifies the root directory of the FDMEE application. This folder is located on the FDMEE server and functions as the root folder for all FDMEE activities. Based on this parameter, FDMEE saves log files, generated files and reports to the appropriate folder under this root directory. Parameters must be set up on the server separately from this setup step.
		When you select the File profile type, the System Setting screen displays the "Create Application Folders" button. This features instructs the system to create a folder structure in the path specified in this field. The folder structure is (with sub-folders in each):
		data
		inbox
		outox
		Within the inbox, locations are created when they are created in the Locations option. See also "FDMEE Application Architecture" on page 18.
		When you specify a folder at the application level, scripts are saved to this folder. This is especially important for event scripts that are different between applications. When you do not set up an application level folder, then you cannot have different event scripts by application.
		Note: Use a Universal Naming Convention (UNC) path for the application root folder when Financial Management and FDMEE are on separate servers. Share permissions on the folder must allow access to the DCOM user for read/write operations. Contact your server administrator to define the required UNC definition.
File	Create Location Folder	This setting instructs the system to create a location folder in the inbox when a location is created. Available values are Yes or No . Set this option once and then do not change it. This option is optional, but recommended.
File	Archive Mode	Specifies whether archived files are copied or moved to the archive location. Enter Copy , Move , or None .
		If you select <i>Copy</i> , then the file is left in the inbox.
		If you select <i>Move</i> , then the file is moved to the archive folder and deleted from the inbox.
		The folder named data is the archive folder.
		When the file is moved to the archive location, it is renamed as follows:
		<process id=""><year><month><day>.<original extension=""></original></day></month></year></process>
		For example, if the source file name is BigFile.csv, and it was loaded for the period Mar-07 with a period key of 03/01/2007, and if the process id was 983, then the resulting file name is 98320070301.csv.
File	Excluded File Upload Wild Cards	Specify file extensions that cannot be uploaded. Enter "*.*" to disallow all file uploads.
File	Batch Size	Specify the number of rows read at a time from the file to memory. This parameter is mainly used for performance. When data is loaded, this settings determines how many records are to stored in the cache. For example, when 1000 is specified; the system stores 1,000 records in cache, Similarly, when 5000 is specified, the system stores 5,000 records in cache and commit. This setting should be determined by Server Memory, and can be adjusted as needed,

Profile Type	Profile Option	Profile Description
File	File Character Set	Specify the method for mapping bit combinations to characters for creating, storing, and displaying text.
		Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41.
		Available character sets are available on the Search and Select screen by clicking
File	Encrypted	Specify the directory where the files that store passwords in encrypted form is located.
	Password Folder	This encrypted password folder is used in conjunction with the "Update Configuration File" button. For more information, see "Working with Batch Scripts" on page 260.
ODI	ODI Agent URL	Specifies the URL of the server where Oracle Data Integrator has been installed.
ODI	ODI User Name	Specifies the Oracle Data Integrator user name used to access the Oracle Data Integrator master repository. For example, enter Supervisor.
		This setting is defined automatically when ODI is configured, but it can be customized if necessary.
ODI	ODI Password	Specifies the Oracle Data Integrator database schema used to access the Oracle Data Integrator master repository. For example, enter Master,
		This setting is defined automatically when ODI is configured, but it can be customized if necessary.
ODI	ODI Execution Repository	Specifies the repository where all scenarios are stored. For example, enter ERPI_REF.
		This setting is defined automatically when ODI is configured, but it can be customized if necessary.
ODI	ODI Work Repository	Specifies the (execution) repository which contains run-time objects (for example, scenarios). The work repository can be linked with only one Master Repository.
		For example, enter FDMEE_WORK_REF.
ODI	ODI Master Repository Driver	Specifies the driver of the ODI master repository.
05.		This setting is defined automatically when ODI is configured, but it can be customized if necessary.
		You can click and select the driver from the Search and Select: ODI Master Repository Driver screen. For example, enter Oracle.jdbc.OracleDriver
ODI	ODI Master Repository URL	Specifies the URL of the server where the Oracle Data Integrator master repository is installed.
		This setting is defined automatically when ODI is configured, but it can be customized if necessary.
		You can click and select the URL from the Search and Select: ODI Master Repository URL screen. For example, enter jdbc:oracle:thin:@serverdatabase.oracle.com:1521:orcl
ODI	ODI Master	Specifies the Oracle Data Integrator master repository user name.
	Repository User	This setting is defined automatically when ODI is configured, but it can be customized if necessary.
ODI	ODI Master	Specifies the Oracle Data Integrator master repository password
	Repository Password	This setting is defined automatically when ODI is configured, but it can be customized if necessary.
Other	EPMA Data Source Name	Specifies the EPMA Data Source name for the EPMA Interface Data Source.

Profile Type	Profile Option	Profile Description
Other	User Language	Specify the system default language of the user version of FDMEE
		Note: FDMEE uses the user language to query the language data, for example, column titles, segment name, and so on.
		You can also click sto navigate to the language.
Other	User Interface Theme	The default theme contains all the colors, styles, and general-use icons that are displayed in the user interface.FDMEE uses BLAF+ as the default value.
		You can also click sto navigate to the theme
Other	Default Check Report	Specify the type of Check Report to use as the default check report. The following are pre-seeded check reports, but you can create a new one and specify it here:
		Check Report—displays the results of the validation rules for the current location (pass or fail status).
		Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods.
		Check Report by Val. Entity Seq.—Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group.
		Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown if warning criteria are met. This report does not show rules that passed the validation.
Other	Default Intersection Report	Specify the type of Intersection Check Report to use as the default intersection check report at the system level. Intersection reports identify data load errors and are generated as part of the data validation step in the Data Load Workbench. The reports are available in two formats: Dynamic Column or Fixed Column format. The Fixed Column displays up to four custom dimensions.
Other	Batch Timeout in Minutes	When a batch job is run in sync mode (immediate processing), specify the maximum time the job can run. In sync mode, FDMEE waits for the job to complete before returning control.
Other	Enable Event Script Execution	Select Yes to enable the execution of application events such as before loading data (BefLoad) or after validation (AftValidate). Select No to disable the execution of application events.
Other	SQL Server Database Provider	Specify the name of the SQL Server database provider.
		Available SQL Server database providers:
		SQLOLEDB
		SQLNCLI10 (SQL Server 2008)
		SQLOLEDB
Other	Log Level	Specify the level of detail displayed in the logs. A log level of 1 shows the least detail. A log level of 5 shows the most detail.
		Logs are displayed in Process Details by selecting the Log link.
POV	Default POV	Specifies the default POV location.
	Location	These preferences take precedence when no equivalent settings are in Application Settings or User Settings
		You can also click to navigate to the default POV location from the Search and Select: Default POV Location screen.

Profile Type	Profile Option	Profile Description
POV	Default POV Period	Specifies the default POV Period. These preferences take precedence when no equivalent settings are in Application Settings or User Settings. You can also click to navigate to the default POV Period from the Search and Select: Default POV Period screen.
POV	Default POV Category	Specifies the default POV Category. These preferences take precedence when no equivalent settings are in Application Settings or User Settings. You can also click to navigate to the default POV Category from the Search and Select: Default POV Category screen.
POV	Global POV Mode	When this is set to Yes , other POVs (Application Level and User Level POVs) are ignored.

Setting Application Level Profiles

Use the Application Settings option to update or clear application level profiles that apply to the target application.

- To set an application level profile:
- 1 Select the **Setup** tab, and then under **Configure**, select **Application Settings**.
- 2 In **Application Settings**, from the **Target Application** drop-down, select the target application to which the application profile applies.
- 3 Select application level profile settings.
- 4 **Optional:** To clear a setting, select the value, and then click **Delete**.

The value is removed but is deleted only when you save it.

5 Click Save.

 Table 3
 Application Level Profile Options

Option	Description
Application Root Folder	The Application Root folder is the root folder for storing all files used to load data to this EPM application. You can use a separate root folder for each EPM application.
	Based on this parameter, FDMEE saves log files, generated files and reports to the appropriate folder under this root directory. Parameters must be set up on the server separately from this setup step.
	Selecting the Create Application Folder button instructs the system to create a folder structure in the path specified in this field. The folder structure is (with sub-folders in each):
	data
	inbox
	outox
	When you specify a folder at the application level, scripts are saved to this folder. This is especially important for event scripts that are different between applications. When you do not set up an application level folder, then you cannot have different event scripts by application.
	Note: Use a Universal Naming Convention (UNC) path for the application root folder when Financial Management and FDMEE are on separate servers. Share permissions on the folder must allow access to the DCOM user for read/write operations. Contact your server administrator to define the required UNC definition.
File Character	Specify the method for mapping bit combinations to characters for creating, storing, and displaying text.
Set	Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41.
	Available character sets are available on the Search and Select screen by clicking 🥄 .
Default POV	Specify the default POV location.
Location	You can also click to navigate to the default POV location from the Search and Select: Default POV Location screen.
Default POV	Specify the default POV Period.
Period	You can also click sto navigate to the default POV Period from the Search and Select: Default POV Period screen.
Default POV	Specify the default POV Category.
Category	You can also click how to navigate to the default POV Category from the Search and Select: Default POV Category screen.
User Language	Specify the application default language of the user version of FDMEE.
	You can also click stonavigate to and select the language.
User Interface Theme	Set the Oracle design pattern for the applications. FDMEE uses BLAF+ as the default user interface value.
пспс	You can also click sto navigate to the theme.
Default Intersection Report	Specify the type of Intersection Check Report to use as the default intersection check report at the application level. Intersection Check reports identify data load errors and are generated as part of the data validation step in the Data Load Workbench. The reports are available in two formats Dynamic Column or Fixed Column format. The Fixed Column displays up to four custom dimensions.

Option	Description
Default Check Report	Specify the type of Check Report to use as the default check report at the application level. The following are pre-seeded check reports, but you can create a new one and specify it here:
	Check Report—Displays the results of the validation rules for the current location (pass or fail status).
	Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods.
	Check Report by Val. Entity Seq.—Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group,
	Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown if warning criteria are met. This report does not show rules that passed the validation.
Enable Event Script Execution	Select Yes to enable the execution of application events such as before loading data (BefLoad) or after validation (AftValidate). Select No to disable the execution of application events.
Log Level	Specify the level of detail displayed in the logs. A log level of 1 shows the least detail. A log level of 5 shows the most detail.
	Logs are displayed in Process Details by selecting the Log link.

POV Locking

Point of View (POV) Lock feature prevents data from being loaded to a selected POV by locking all locations related to a target application for a given period and category. When you select the "Lock POV" button, a pop-up is displayed with populated Category and Period values based on the profile. You can update the category and period if desired and then lock the POV.

An "Unlock POV" is also available so that you can unlock a locked POV. The POV lock is referenced in:

- Data Load Workbench
- Data Load Rule
- Batch Execution
- ➤ To lock a POV:
- 1 Select the **Setup** tab, and then under **Configure**, select **Application Settings**.
- In **Application Settings**, and then from the **Target Application** drop-down, select the target application to which the application profile applies.
- 3 Click LOCK POV.

The Lock POV for All Locations screen is displayed.

- 4 **Optional:** From **Category**, select the category to update.
- 5 **Optional:** From **Period**, select the period to update.
- 6 Click OK.

All locations for the selected target application are be locked.

To unlock the locked POV, click Unlock.

Setting User Level Profiles

Use the User Settings option to update or clear user level profiles that apply to the user.

Note: When the Global mode is defined, then user level profiles for the POV are not applicable.

- To set a user level profile:
- 1 Select the **Setup** tab, and under **Configure**, select **User Settings**.
- 2 In **User Setting**, select the options to add or modify.
- **Optional:** To clear a setting, select the value and from your keyboard, click **Delete**.

The value is removed, but it is deleted only when you save it.

4 Click Save.

Table 4 User Level Profile Settings

Option	Description
File Character	Specify the method for mapping bit combinations to characters for creating, storing, and displaying text.
Set	Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41.
	Available character sets are available on the Search and Select screen by clicking 🥄 .
Default POV Location	Specify the default POV location.
Location	You can also click \(\frac{\leftrightarrow}{\leftrightarrow} \) to navigate to and select the default POV location from the Search and Select: Default POV Location screen.
Default POV	Specify the default POV Period.
Period	You can also click to navigate to and select the default POV Period from the Search and Select: Default POV Period screen.
Default POV	Specify the default POV Category.
Category	You can also click \(\frac{1}{2} \) to navigate to and select the default POV Category from the Search and Select: Default POV Category screen.
User Language	Select the default language of the user versionFDMEE user interface.
	You can also click 4 to navigate to and select the language.
User Interface	Specify the default theme of the user version of the FDMEE user interface.
Theme	You can also click sto navigate to the theme.

Option	Description
Default Check Report	Specify the type of Check Report to use as the default check report at the user level. The following are pre-seeded check reports, but you can create a new one and specify it here:
	Check Report—Displays the results of the validation rules for the current location (pass or fail status).
	 Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods.
	• Check Report by Val. Entity Seq.—Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group,
	• Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown if warning criteria are met. This does not show rules that passed the validation.
Default Intersection Report	Specify the type of Intersection Check Report to use as the default intersection check report at the user level. Intersection reports identify data load errors and are generated as part of the data validation step in the Data Load Workbench. The reports are available in two formats Dynamic Column or Fixed Column format. The Fixed Column displays up to four custom dimensions.
Log Level	Specify the level of detail displayed in the logs. A log level of 1 shows the least detail. A log level of 5 shows the most detail.
	Logs are displayed in Process Details by selecting the Log link.

Setting Security Options

Within FDMEE, administrators can secure almost any user interface and report feature. FDMEE supports five levels of security:

- Role level security—Controls access to components of the user interface that each user can access.
- Report security—Controls the reports that can be executed based on the report groups assigned to a role.
- Batch security—Controls the batches that can be executed based on the batch groups assigned to a role.
- Custom scripts security—Controls the custom scripts that can be executed based on the custom script groups assigned to a role.
- Location security—Controls access to locations

Security levels are applied to users. Role and Location security levels assigned to users are compared at runtime. If a user is assigned a level that is equal the level assigned to the feature that the user is trying to access, the feature is available to the user.

Role Level Security

FDMEE security enables you to customize user access to user interface functions using the concept of roles. Roles are permissions that grant user access to functions. In FDMEE, default roles are assigned to functions that aggregate and tailor specific requirements. After the functions are assigned to a role, the corresponding role is mapped to users when provisioning users in

Shared Services. The process of granting roles to users is described in the *Oracle® Enterprise Performance Management System User and Role Security Guide*.

- ➤ To add role level security:
- Select the Setup tab, and then under Configure, select Security Settings.
- 2 In Security Setting, select the User Interface tab.
- 3 From the Role drop-down, select the role to which to assign access.

A list of roles is described in Table 5, "Role and Descriptions," on page 45.

The role category determines the display of functions associated with the selected role.

- 4 In the **Function** field, select the function to assign to the role.
- 5 Click Save.

Table 5 Role and Descriptions

Role	Description
Administrator	Grants access to all FDMEE functions.
Create Integration	Creates FDMEE metadata and data rules.
Run Integration	Runs FDMEE metadata and data rules and fills out runtime parameters. Can view transaction logs.
Drill Through	Controls whether you can drill to the FDMEE landing page, which controls drilling to the source system.
Write-Back	Enables data write-back to the ERP source system.
HR Integration	Runs Human Resource data rules and fills out runtime parameters. Can view transaction logs.
Intermediate 2-9	Roles for intermediate levels are defined by the administrator.

Defining User Interface Security

- To add batch security:
- 1 Select the **Setup** tab, and then under **Configure**, select **Security Settings**.
- 2 From the **Role** drop-down, select the role to which to assign batch security.

A list of roles is described in Table 5, "Role and Descriptions," on page 45.

- 3 Select the User Interface tab.
- 4 From Function, select the user interface function to which to assign security.
- 5 Click Save.

Defining Report Security

To define report security, you assign reports of a selected type to a group (see "Adding Report Groups" on page 325) Next you assign the report group to a role. The role has access to all reports in the groups at execution time.

- To add report level security:
- Select the Setup tab, and then under Configure, select Security Settings.
- 2 From the **Role** drop-down, select the role to which to assign the report security.
 - A list of roles is described in Table 5, "Role and Descriptions," on page 45.
- 3 Select the Report tab.
- 4 From **Report Group**, select the report group to which to assign report security.
- 5 Click Save.

Defining Batch Security

To define batch security, you assign batches of a selected type to a group (see "Adding a Batch Group" on page 254). Next you assign the batch group to a role. The role has access to all batches in the groups at execution time.

- ➤ To add batch security:
- Select the Setup tab, and then under Configure, select Security Settings.
- 2 From the Role drop-down, select the role to which to assign batch security.
 - A list of roles is described in Table 5, "Role and Descriptions," on page 45.
- 3 Select the Batch tab.
- 4 From **Batch Group**, select the batch group to assign batch security.
- 5 Click Save.

Defining Custom Script Security

To define custom scrip security, you assign custom scripts of a selected type to a group (see "Adding a Custom Script Group" on page 288). Next you assign the custom scripts group to a role. The role has access to all custom scripts in the groups at execution time.

- To add role level security:
- 1 Select the Setup tab, and then under Configure, select Security Settings.
- 2 From the Role drop-down, select the role to which to assign custom script security.
 - A list of roles is described in Table 5, "Role and Descriptions," on page 45.
- 3 In Security Setting, select the Custom Security tab.
- 4 Select the **Custom Script** tab.
- 5 From Custom Script Group, select the custom script group to which to assign custom script security.
- 6 Click Save.

Defining Location Security

Location security (user access to locations) for FDMEE is configured and enforced by options on the Location Security Settings tab. You define the user groups to create for each location. When a Location is created or updated, then you can create as many groups as defined in the system settings for the Location. Additionally, a "Maintain User Groups" enables you to create user groups in mass for the location.

Several dependent processes must occur before Location Security is fully implemented:

1. When a Location is created, one or more User Groups are created automatically in Shared Services.

The user group contains the name of the Location and additional prefix and suffix information based on the user preference. In addition, roles are provisioned for User Groups.

- 2. The administrator provisions the users to the User Groups.
- When your log in, FDMEE determines the groups assigned to the user.
 Based on the name of the group, FDMEE determines the accessible locations.
- 4. The POV region filters the locations based on user access.

Note: If the web services and batch scripts are used, then location security is still maintained and enforced.

- To display the Location Security tab:
- 1 Select the **Setup** tab, and then under **Configure**, select **Security Settings**.
- 2 Select the Location Security tab.
- To add a user group for location security:
- 1 From the **Setup** tab, and then under **Configure**, select **Security Settings**.
- 2 Select the Location Security tab.
- 3 In the Location summary grid, click Add.

A *LOCATION* name row is added. When the group is saved, the Group name is in the form of Prefix_Location_Suffix, for example, FDMEE_*LOCATION_*DATA.

The prefix and suffix help identify groups in Common Shared Services (CSS).

4 In the Security Setting Details grid, enter a description of the user group in the Description field.

For example, enter: Group for Creating and Running Integration.

5 In the Prefix field, enter FDMEE.

When the group is saved, the prefix is prepended to the group name.

6 In the Suffix field, select the name of the function or rule that the user is provisioned to access.

For example, specify:

• Data Rule Metadata Rule (Run Integration role)

- Write-back (Write-back role)
- HR Rule (HR Integration role)
- Create Integration
- Drill Through

When the group is saved, the suffix is appended to the group name.

- 7 Select the list of roles provisioned for the user group by selecting the appropriate roles:
 - Create Integration
 - Drill Through
 - Run Integration
 - HR Integration
 - Write Back
 - Intermediate 2-9

The list of roles is described in Table 5, "Role and Descriptions," on page 45.

- 8 Click Save.
- 9 To create users groups in mass for the location, click Maintain User Groups.
- To disable security by location:
- 1 From the Setup tab, and then under Configure, select Security Settings.
- 2 Select the Location Security tab.
- 3 From the Location grid, select the location to disable, and then, click Disable Security by location.
- 4 Click Save.

Setting up Source Systems

Subtopics

- Registering Source Systems
- Deleting Registered Source Systems
- Editing Registered Source System Details
- Adding file-based Data Load Definitions

In some cases, you may have multiple general ledger or human resource systems. You can use FDMEE to extract data and metadata from any instance.

For information on the source systems that FDMEE supports, see the *Oracle Hyperion Enterprise Performance Management System Certification Matrix*.

1. Register a source system. See "Registering Source Systems" on page 49.

2. Edit source system settings as necessary. See "Editing Registered Source System Details" on page 51.

For information on removing a registered source system, see "Deleting Registered Source Systems" on page 51.

Before you begin using FDMEE, you must register your source systems. Follow this process:

- 1. Register the source system. See either "Registering Source Systems" on page 49.
- 2. Edit source system settings as necessary. See "Editing Registered Source System Details" on page 51.

For information on removing a registered source system, see "Deleting Registered Source Systems" on page 51.

Note: For information on viewing FDMEE processes or jobs, see "Viewing Process Details" on page 180.

Registering Source Systems

The source system page displays all registered source systems in a table in the Summary pane. By default, the following columns are displayed:

- Name—Name of the source system
- Type—Shows the type of supported source system
- Description—The description that you entered when you registered the source system
- Drill URL—The drill URL you entered when you registered the source system
- To add a file based source system:
- 1 Select the **Setup** tab, and then under **Register**, select **Source System**.
- 2 Enter the source system details:
 - a. In **Source System Name**, enter the source system name.
 - b. In **Source System Description**, enter a description of the source system.
 - c. In **Source System Type**, select **File**.
 - d. Enter the Drill URL.

For example, http://machinename.us.oracle.com:6362

The Drill-Through URL identifies the URL to use for drilling through to any source system.

The Drill-Through URL identifies the URL to use for drilling through. Users can drill through to Oracle General Ledger or PeopleSoft Enterprise Financial Management from an Enterprise Performance Management (EPM) system application that displays data loaded from the source system. When you click a hyperlink, you can navigate to the Oracle General Ledger Balances page or PeopleSoft Enterprise Financial Management Inquiry page.

For JD Edward source systems, uses can drill through to the JD Edwards balances page.

3 Click Save.

After you add a source system, you can select the source system in the table, and the details are displayed in the lower pane.

- To add an ERP source system:
- 1 Select the **Setup** tab, and then under **Register**, select **Source System**.
- 2 In Source System, click Add.
- 3 Enter the source system details:
 - a. In **Source System Name**, enter the source system name.
 - b. In **Source System Description**, enter a description of the source system.
 - c. In **Source System Type**, select the source system type.

Available source systems

- E-Business Suite Release 11i
- PeopleSoft Financials Release 9
- Peoplesoft HCM Release 9
- Fusion Applications
- SAP
- SAP BW (Business Warehouse)
- JD Edwards Enterprise One
- File
- Others
- d. Enter the Drill URL.

The Drill-Through URL identifies the URL to use for drilling through. Users can drill through to Oracle General Ledger.

The URL is used to launch Fusion, E-Business Suite or PeopleSoft. For example, http://machinename.us.oracle.com:6362

The Drill-Through URL identifies the URL to use for drilling through. Users can drill through to Oracle General Ledger or PeopleSoft Enterprise Financial Management from an Enterprise Performance Management (EPM) system application that displays data loaded from the source system. When you click a hyperlink, you can navigate to the Oracle General Ledger Balances page or PeopleSoft Enterprise Financial Management Inquiry page.

For JD Edward source systems, uses can drill through to the JD Edwards balances page.

e. In **ODI Context Code**, enter the context code.

The ODI context code refers to the context defined in Oracle Data Integrator. A context groups the source and target connection information.

The default context code is GLOBAL.

4 **Optional:** If you use PeopleSoft's Commitment Control functionality, select **Enable Commitment Control**.

See *Appendix E: PeopleSoft's Commitment Control* for more information.

5 Click Save.

After you add a source system, you can select the source system in the table, and the details are displayed in the lower pane.

After you register a source system, you must initialize the source system. Initializing the source system fetches all metadata needed in FDMEE, such as ledgers, chart of accounts, and so on. It is also necessary to initialize the source system when there are new additions, such as chart of accounts, segments/chartfields, ledgers, and responsibilities.

6 To initialize a source system, click **Initialize**.

Note: Depending on the size of the source system, it may take several minutes to initialize.

Deleting Registered Source Systems

You can delete registered source systems if you do not plan to use the source system with FDMEE.

Caution!

Use caution when deleting registered source systems! Part of the procedure for deleting a source system is to delete the target application. When you delete the target application, other artifacts are deleted. When you delete a registered source system, the source system is removed from the **Source System** screen and all import formats, locations, metadata rules, and data rules associated with the source system are removed.

- To remove a registered source system:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 Use the Target Application page to remove all target applications that have rules or mappings to the source system.
- 3 Select the **Setup** tab, and then under **Register**, select **Source System**.
- 4 In **Source System**, select the source system to remove and then, click **Delete**.

Tip: To undo a deletion, click **Cancel**.

5 Click OK.

Editing Registered Source System Details

Sometimes, source system details change. You can edit the source system details as needed. Keep in mind that after you add a source system type, you should not modify it.

- To edit registered source system settings:
- 1 Select the **Setup** tab, and then under **Register**, select **Source System**.
- 2 Select the source system.
- 3 Edit the source system details or ODI details as necessary.
- 4 Click Save.

If you make metadata changes in the source system, for example, you add a new segment, chartfield values or hierarchies, you must initialize the source system.

5 Click Initialize.

Adding file-based Data Load Definitions

Source systems of the type "file" are used in import formats to import data from fixed and delimited files.

FDMEE creates a file-based data load system automatically. If you need to create an alternate file-based data load source system, follow the procedure below.

To use file-based import formats, you must define a file-based data load.

- To add a file-based data load definition:
- 1 Select the Setup tab, and then under Register, select Source System.
- 2 In Source System, click Add.
- 3 Enter the source system details:
 - a. In **Source System Name**, enter the file-based data load system name.
 - b. In **Source System Description**, enter a description.
 - c. In **Source System Type**, select **File**.
 - d. In **Drill Through URL**, specify the URL that identifies the URL to use for drilling through.
 - e. In **ODI Context Code**, enter the context code.

The ODI context code refers to the context defined in Oracle Data Integrator. A context groups the source and target connection information.

4 Click Save.

After you add a file-based data load system, select the source system in the table. The details of the system are displayed in the lower pane.

Working with Source Adapters

A source adapter is an integration framework in FDMEE that enables you to extract data from source system in a flexible and customizable manner. A source adapter consists of two components:

- Oracle Data Integrator (ODI) Project—Contains the code to extract and load the data from the source system to the FDMEE staging tables.
- Adapter Definition XML—Contains the definition of the integration. It contains three components: source columns, parameters and the drill URL.

Three pre-packaged integrations are delivered using this framework:

- SAP ERP Financials
- Open interface to load from any source system
- JD Edwards

Using a Pre-packaged Integration for SAP

For pre-packaged integrations for SAP, review the instructions in the readme to download the necessary information from Oracle's integration partner. Oracle delivers the Oracle Data Integrator (ODI) Project and an Adapter Definition XML file. For the SAP integrations, the integration partner delivers the ODI Project and Adapter XML.

- To use the pre-packaged integration:
- 1 Use the ODI Console to import the Project & Model definitions into the Work Repository setup for FDMEE.
- 2 Copy the Adapter Definition XML to a location under the directory that has been set up as the Application Root directory in System Settings.
- 3 Use the Source Adapter screen to import the Adapter XML file.

To import a file:

- a. Select the **Setup** tab, and then under **Configure**, select **Source Adapter**.
- b. In the Source Adapter, select Import.
- c. In **Select file to import**, navigate to the file to import, and then, click **OK**.

The file name depends on the adapter desired. For example, select:

- SAP GLNew Adapter.xml
- SAP_GLClassic_Adapter.xml
- SAP_PC_Adapter.xml
- SAP_CC_Adapter.xml
- SAP_AP_Adapter.xml
- SAP_AR_Adapter.xml
- d. Optional: To browse for a file, select **Upload**.
- e. In **Select a file to upload**, click **Browse** to navigate to the file to import, and then, click **OK**.
- 4 Create an import format of new type **Source Adapter** that defines the mapping between the source columns (identified in the source adapter) and the target application dimensions.

5 Define the Locations and Data Rule.

When you execute the data rule, the new ODI Scenario associated with the import format is used to extract the data from the source and stage it in the FDMEE staging table (TDATASEG).

Adding Source Adapter Definitions

Use the Source Adapter screen to review prepackaged adapters or to customize prepackaged adapters. Do not directly modify the prepackaged adapters. As Oracle ships updates to the prepackaged adapter, the changes you make are overwritten. Always make a copy of the prepackaged adapter and make the necessary changes. You can also use this screen to create new custom adapters for custom load integrations.

The Source Adapter screen includes three sections:

- Source adapter information summary. See "Defining Source Adapter General Information" on page 54.
- Source adapter detail. See "Defining Source Adapter Detail" on page 57.
- Three sub-tabs from which you select: source columns information, parameters to pass to the ODI scenario, and URLs for drill through.

For more information, see:

- o "Selecting Source Columns" on page 58
- o "Defining Parameters" on page 58
- o "Defining URLs for Drill Through" on page 59

Defining Source Adapter General Information

Use the Source Adapter Summary section to view, define, delete, export, import, and copy summary information about a source adapter.

Adding a Source Adapter

- To add a source adapter:
- Select the Setup tab, and then under Register, select Source Adapter.
- 2 In Source Adapter, from the Source Adapter summary task bar, click Add.

Blank entry rows are added to the Source Adapter summary section.

Complete the following steps in the Source Adapter details section.

3 In Adapter Key, enter a user defined identifier for the adapter.

For example for an Open Interface source adapter, enter **OPENINT**. The value in this field cannot be modified after it has been created.

4 In Adapter Name, enter a user defined name for the adapter.

For example, enter Open Interface Adapter.

- 5 In **ODI Object Name**, enter the name of the package or scenario.
- 6 In ODI Object Type, select either Package or Scenario.
 - Package—A Package is made up of a sequence of steps organized into an execution diagram. Packages are the main objects used to generate scenarios for production. When you select Package as the ODI object type, then the source columns are shown, and you can specify maps in the import format in which the scenario is generated.
 - Scenario—ODI compiled code that must match exactly as specified in the ODI.
- 7 In **ODI Object Project code**, enter the template ODI project code.
- 8 In Source System Type, select the source application type.
- 9 Click Save.

Deleting a Source Adapter

Deleting a source adapter removes the parameters, drill-down URLs and source columns associated with the source adapter.

- To delete a source adapter:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Adapter**.
- 2 In the Source Adapter summary grid, select the source adapter and click Delete.

The message: "This also deletes associated Parameters, Drill Down URLs and Source Columns. Delete Source Adapter?"

3 Click OK.

Exporting a Source Adapter

Use the Export feature to migrate an adapter definition from one environment to another in the XML file format.

- To export a file:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Adapter**.
- In Source Adapter, on the Source Adapter summary task bar, select either Export Current or Export All from the Export drop-down.
 - Export Current—Exports a selected source adapter from the Source Adapter Summary list.
 - Export All—Exports all source adapters from the Source Adapter summary list.
- 3 To export a selected file, in **Specify file location**, navigate to the file to export, and then, click **OK**.

You can enter the full path name and file name in the File Name field.

4 **Optional**: From **Open File**, save the file to a local directory on the desktop.

- **Optional**: To browse for a file to export, select **Upload**, and then browse to the file to export. When the File Upload screen is displayed, navigate to the file to upload, and then, click **Open**.
- 6 Click OK.

Importing a Source Adapter

- To import a source adapter:
- 1 Select the Setup tab, and then under Register, select Source Adapter.
- 2 In Source Adapter, on the Source Adapter summary task bar, select Import.
- 3 In Specify file to import, navigate to the file, and then, click OK.
- 4 in **Open File**, open the XML file or save it to a directory of your choice.
- Optional: To browse to a file, select Upload. When the File Upload screen is displayed, navigate to the file, and click Open.
- 6 Click OK.

The source adapter is added to the summary list.

Copying a Source Adapter Key

You can copy a source adapter key and its associated detail to a new adapter key.

- To copy a source adapter key:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Adapter**.
- 2 In Source Adapter, on the Source Adapter summary task bar, select Copy.
- 3 In Copy, enter the name of the new adapter key, and then, click OK.

Using the SAP BW Source Adapter

You can use an SAP Business Warehouse (BW) source adapter to extract data from an SAP BW source system. The SAP BW integration with FDMEE differs from the standard and prepackaged integration frameworks. SAP (BW) comes with many predefined InfoCubes, or you can define your own InfoCubes.

- ➤ To add an SAP BW source adapter:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Adapter**.
- 2 In Source Adapter, on the Source Adapter summary task bar, click Add.
 - Blank entry rows are added to the Source Adapter summary section.
- 3 In the Source Adapter details grid, in Adapter Key, enter a user defined identifier for the adapter.
 - For example for an SAP BW source adapter, enter NK_SAP. The value in this field cannot be modified after it is created.

4 In *Adapter Name, enter a user defined name for the SAP BW adapter.

For example, enter NK_SAP.

5 In *ODI Package Name, enter the name of the package.

For example, enter NK_SAP.

6 In **ODI Project Code**, enter the template ODI project code name.

The ODI Object Project code identifies an object within the project.

- 7 In Source System Type select the SAP BW source system.
- 8 In **Cube Name**, enter the predefined InfoCube name.
- 9 Click Import Cube Definition.

The SAP BW cube information is imported into FDMEE, which creates the source columns in the Source Column tab.

10 In the Source Column grid, map each SAP BW cube column in the Column Name field to the type of data column in the Classification drop down.

For example, for the IOPK_0FIGL_C101_0CO_AREA_PK_CO_AREA column, select the Fact classification.

Classification types:

- Fact
- Year
- Period
- Period Number
- 11 In **Display Name**, enter the name to display for the column.

This Display Name is shown on the Import Format screen and Data Load Workbench.

12 Select the **Parameters** tab, and specify any filter parameters.

See "Defining Parameters" on page 58 for more information.

13 In the Source Adapter summary grid, click Generate Template Package.

The template package is used to generate the final scenario based on the import format mappings. If any customizations are required, you can customize the auto generated template package.

- 14 Click Save.
- 15 Define the import format and generate the process.

Defining Source Adapter Detail

When you have associated the source adapter with a source system type, the next steps include:

- Selecting Source Columns. See "Selecting Source Columns" on page 58.
- Defining Parameters. See "Defining Parameters" on page 58.

• Defining URLs for drill through. See "Defining URLs for Drill Through" on page 59.

Selecting Source Columns

Use source columns to specify the columns that are available for mapping in the import format. There is no validation of the table or column names in this section—the source columns are registered so that they are available in import format for mapping.

Adding Source Columns

The source columns added in this section must match the columns specified in the data model for the source table in the Oracle Data Integrator.

- To add a source column:
- 1 Select the Setup tab, and then under Register, select Source Adapter.
- 2 In Source Adapter, select the Source Column tab.
- 3 From the Source Column task bar, select Add.

Entry fields are displayed for the Column Name, Table Name, and Display Name columns.

4 In Column Name, enter the source column name.

For example, enter SEGMENT1. The source column name must have a corresponding column name in the ODI model.

5 In Table Name, enter the table name.

For example, enter LOGICAL_BALANCES. The table name must have a corresponding table name in the ODI model.

6 In **Display Name**, enter the display name used for the import format.

Defining Parameters

Use the Parameters tab to specify the list of parameters for a ODI Scenario. The parameter definition includes a non-translated parameter name and a translated prompt. The prompt is used for display on the data rule page.

This section includes information on:

- Adding Parameters. See "Adding Parameters" on page 58.
- Deleting Parameters. See "Deleting Parameters" on page 59.

Adding Parameters

When you add a parameter for the source adapter, a definition is created, which includes a non-translated parameter name and a translated prompt.

- To add a source column:
- 1 Select the Setup tab, and then under Register, select Source Adapter.

- 2 In Source Adapter, select the Parameters tab.
- 3 From the Parameters task bar, select Add.

Entry fields are displayed for the Parameter Name, Parameter Data Type, and Parameter Prompt columns.

4 In Parameter Name, enter the parameter name.

For example, enter p_actual_flag. This is the name of the parameter in the ODI project.

- 5 In Parameter Data Type, select the data type of the parameter:
 - Char
 - Number
 - Date
- 6 In Parameter Prompt, enter the parameter prompt.

For example, enter Balance Type in this row.

Note: When a new parameter is added and you are in a multilanguage environment, then set the browser locale to the appropriate language and edit the prompts as needed in that language.

Deleting Parameters

- To delete a parameter:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Adapter**.
- 2 In Source Adapter, select the Parameters tab.
- 3 On the **Parameters** task bar, select the parameter, and then, click **Delete**.
- 4 Click OK.

Defining URLs for Drill Through

FDMEE provides a framework for using URLs for drill through. You click the hyperlink in the amount cell, and the source system is launched in a new EPM Workspace tab or a new window. Multiple drill through URLs are provided to drill to different pages in the source system based on the import format mapping.

Note: For more information about the FDMEE drill-through feature, see "Drilling Through" on page 22.

Adding Drill Through URLs

To create a drill through URL, specify the URL name, URL, request method, and URL prompt.

- To add a drill through URL:
- 1 Select the Setup tab, and then under Register, select Source Adapter.
- 2 In Source Adapter, select the Drill URL tab.
- 3 On the Drill URL task bar, select Add.

Entry fields are displayed for the Drill URL Name, Drill URL, Request Method, and Drill URL Prompt columns.

- 4 In **Drill URL Name**, enter a user defined name for the drill through URL.
- 5 In Drill URL, enter the URL used for the drill through.

Enter the URL without the server and port information. The URL must contain the parameter name and column name from the TDATASEG table enclosed in the symbol \$.

For example, enter: LEDGER_ID=\$ATTR1\$&GL_PERIOD=\$ATTR2\$.

In the above example the value of ATTR1 is passed as a value for the LEDGER_ID parameter, and ATTR2 is passed as the value for the GL_PERIOD parameter. Parameters are separated by the "%" character.

In some cases, the you may want to refer to the column name in the FDMEE staging table based on the source mapping that is defined in the import format. When this is the case, the user specifies the source column name as the parameter for the URL, and this tells the system to translate the source column name to the related column in TDATASEG when the URL is parsed. To specify a source column from the import format as the value for a parameter in the URL, enclose the value with \$\$ on each side

For example, enter: COMPANY CODE=\$\$RBUKRS\$\$

In the above example, the value for the COMPANY_CODE parameter is determined using the import format. If RBUKRS is mapped to the ENTITY dimension, then the value of the ENTITY column is passed as the parameter.

To specify the request-response between a client and server for the drill URL format, type either:

- GET—Form data is encoded into the URL. For example, specify GET@http://www.oracle.com/.. If no method is specified, then GET is the assumed request-response.
- POST—Form data is displayed in the message body. For example, specify POST@http://www.oracle.com/.
- 6 In the **Drill URL Prompt**, enter a user-defined prompt for the drill-through prompt.

For example, enter Default.

Drill Through using JavaScript

You can specify a JavaScript based drill through that allows you to drill through to a graphical user interface instead of an URL. To do this, you write a script in the Script Editor and store it in a custom script directory. The JavaScript for the drill through is associated with the source adapter.

The JavaScript for the drill through is associated with the source adapter. You invoke JavaScript from the drill through link by specifying "#javascript" instead of "http URL" in the source system. You can also pass contextual parameters to a JavaScript function similar to the http drill URL.

- To add a drill through using:
- 1 Select the **Setup** tab, and then under **Register**, select **Source System**.

For more information about source systems, see "Setting up Source Systems" on page 48.

- 2 Select the source system file for the JavaScript based drill through.
- 3 In the detail section, from Drill Through URL, enter #javascript.
- 4 Click Save.
- 5 From the **Setup** tab, and then **Scripts**, select **Script Editor**.
- 6 From Script Editor, click New.
- 7 From Script Type, select Drill.

Note: The Technology drop down is Javascript because drill through scripts can only be created based on JavaScript.

- 8 In **File Name**, enter a name for the script.
- 9 Click OK.
- 10 From the Target Application drop down, select the target application for this drill through script.
- 11 Write the code for the custom in the Script Editor.

Make sure the name of the Javascript function is same as the file name.

For example,

```
function main($ATTR1$,$ATTR2$){
var ledger_id = $ATTR2$;
var currency = $ATTR1$;
var resp_id = 300;
if(ledger_id==1)
 resp_id = 100
else if(ledger_id ==2)
 resp_id = 200
alert('Currency:' + $ATTR1$);
alert('ledger_id:' + ledger_id);
var url = 'http://machinename.us.oracle.com:6362/OA_HTML/RF.jsp?
function_id=1045073&CALLING_PAGE=FDM_DRILLDOWN&resp_appl_id=&resp_id=&SOB_ID=1&LED_I
D='+ledger-id+'&fdm_per=Jan-03&fdmccid=13098&fdm_currency+'+currency
+'fdm_balance_type=A
alert('url:' + url);
window.open(url);
}
```



- 12 Click Save.
- 13 Select the Setup tab, and then under Register, select Source Adapter.

For more information about source adapters, see "Working with Source Adapters" on page 52.

- 14 In Source Adapter, select the Drill URL tab.
- 15 On the Drill URL task bar, select Add.

Entry fields are displayed for the Drill URL Name, Drill URL, and Drill URL Prompt columns.



- 16 In **Drill URL Name**, enter a user defined name for the drill through URL.
- 17 In the **Drill URL**, enter the Function name of drill through

The Function name should be same as file name.

For example, you could enter main(\$ATTR1\$,\$ATTR2\$).

- 18 In Drill URL Prompt, enter a user-defined prompt for the drill-through prompt.
- 19 Click Save.

When you drill down from the workbench, a new window is launched with the specified URL.

Deleting Drill URLs

- To delete a drill through URL:
- 1 Select the Setup tab, and then under Register, select Source Adapter.
- 2 In Source Adapter, select the Drill URL tab.

- 3 On the Drill URL task bar, select the drill-through URL, and then, click Delete.
- 4 In Delete Source Adapter Drill URL, click OK.

Registering Target Applications

Subtopics

- Creating a Custom Target Application
- Adding Lookup Dimensions
- Defining Application Dimension Details
- Defining Application Options for Essbase and Planning
- Defining Application Options for Public Sector Planning and Budgeting
- Defining Application Options for Financial Management
- Deleting Registered Target Applications

You must register target applications for use with FDMEE. When you register target applications, you select the target application that requires metadata and/or data from one or more source systems.

- ➤ To register target applications:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In Target Application, in the Target Application summary grid, click Add.
- 3 In Select Application, from Application Type, select the application.

Valid application types:

- Financial Management
- Planning
- Essbase
- Account Reconciliation Manager
- Applications Registered in FDM
- Custom Applications
- FDM Target Applications only: From FDM Application, select the FDM application, and then from Name, select the application name, and then click OK.
- 5 In Application Details, enter the application name.
- 6 Click OK.
- 7 If using an Essbase database, select the **Essbase Database Name**.
- 8 In Target Application, click Save.
- 9 Define the dimension details.

See "Defining Application Dimension Details" on page 67.

Note: Do not modify dimensions in the EPMA application after you have registered the target application in FDMEE. Any action such as sharing or removing the dimension, and then re-adding it, alters the dimensions ID and creates a mismatch between EPMA and FDMEE.

10 Select the application options.

For Essbase or Planning applications, see "Defining Application Options for Essbase and Planning" on page 68.

For Public Sector Planning and Budgeting, see "Defining Application Options for Public Sector Planning and Budgeting" on page 73.

For Financial Management, see "Defining Application Options for Financial Management" on page 74.

Note: No application options are available for the Account Reconciliation Manager.

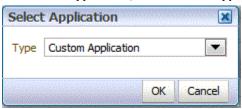
Creating a Custom Target Application

A custom target application allows you to load data from a supported source, and then extract the data into a flat file instead of loading it to Essbase, Planning, and Financial Management. You can define the custom target application with required dimensionality. Instead of exporting the data to a target application, FDMEE generates a data file that can be loaded to an external system using a custom process.

When creating a custom target application, note the following:

- Data is written to the file in the following predefined order: Account, Entity, UD1, UD2,..UD20, AMOUNT.
- The sequence that is specified when you create the custom app definition is used to order the dimensions for mapping processing. Note the sequence in case you have scripts that have an order of processing dependency.
- Metadata Rule—Metadata load is not applicable for custom applications. The Location associated with a custom application is filtered in the metadata load POV.
- Data Rule—The POV category is not validated. Plan types are not available.
- Data Load Mapping—Target values are not validated for custom applications.
- Data Load Execution—When the option to export the data file is enabled, FDMEE creates
 an output data file using the AIF_HS_BALANCES view. The name of the data file is
 <Target App Name>_<Process ID>.dat, and it is written to the <APPL ROOT
 FOLDER>/outbox directory. In addition the OUTPUT_FILE column in the
 AIF_PROCESSES table is updated. The data file can be accessed from the Process Details
 page from the OUTPUT file column.
- Check Rules can be created from a custom target applications. Check rules cannot be based on target values.
- Write-back is not applicable for custom target applications.
- An export mode is not supported for custom target applications.

- To define a custom target application:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In Target Application, in the Target Application summary grid, click Add.
- 3 From Select Application, select Custom Application and click OK.



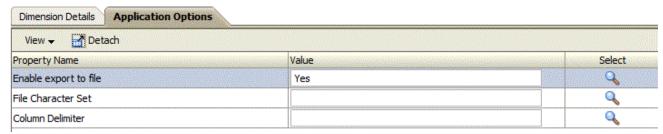
- 4 In Application Details, enter the application name.
- 5 Select the **Dimension Details** tab.
- 6 Specify the Dimension Name.
- 7 Select the Target Dimension Class or click to select the Target Dimension Class for each dimension that is not defined in the application.

The dimension class is a property that is defined by the dimension type. For example, if you have a Period dimension, the dimension class is also "Period." For Essbase applications, you must specify the appropriate dimension class for Account, Scenario, and Period. For Public Sector Planning and Budgeting applications, you must specify the dimension class for Employee, Position, Job Code, Budget Item, and Element.

- In **Data Table Column Name**, specify the table column name of the column in the staging table (TDATASEG) where the dimension value is stored.
 - Click to search and select a data table column name.
- 9 In **Sequence**, specify the order in which the maps are processed.

For example when Account is set to 1, Product is set to 2, and Entity is set to 3, then FDMEE first processes the mapping for Account dimension, followed by Product, and then by Entity.

10 Click Application Options.



In **Enable export to file**, select **Yes** to have FDMEE create an output data file for the custom target application.

A file I created in the outbox folder on the server with the following name format: <LOCATION>_<SEQEUENCE>.dat. For example when the location is named "Texas" and the next sequence is 16, then the file name is Texas_15.dat. The file is created during the export step of the workflow process."

When the **Enable export to file** option is set to **No**, then the Export to Target option is not available in execution window.

12 In File Character Set, select the file character set.

The file character set determines the method for mapping bit combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41.

13 In Column Delimiter, select the character to use for delimiting columns in the output file.

Available column delimiters are:

- NA
- •
- •
- •
- :
- •
- 14 Click Save.

Adding Lookup Dimensions

Lookup dimensions can be created and assigned with data columns for target applications. Lookup dimensions are used for mapping and reference. They can only be used in FDMEE and do not affect the dimensionality of a target application. They can also be used with member mapping functionality to cross-reference multiple source segments and chartfields and assign a target value.

- To add a lookup dimension:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In the Target Application summary grid, select a target application.
- 3 Select the Dimension Details tab.
- 4 Click Add.

Blank dimension name and data table column name entry fields are displayed.

- 5 In **Dimension Name**, enter the lookup dimension name.
- 6 In Data Table Column Name, select the data column from which to base the lookup dimension.
- 7 Click OK.

The lookup dimension is added to the dimension detail list with the target dimension class name of "LOOKUP."

Defining Application Dimension Details

The dimension details differ for each application type. For Public Sector Planning and Budgeting and Essbase applications, you reclassify the dimensions and change the dimension class as necessary.

- To define dimension details:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In the Target Application summary grid, select an Essbase or Planning application.
- 3 Select the **Dimension Details** tab.
- 4 Select the **Target Dimension Class** or click to select the **Target Dimension Class** for each dimension that is not defined in the application.
 - The dimension class is a property that is defined by the dimension type. For example, if you have a Period dimension, the dimension class is also "Period."
 - For Essbase applications, you must specify the appropriate dimension class for Account, Scenario, and Period. For Public Sector Planning and Budgeting applications, you must specify the dimension class for Employee, Position, Job Code, Budget Item, and Element.
- 5 Optional: Click Refresh Metadata to synchronize the application metadata from the target application.
- 6 In **Data Table Column Name**, specify the table column name of the column in the staging table (TDATASEG) where the dimension value is stored.
 - Click \(\frac{1}{2} \) to search and select a data table column name.
- 7 In **Sequence**, specify the order in which the maps are processed.
 - For example when Account is set to 1, Product is set to 2, and Entity is set to 3, then FDMEE first processes the mapping for Account dimension, followed by Product, and then by Entity.
- For Essbase applications and Oracle Hyperion Profitability and Cost Management only: Select the Create Drill Region.

The selection of the dimensions to create a drill-region on really defines the granularity for the drillable region. For example, if you only select the year dimension, you can drill on a cell that at least does not have a null for year.

9 Click Save.

The target application is ready for use with FDMEE.

Tip: To edit the dimension details, select the target application, then edit the application or dimension details as necessary. To filter applications on the Target Application page, ensure that the filter row is displaying above the column headers. (Use the to toggle the filter row.) Then, enter the text to filter.

Defining Application Options for Essbase and Planning

After defining the application details and dimension details, for Essbase , define the application options.

Note: The user attempting to load data to Planning must be provisioned with Essbase administrator or Planning administrator rights.

- To define application options for Essbase applications:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- In the Target Application summary grid, select a Essbase target application.
- 3 After defining the application details and dimension details in Application Detail, select the Application Options tab.
- Complete the application options as needed.
 Essbase application options are described in Table 6 on page 68.
- 5 Click Save.

Table 6 Essbase Application Options and Descriptions

Option	Description
Purge Data File	When a file based data load to Essbase is successful, specify whether to delete the data file from the application outbox directory. Select Yes to delete the file, or No to retain the file.
Prefix Dimension for Duplicate Members	When set to Yes member names are prefixed by the Dimension Name. The member name that is loaded is in the format [Dimension Name]@[Dimension Member]. The prefixed dimension name is applied to all dimensions in the application when this option is enabled. You cannot select this option if there is a dimension in the target that has duplicate members. That is, only select this option when the duplicate members cross dimensions.
	If the application supports Duplicate Members and Prefix is set to No , then the user is responsible to specify the fully qualified member names. Refer to Essbase Documentation for fully qualified member name format
	Note: Planning does not support duplicate members.
Global User for	Option to override the Single Sign-On logon to the Essbase and the Planning applications.
Application Access	When a user name is specified for this option, this user name is used to access Essbase/Planning applications instead of the FDMEE sign-on user name.
	Specify the user name of a user who has administrator access to the Planning application, and/or Application/Database Manager access to the Essbase applications.
Batch Size	Specify the batch size used to write data to file. The default size is 10,000.

Option	Description
Load Rule Name	Specify the rule name for loading data from the TDATASEG staging table to Essbase.
	Available methods:
	File—Data is written to a data file in the OUTBOX directory of the application (defined in System Settings). The file name is in the format APPLICATION NAME <a href="mailto:APPLICATION NAME APPLICATION NAME <a a="" href="mailto:APPLICATION NAME <a href=" mailto:application="" name<=""> <a href="mailto:APPLICATION NAME <a href=" mailto:applicati<="" td="">
	The file load method creates an application file type of ESSFILE.
	SQL—Uses SQL Method to load data. The SQL load method is the default.
	The SQL method creates an application file type of ESSSQL.
Check Entity	Specify the calculation method for check entities.
Calculation Method	Available methods are:
	dynamic—Check entity data is calculated based on the data at retrieval time.
	calculation script—Check entity data is calculated based on a predefined calculation script.
Source Language	Select the default language for the member descriptions.
for Member Description	To understand how languages are processed, see "How Languages are Processed" on page 115.
Drill Region	Select Yes, to create a drill region. A drillable region is created to use the drill through feature for Essbase, or Planning data.
	Note: FDMEE does not support drilling through to human resource data.
	The drill region URL allows Essbase, Oracle Smart View for Office, and Oracle Hyperion Financial Reporting to drill to the proper landing page.
	When loading data from FDMEE, the drill region is loaded to Planning data.
	A drill region includes the Entity/Account/Scenario/Year/Period for Planning, a URL to get back to FDMEE, and a region name. For Essbase, you select the dimension to use for the drill region.
	Select Yes to enable or No to disable.

Using Calculation Scripts

When loading data to Essbase or Planning, FDMEE enables you to run custom calculation scripts. A calculation script is a series of calculation commands, equations, and formulas that allow you to define calculations other than those defined by the database outline. Calculation scripts are defined in Essbase and Planning.

Custom calculation scripts are supported in the target application and the check entity group. In the target application:

- Scripts can be parameterized so that you can specify a parameter for each script.
- Scripts can be assigned to run before or after the data load.
- Scripts are assigned a scope of applicability. The scope may be at the Application, Category, Location, and Data Rule level. If scripts are assigned at the lowest level, then they take precedence over script at higher level. Data Rule is the lowest level and Application is the highest level. Only scripts at one level are executed.

- To add a calculation script:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 From the Calculation Script tab, select a calculation script.
- 3 Click display the Parameter window.
- 4 For Planning applications only, select the **Plan Type** to retrieve the parameters defined in the calculation script.
- 5 Click the Add to add a script parameter.
- 6 Select the parameter that has been defined for the script.

You can also click **Browse**, select the parameter, and click **OK**. You can also manually type in the parameter name.

- 7 In Script Value select a predefined parameter value, or select Custom to specify your own script value.
- 8 Optional: If you selected Custom in the Script Value field, enter the value in Custom.

For example, enter **0001**.

- 9 Click OK.
- 10 In Script Name, specify the name of the script.
- 11 Click to add the parameters for the calculation script.

For more information, see "Adding Calculation Script Parameters" on page 71.

12 In Script Scope, select the scope of applicability,

Note that scripts assigned at the lowest level take precedence over scripts at higher level.

Available scopes in order of highest scope to lowest are:

- Application (default scope)
- Category
- Location
- Data Rule
- 13 In **Scope Entity**, select the specific value associated with the script scope.

For example, if the script scope is "Location," select the location associated with the application.

The Application script scope is disabled for the scope entity.

14 In **Event**, select the event that executes the calculation script.

Available events:

- Before Data Load
- After Data Load
- Before Check
- After Check

15 In **Sequence**, specify the order in which the script is executed.

Since multiple scripts can be executed for a event, this sequence values provides the numeric order in which each script is executed. You can enter any number, but the number has to be unique within an Event.

16 Click Save,

Adding Calculation Script Parameters

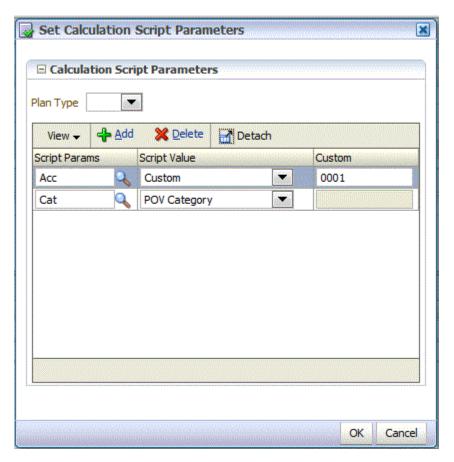
You parameterize calculation scripts by identifying and defining the list of values in the scripts. Parameter values can be predefined for a list of values, for example, the POV Period, POV Location, POV Category, Check Group Entity, and all data load entities.

You can also specify custom parameters, in which case, you can enter any Essbase filter syntax.

Additionally, you can reference values stored in the Integration Option 1-4 fields in the Location table to drive calculation logic. Location Option 1, Location Option 2, Location Option 3, and Location Option 4 values can be referenced. In this case, the exact string that is passed as a parameter has to be stored in the Integration Option field of the Location table.

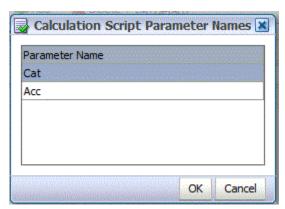
Similarly, you can reference four option fields in the Data Rule for parameters. These fields are Data Rule Option 1, Data Rule Option 2, Data Rule Option 3, and Data Rule Option 4. The column added to the Data Rule is called RULE_OPTION1, RULE_OPTION2, RULE_OPTION3 and RULE_OPTION4.

- ➤ To add calculation script parameters:
- 1 From the Calculation Script tab, add or select a calculation script.
- 2 Click to add the parameters for the calculation script.
- 3 For **Planning applications only**: select the **Plan Type** for Accounting and Entity dimensions.



4 In Script Parameters, select the parameter that has been defined for the script.

You can click the **Browse** icon, select the parameter, and click **OK**.



- 5 In Script Value, select the value to which to apply the calculation script and click OK.
- To add a custom script parameter:
- 1 From the Calculation Script tab, add or select a calculation script.
- 2 Click to add the parameters for the calculation script.
- 3 In Script Parameters, select the parameter that has been defined for the script.

You can click the **Browse** icon, select the parameter, and click **OK**.

- 4 In Script Value, select Custom and click OI.
- 5 In **Custom**, enter the filter syntax for the custom value.

For example, enter **0001**.

Defining Application Options for Public Sector Planning and Budgeting

After defining the application details and dimension details, for Public Sector Planning and Budgeting applications, you define the application options. In FDMEE, you must specify a parent member to populate all dimensions applicable for your model. You can optionally enter a parent member for earnings and benefits.

You should enter the value as necessary, for the following options:

- To define application options for Public Sector Planning and Budgeting Applications
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In the Target Application summary grid, select a Public Sector Planning and Budgeting application.
- 3 After defining the application details and dimension details in Application Detail, select the Application Options tab.
- 4 Complete the application options as needed.

Public Sector Planning and Budgeting application options are described in Table 7 on page 73.

5 Click Save.

The target Public Sector Planning and Budgeting application is ready for use with FDMEE.

 Table 7
 Public Sector Planning and Budgeting Application Options and Descriptions

Option	Description	
Budget Model Type	Enter the Public Sector Planning and Budgeting model. (For Employee only, enter EMPLOYEE. For Position only, enter POSITION, or for both Employee and Position, enter Total Existing Positions).	
Parent Member for all Positions	Enter the parent member for all positions imported from your human resource source system. For example, enter Total Existing Positions.	
Parent Member for all Entities	Enter the parent member for all departments imported from your human resource source system. For example, enter Total Entity.	
Parent Member for all Employees	Enter the parent member for all employees imported from your human resource source system. For example, enter Existing Employees.	
Parent Member for Salary Elements	Enter the parent member for salary grades imported from your human resource source system. For example, enter Salary Grades.	

Option	Description			
Parent Member for Earnings Elements	Enter the parent member for earnings codes imported from your human resource source system. For example, enter Additional Earnings.			
	Specifying parent members for earnings and benefit elements is optional. When you do not specify a parent member, the corresponding elements are not loaded into the application when you run the human resource data load rule.			
Parent Member for all Job	Enter the parent member for all job codes imported from your human resource source system. (The parent member for job is applicable only to Employee Only models.)			
Parent Member for Benefit Elements	Enter the parent member for benefit codes imported from your human resource source system. For example, enter Benefit Defaults.			
Source	Select the default language for the member descriptions.			
Language for Member Description	To understand how languages are processed, see "How Languages are Processed" on page 115.			
Drill Region	Select Yes, to create a drill region. A drillable region is created to use the drill through feature for Essbase, Financial Management, or Planning data.			
	Note: FDMEE does not support drilling through to human resource data.			
	The drill region URL allows Essbase, Financial Management, Planning, Oracle Hyperion Smart View for Office, Fusion Edition, and Oracle Hyperion Financial Reporting, Fusion Edition to drill to the proper landing page.			
	When loading data from FDMEE, the drill region is loaded to either Financial Management or Planning data.			
	A drill region includes Account/Scenario/Year/Period forFinancial Management or Entity/Account/Scenario/Year/Period for Planning, a URL to get back to FDMEE, and a region name. For Essbase, you select the dimension to use for the drill region.			
	Select Yes to enable or No to disable.			

Defining Application Options for Financial Management

After defining the application details and dimension details, for Financial Management application, you define the application options.

- ➤ To define Financial Management application options:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In the **Target Application** summary grid, select a Financial Management target application.
- 3 After defining the application details in Application Detail, select the Application Options tab.
- 4 Complete the application options as needed.
 - Financial Management application options are described in Table 8 on page 75.
- 5 Click Save.

 Table 8
 Financial Management Application Options and Descriptions

Option	Description				
Check Intersection	Enables the checking of Financial Management data intersections (account, entity, and so on) during the Validation step in the Data Load Workbench. The default setting is Yes .				
Load Line Item	Enables loading of line-item detail to Financial Management.				
Detail	Select Yes to enable or No to disable.				
Line Item Detail Load Type	Specify whether line item detail or summary data is loaded for cells. The default is Load Detail , which displays details for the selected cell. (Specifies whether an account can have line items.) When this option is Load Summarized , cells show summarized information.				
	Load Summarized				
	Load Detail				
Enable Data	Enables the data load process.				
Load	Select Yes to enable or No to disable.				
Load Process	Select the process for loading data. Select Scan to scan the file for invalid records before loading it to the application. Select Load to load the file only.				
	Scan— Validates data and lists invalid records into the Log. When this option is selected, data is not loaded to the target application.				
	Load—Validates and loads data to the target application.				
Load Method	Select the method for loading a data file into an application. Available load methods:				
	• Replace —Replaces the data in the application with the data in the load file. For each unique combination of Scenario, Year, Period, Entity, and Value in the data file, the Replace option clears all account values from the application, then loads the value from the data file.				
	Note: You may need to create several small files to load a data file using the Replace mode, especially when the data is very large or when the file contains ownership data. An error message is displayed When the file is too large when you try to load it.				
	• Merge—Overwrites the data in the application with the data in the load file. For each unique point of view that exists in the data file and in the application, the value in the data file overwrites the data in the application.				
	Data in the application that is not changed by the data load file remains in the application.				
	When you select the Accumulate In File option in conjunction with the Merge option, the system adds all values for the same point of view in the data file, and overwrites the data in the application with the total.				
	For each unique point of view that is in the data file but does not have a value in the application, the value from the data file is loaded into the application.				
	Accumulate—Select the Accumulate option to accumulate the data in the application with the data in the load file. For each unique point of view in the data file, the value from the load file is added to the value in the application.				
	Note: Data for system accounts is not accumulated.				
	 Replace by Security—Performs a data load in Replace mode in which only the members to which you have access are loaded. This option enables you to perform a data load in Replace mode even when you do not have access to all accounts. When you perform the Clear operation for a period in a sub-cube, only the cells to which you have access are cleared. Data, cell text, and line item detail are cleared, but cell attachments are not cleared. 				

Option	Description			
Accumulate in File	You use the Accumulate in File option in conjunction with the Merge and Replace options. When a data load file contain multiple lines of data for the same point of view, this option first accumulates the data in the file and then loads the totals into the application based on the selected load option. For each unique point of view in the data file, the value from the load file is added to the value in the application. For example, when you have 10,20 and 30 in the file, 60 loaded. Select Yes to enable or No to disable.			
Has Ownership	When the file that you are loading contains ownership data, you must indicate this option. If you do not select this option and the data file contains ownership or shares data, an error occurs when you load the file.			
	Select Yes to enable or No to disable.			
Enable Data Protection	Enables FDMEE to protect target-system data from being overwritten during data loads; is based on a specified protection value. Use this option when data is entered into the target system through a method other than FDMEE.			
	Select Yes to enable or No to disable.			
Protection Value	The value that is to be protected during the Load step when Enable Data Protection is enabled. The value in this field is to be a dimension value inside of Financial Management across any dimension.			
	For example, enter Protect This .			
Protection Operator	Select operator (= or <>). This is used only when the Enable Data Protection option is enabled. The option allows you to state that the data to be protected is equal (=) or not equal (<>) to the "Protection Value".			
Enable Journal	Enables the loading of journal files.			
Load	You can load working, rejected, submitted, approved, and posted journals as well as standard and recurring journal templates. You cannot load automated consolidation journals because they are created by the consolidation process.			
	The default setting for this option is No .			
	This setting is also used with the Data Value selected for the location on the Location screen to determine when and how data is loaded to Financial Management as journals.			
	Select Yes to enable or No to disable.			
Drill Region	Select Yes to create a drill region.			
	Drillable region definitions are used to define the data that is loaded from a general ledger source system and specify the data drillable to FDMEE.			
	In data grids and data forms, after the regions have been loaded, cells that are drillable are indicated by a light blue icon at the top left corner of the cell. The cell context menu displays the defined display name, which then opens the specified URL.			
	A region definition load file consists of the following information:			
	Scenario, Year, Period, Entity, Account			
	Display Name (for cell context menu) and URL (to drill to)			
Enable Cell Text Loading	Enables the loading of text and documents to a data cell. FDMEE archives documents in EPM_ORACLE_HOME/products/FinancialDataQuality/data directory.			
	Select Yes to enable or No to disable.			

Option	Description
Enable	Enables consolidation in the data load.
Consolidation	Consolidation is the process of gathering data from dependent entities and aggregating the data to parent entities. Launching consolidation runs the consolidation process for the specified scenario, year, period, entity, and value. As a part of that process, consolidation for all descendant entities and all prior time periods within the same year is run, when it has not been run previously. The consolidation process runs all calculation rules functions for each affected entity and value, and the translation process runs as necessary to convert from child entities to parent entities. Select Yes to enable or No to disable.
Consolidation Type	Select the consolidation type for the data load.
71.	Available consolidation types:
	• Impacted—The Consolidate (Impacted Consolidation) option is available for any parent entity with a calculation status of CN or CN ND. When you select this option for a parent entity, the system performs calculations (Calculate, Translate, Consolidate) for any dependent entity within the consolidation path of the selected parent that has a calculation status of CN, CN ND, CH, TR, or TR ND, on the current period or on any prior period in the current year. Consolidate is the most efficient option because only entities that require consolidation are updated.
	Process units with a status of NODATA on the current period and all prior periods are skipped. Process units with a status of OK or OK SC on the current period are not recalculated, retranslated, or reconsolidated. If the selected parent has a status of CN or CN ND in the prior period, consolidation runs for all periods from the first period in the current year where the parent is impacted until the current period.
	• All with Data—The Consolidate All with Data option is available for any parent entity, regardless of its status. When you select this option for a parent entity, the system consolidates every dependent entity within the consolidation path of the selected parent that contains data, regardless of its status, in the current period or in any of the prior periods. The system also runs calculation rules for the dependent entities. It does not perform a consolidation on entities that contain zeroes or no data. This option is useful for updating system status from OK SC to OK after metadata changes.
	Process units with a status of NODATA on the current period and all prior periods are skipped. Consolidation Options units with a status of OK or OK SC on the current period are recalculated, retranslated, and reconsolidated. If the selected parent has a status of CN or CN ND in the prior period, consolidation runs for all periods from the first period in the current year where the parent is impacted until the current period.
	• All—The Consolidate All option is available for any parent entity, regardless of its status. When you select this option for a parent entity, the system performs calculations for every process unit within the consolidation path of the selected parent, regardless of its status. It consolidates all entities whether they contain data or not. This option is useful when an update from prior periods is required, or when an entity with no data needs to be populated using allocations. This option should be used sparingly because the system does not omit entities with no data, which can have a significant impact on consolidation performance.
	Process units with a status of NODATA on the current period are calculated, translated, and consolidated. Process units with a status of OK or OK SC on the current period are recalculated, translated, and reconsolidated. If the selected parent has a status of CN or CN ND in the prior period, consolidation runs for all periods from the first period in the current year where the parent is impacted until the current period.
	 Entity—Calculates the contribution of each entity to its parent, and then totals all contributions to arrive at the consolidated numbers.
	 Force Entity Only—Forces calculation to run for all selected contribution values to arrive at the consolidation numbers.
Enable Force	Enables the execution of the default calculation call prior to a consolidation run.
Calc	Select Yes to enable or No to disable.
Enable Force	Enable to force translation to run for all selected cells.
Translate	Select Yes to enable or No to disable.

Option	Description		
Translation Level	Determines the translation level (levels to include for translation) of rows and columns when loading data. Available levels:		
	<entity adjs="" curr=""></entity>		
	<entity curr="" total=""></entity>		
	<entity currency=""></entity>		
	<parent adjs="" curr=""></parent>		
	<parent curr="" total=""></parent>		
	<parent currency=""> (Default)</parent>		
Enable Multi- Load Zero Loading	Select Yes to load 0 values during a multiple period load.		
Journal Status	To load journals with a specific status, select the status. The journal status indicates the current state of the journal.		
	Available journal statuses:		
	1—Working: Journal is created. It has been saved, but it can be incomplete. For example, a label or single entity may need to be assigned.		
	2—Submitted: Journal is submitted for approval.		
	3—Approved: Journal is approved for posting.		
	4—Posted: Journal adjustments are posted to the database (default).		
	5—Journal is rejected or unposted.		
Journal Default	Specify the default value of the journal. The default setting for this option is <entity adjs="" curr="">.</entity>		
Value	Available values are:		
	• [Contribution Adjs]		
	[Parent Adjs]		
	<entity adjs="" curr=""></entity>		
	<parent adjs="" curr=""></parent>		
Journal Enable	Assign a journal id (journal numbering) to entities that are being loaded.		
JV ID per Entity	Select Yes to assign one journal id for each entity in the POV. Select No , to assign one id for all data in the POV. This option is only used when loading journals.		
	The default setting is Yes .		
	This option only applies to FDMEE data that is imported as data and not through the Journal interface. Because regular data that is imported into FDMEE can be loaded to Financial Management as a journal and Financial Management requires all journal to have a JV ID, this option enables FDMEE to determine how the JV ID's are created.		
Journal Balance	Select the journal balancing attribute used in the journal subsection.		
Attribute	Available attributes are:		
	B—Balanced		
	U—Unbalanced		
	E—Balanced By Entity		
	This option is only used when loading journals to Financial Management. The default setting is U-Unbalanced.		

Option	Description	
Source Language for Member Description	Indicates the source language for the description when members are loaded. Available languages are shown on the drop-down.	
Global User for Application Access	Option to override the Single Sign-On logon to a Financial Management application. When a user name is specified for this option, this user name is used to access Financial Management applications instead of the FDMEE sign-on user name. Specify the user name of a user who has administrator access to Financial Management applications.	

Enabling Intercompany Partner Dimensions for Financial Management

FDMEE supports intercompany partner (ICP) dimensions used by Financial Management. The ICP dimension represents a container for all intercompany balances that exist for an account. ICP is a reserved dimension used in combination with the account dimension and custom dimensions to track and eliminate intercompany transaction details.

When working with ICP transaction in FDMEE, note the following:

- ICP transactions can be loaded only from a file.
- If the import format contains a dimension of type "ICPTRANS", then it is considered an ICP Transaction load.
- If the import format does not contain any dimension of type "ICPTRANS" then it is considered a standard data load.
- When ICP transactions are enabled, the following attributes are included:
 - o Attr1 Transaction ID
 - o Attr2 Transaction Sub ID
 - Attr3 Reference ID
 - Attr4 Transaction Date
 - Attr5 Entity Currency Amount
 - o Attr6 Transaction Currency
 - Attr7 Conversion Rate
 - o Attrll Comment

ICP transactions are included in the import format.

When registering a Financial Management application, an "Enable ICP Transactions" option is available. When this option is enabled, additional ICP dimensions are added to Dimension Details. These dimensions ware registered with dimension classification "ICPTRANS" (ICP Transactions) and the attributes of the values are stored in the data table column name "ATTR". Source values of the ICP dimensions cannot be transformed by mapping. If some transformation is required, then you use a script from another dimension map. Also see "Entity and Intercompany" on page 112.

Below is the list of dimensions to be added:

- To enable intercompany partner dimensions:
- 1 Select the Setup tab, and then under Register, select Target Application.
- 2 In the Target Application summary grid, select an Financial Management application
- 3 Select the Dimension Details tab.
- 4 Click Enable ICP Transactions.

Deleting Registered Target Applications

Use caution when deleting registered target applications. When you delete a registered target application, the target application is removed from the Target Application screen, and all metadata and data rules associated with the application are removed.

Use caution when deleting registered target applications. When you delete a registered target application, the target application is removed from the Target Application screen.

- To delete a registered target application:
- 1 Select the **Setup** tab, and then under **Register**, select **Target Application**.
- 2 In Target Application, select the target application and then click Delete.
- 3 Click OK.

When you delete a target application, the application is marked for deletion and is not available for any metadata or data rule processes, including the import format and location creation options. All the existing rules involving the target application are removed.

Note: After a target application is deleted and the process has run successfully, use the Target Application screen to set up the same application and redefine the rules.

4 Click Save.

Selecting Source Accounting Entities

Select accounting entities to specify the Fusion Financials or ledgers or PeopleSoft business units from which the metadata and/or data is extracted.

After you register and initialize your source systems for use with FDMEE, select the source system accounting entities (ledgers or business units) to use. Segments and chartfields are the terminologies specific to Fusion, E-Business Suite, and PeopleSoft.

- To select source accounting entities in a source system:
- 1 Select the Setup tab, and then under Register, select Source Accounting Entity.
- 2 In Source Accounting Entities, select the Entities tab.
- 3 Select the source system type.

4 Select a source system.

The source accounting entities are displayed in a grid. You can click the column header to sort any column. The Entities tab displays the following columns:

For Fusion and E-Business Suite source systems:

- Select—A check mark indicates that the accounting entity (ledger) is available.
- Accounting Entity—Name of the ledger
- Chart of Accounts—Chart of accounts name
- Currency—The functional currency of the Fusion and E-Business Suite ledger
- Calendar—The Fusion and E-Business Suite ledger calendar. The Fusion and E-Business Suite ledger is a collection of chart of accounts, currency, and calendar. For example, 4–4–5, Weekly, Accounting, and so on.
- Responsibility Name—Displays the general ledger drill-through responsibility. The drill through responsibility must be set in FDMEE to enable users to drill through to E-Business Suite. The responsibility selected must have the authority to view summary journals and journal details for the selected ledger in the E-Business Suite.

For PeopleSoft source systems:

- Select—Select the check box to make the business unit available.
- Business Unit—Business unit name
- Currency—The base currency for the business unit
- For Fusion and E-Business Suite source systems, select the general ledger Responsibility Name. See "Assigning General Ledger Responsibility" on page 81.
- 6 For each ledger or business unit that you want to make available in FDMEE, select the check box.
- 7 Click Save.
- Tip: To filter by the business unit or accounting entity, ensure that the filter row is displayed above the column headers. (Click the to toggle the filter row.) Then, enter the text to filter.

Assigning General Ledger Responsibility

In the E-Business Suite General Ledger, the system administrator assigns users general ledger responsibility. General ledger responsibility provides the authentication required for FDMEE to drill through to the Fusion and E-Business Suite journal summary page.

- To assign general ledger responsibility:
- In Source Accounting Entities, in the Responsibility Name column, click select a General Ledger Responsibility.

The Responsibility Name is the Responsibility ID the user is logged in under when drilling through to the source system. Select the responsibility name *only* to drill through: otherwise, leave it blank.

- 2 Repeat the above process as necessary for all selected ledgers in the source system.
- 3 Click Save.

Working with Source Accounting Entity Groups

An accounting entity group is a logical grouping of common accounting entities (for example, the same Chart of Accounts, the same Calendar, or the same currency). Accounting entity groups are used to extract data from multiple accounting entities in a single data rule execution. They facilitate sharing the same data because multiple accounting entities can belong to more than one accounting entity groups.

To work with accounting entity groups, note the following:

- Accounting entity groups can be used only with data rules.
- If a data rule in a location includes an accounting entity, then the rule is constrained by the accounting entity in the definition. In this case, the data rule in the location cannot use an accounting entity group.
- If a data rule in a location has no accounting entity, then an accounting entity or an accounting entity group must be specified in the definition. The data rule execution extracts data from either a single accounting entity or from all the accounting entities in an accounting entity group.
- Write-back rules accept only an accounting entity (and not an accounting entity group) in their definition.
- Metadata rules accept only an accounting entity (and not an accounting entity group) in their definition.
- FDMEE does not enforce that entities belong to the same Chart of Accounts.

You view and maintain accounting entity groups using the Entity Groups tab in the Source Accounting Entities feature. The Entity Groups tab consists of two regions: Entity Groups, to which you can add a new group, and the Entity Groups Entities, from which you can add accounting entities.

- To add an source accounting entity group:
- 1 Select the **Setup** tab, and then under **Register**, select **Source Accounting Entity**.
- 2 In Source Accounting Entities, select the Entity Groups tab.
- 3 Select the source system type.
- 4 In the Entity Groups grid, click Add.
 - Blank Name and description rows are added at the top of the grid.
- 5 In Name, enter the name of the accounting entity group.

- 6 In **Description**, enter a description of the accounting entity group.
- 7 In the Entity Group Entities grid, select the accounting entities to add to the accounting entity group.

The source accounting entities are displayed in the Entity Group Entities grid. You can click the column header to sort any column. The Entity Group Entities grid displays the following columns:

For Fusion and E-Business Suite source systems:

- Select—A check mark indicates that the accounting entity (ledger) is available.
- Accounting Entity—Name of the ledger
- Chart of Accounts—Chart of accounts name
- Currency—The functional currency of the Fusion and E-Business Suite ledger
- Calendar—The Fusion and E-Business Suite ledger calendar. The Fusion and E-Business Suite ledger is a collection of chart of accounts, currency, and calendar. For example, 4–4–5, Weekly, Accounting, and so on.
- Responsibility Name—Displays the general ledger drill-through responsibility. The
 drill-through responsibility must be set in FDMEE to enable users to drill through to
 E-Business Suite. The responsibility selected must have the authority to view summary
 journals and journal details for the selected ledger in the E-Business Suite.

For PeopleSoft source systems:

- Select—Select the check box to make the business unit available for the accounting entity group.
- Business Unit—Business unit name
- Currency—The base currency for the business unit
- 8 Click Save.

Working with Import Formats

Subtopics

- Defining the Import Format
- Defining Import Format Mappings
- Adding Import Expressions
- Defining the Data Load Mappings
- Defining the Write-Back Mappings (EBS and PeopleSoft only)
- Defining Import Formats for file-based Mappings

You define import formats to map source segments or chartfields to target dimensions. You can create import formats to:

 Define the import format mapping between the source system segments or chartfields and the dimensions (used in the metadata load and data load process) of the EPM target application.

- Define the import format mapping between the dimensions and the source system segments or chartfields (used in the budget write-back process) of the EPM target application.
- Define and maintain import format information and import format mapping information for: columns in the source system and EPM target applications for source adapter based integrations.

You define separate import formats for data load rules and write-back rules. See "Defining Data Load Rules to Extract Data" on page 143 and Chapter 6, "Data Write-Back."

Note: You do not need to define import mappings for use with human resources source systems.

You work with import formats on the Import Format screen, which consists of three sections:

- Import Format Summary—Displays common information relevant to source and target.
- Import Format Detail—Enables you to add and maintain import format information.
- Import Format Mappings—Enables you to add and maintain import format mapping information.

Defining the Import Format

Use the Import Format summary section to view, add, and delete import format summary information.

Viewing Import Format Information

Table 9 Import Format View Options and Descriptions

View Option	Description			
View ▼	Customizes views. Options include:			
	Columns—Select the columns to display in the data grid including:			
	○ Show All			
	o Name			
	Source System			
	Accounting Entity			
	Source Adapter			
	 Target Application 			
	Show All			
	Name			
	Source System			
	Target Application			
	• Reorder Columns —Use to change the order of the columns. When you select this option, the Reorder Columns scree is displayed. You can select a column and then use the scroll buttons on the right to change the column order.			
	 Detach/Attach—Detaches columns from the data grid. Detached columns are displayed in their own window. To return to the default view, select View, and then, click Attach or click Close. 			
	Query by Example—Use to toggle the filter row. Use the filter row to enter text to filter the rows that are displayed for a specific column. To clear a filter, remove the text to filter by in the text box, and then click Enter. All text that you enter is case sensitive.			

Adding Import Formats

Three types of import formats can be added based on the source system type:

- Source ERP
- Source adapter
- File

When adding formats, note that specific import format detail fields are available or unavailable based on the source system type.

- To add an import format for an accounting entity:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 In the Import Format summary task bar, select Add.

In the upper grid of the Import Formats screen, a row is added.

3 In Name, enter a user-defined identifier for the import format.

The value in this field cannot be modified after a mapping is created for this import format.

4 In the **Source System**, select an accounting entity source system type.

5 In Accounting Entity, select the accounting entity.

For Fusion and E-Business Suite source systems, the Accounting Entity is the ledger. For PeopleSoft source systems, the Accounting Entity is the business unit.

- 6 In Target Application, select the target application.
- 7 In **Description**, enter a description of the import format.
- 8 Define the import mappings from the source system.

See "Defining the Data Load Mappings" on page 93.

For information on one-to-one mappings and many-to-one mappings, see "Defining Import Format Mappings" on page 88.

- 9 Define the import mappings from the EPM application.
- To add an import format for a source adapter:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 On the Import Format summary task bar, select Add.

In the upper grid of the Import Formats screen, a row is added.

3 In Name, enter a user defined identifier for the import format.

The value in this field cannot be modified once mapping has been created for this import format.

- 4 In Source System, select the source adapter.
- 5 In Source Adapter, select the source adapter name.
- 6 In Target Application, select the target application.
- 7 In Drill URL, enter the URL that identifies the URL to use for drilling through when using this import format.
- 8 In the **Description**, enter a description of the import format.
- 9 Define the import mappings from the source system. See "Defining the Data Load Mappings" on page 93.

For information on one-to-one mappings and many-to-one mappings, see "Defining Import Format Mappings" on page 88.

- 10 Define the import mappings from the EPM application.
- To add an import format for a file-based data load system:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- In the Import Format summary task bar, select Add.

In the upper grid of the Import Formats screen, a row is added.

3 In Name, enter a user defined identifier for the import format.

The value in this field cannot be modified once mapping has been created for this import format.

- From the **Source System** drop-down, select **File**.
- From File Type, select the format of the file.
 - Fixed
 - Delimited—Individual data values use delimiters, such as quotation marks, commas to separate two names of data,
 - Multi-period—Delimited text or Excel files that can contain multiple periods.
- If the file type is delimited, in the File Delimiter field, select a type of delimiter.

Available delimiter symbols:

- , (comma)
- ! (exclamation)
- ; (semi-colon)
- : (colon)
- (pipe)
- tab
- In Target Application, select the target application.
- In Drill URL, enter the URL used for the drill through.

To display the **Drill URL** editor, click



The Drill URL consists of the following values:

- URL Name
- URL Prompt
- URL—Enter the URL without the server and port information. The URL must contain the parameter name and column name from the TDATASEG table enclosed in the symbol \$.

Enter the URL without the server and port information. The URL must contain the parameter name and column name from the TDATASEG table enclosed in the symbol \$.

For example, enter: LEDGER_ID=\$ATTR1\$&GL_PERIOD=\$ATTR2\$

In the above example the value of ATTR1 is passed as a value for the LEDGER_ID parameter, and ATTR2 is passed as a value for the GL PERIOD parameter. Note that parameters are separated by the "&" character.

In some cases, the you may want to refer to the column name in the FDMEE staging table based on the source mapping that is defined in the import format. When this is the case, you specify the source column name as the parameter for the URL, and this instructs the system to translate the source column name to the related column in TDATASEG when the URL is parsed. To specify a source column from the import format as the value for a parameter in the URL, enclose the value with \$\$ on each side

For example, enter COMPANY CODE=\$\$RBUKRS\$

In the above example, the value for COMPANY_CODE parameter is determined using the import format. If RBUKRS is mapped to the ENTITY dimension, then the value of the ENTITY column is passed as the parameter.

To specify the request-response between a client and server for the drill URL format, type either:

- GET—Form data is encoded into the URL. For example, specify GET@http://www.oracle.com/.. If no method is specified, then GET is the assumed request-response.
- POST—Form data is displayed in the message body. For example, specify POST@http://www.oracle.com/.
- 9 In **Description**, enter a description of the import format.
- 10 Click Save.

To define import formats for file-based mappings, see "Defining Import Formats for file-based Mappings" on page 96.

Deleting an Import Format

- To delete an import format:
- 1 Select the Setup tab, and then under Integration Setup, select Import Format.
- In Import Format, from the Import Format summary grid, select the import format, and then, click Delete.
- 3 In **Delete Import Format**, click **OK**.

Querying by Example

You can filter the import formats in the Import Format summary section using the Query by Example feature. To filter by Import Format Name, ensure that the filter row is displayed above the column headers.

- To query by example:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 In Import Format, on the Import Format task bar, select A blank row is displayed above the column headers.
- 3 Enter text to filter the rows that display.

Defining Import Format Mappings

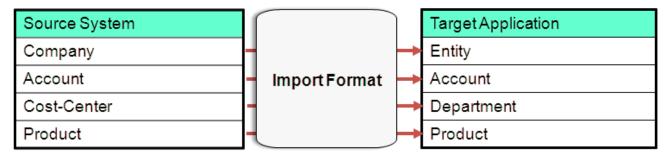
When you create an import, you define the settings and the import format mapping definition. You can create:

• One-to-One Mappings—Single Segment and Single Chartfield mappings:

- Define a simple one-to-one mapping between source segments and target dimension members.
- o Pull all members from the source value set as members into the target dimension.

Note: The topic is not available for file-based data load definitions.

The following figure shows a one-to-one mapping between segments or chartfields in a source system and dimensions in a target EPM application.



Note: Each member in the source is created as a single member in the target dimension.

• Many-to-One Mappings—You can concatenate segments or chartfields to map multiple segments or chartfields from the source into a single EPM dimension member.

When dimensions are mapped as a concatenated segment, the new dimension is created based on the traversal order that you define for the source hierarchies into the concatenated member target dimension. Up to five segments may be concatenated into the target dimension. The following table shows an example of how the segments map to dimensions. In this example, two segments (Company and Department) map to one dimension (Entity).

Segment / Chartfield	EPM Dimension		
Company	Entity		
Department			
Product	Product		
Account	Account		

Concatenating Source Dimensions for Source Applications

The Import Format allows for the concatenation of source dimensions as the source of target application dimensions. For EBS metadata loads you load the concatenated dimension and the load does not require hierarchies to be associated. However, PeopleSoft sourced applications handle metadata loads differently than EBS.

For PeopleSoft source applications you must have hierarchies associated with the concatenated dimension. This is a requirement for PeopleSoft metadata loads.

If dimension X is sourced from Product + Department, then your metadata rule requires you to have a row in the Hierarchy region which has two hierarchies, one for Product and one for Department. Traversal order must be specified as well and would be Product = First, Department = Second.

This requirement may lead to the inability to handle orphan members (i.e. members not contained within the hierarchies) due to the PeopleSoft dimension structure.

Adding Import Expressions

FDMEE provides a set of powerful import expressions that enable it to read and parse virtually any trial balance file into the FDMEE database. Advanced expressions are entered into the Expression column of the field grid of the Import Formats screen. Import expressions operate on the value that is read from the import file.

Also see "Stacking Import Expressions" on page 92 and "Processing Order" on page 92.

- To add an import format expression:
- 1 Select the Setup tab, and then under Integration Setup, select Import Format.
- In Import Format, from the Import Format Mapping grid, select the file-based source column.
- 3 In Expression, specify the import expression.
- 4 Optional: You can also specify the expression type and value on the Add Expression field.
 - a. Click
 - In Add Expression, and then under Expression Type, select the expression type.
 The number and types of expressions available depend on the field that is being modified
 - The number and types of expressions available depend on the field that is being modified (for example, Account or Account Description).
 - c. In **Expression Value**, enter the value to accompany the expression and click **OK**.
- 5 In Import Format Mapping, click OK.

Import Expression Types

Expressions supported by FDMEE:

- "Nonstandard Numeric Sign Conventions" on page 91
- "Converting from European to U.S. Notation" on page 91
- "Padding Fields with Leading Fills" on page 91
- "Padding Fields with Trailing Fills" on page 91
- "Multiplying by Whole-Number and Decimal Factors" on page 92
- "Disabling Zero Suppression" on page 92

Nonstandard Numeric Sign Conventions

The Sign expression is used to manage nonstandard numeric sign conventions. FDMEE interprets numbers with leading and trailing minus signs and numbers within parentheses as negative numbers. You can also use other leading and trailing characters to indicate negative numbers. To define custom signs, use expressions that follow this form: Sign=[Positive String], [Negative String].

For example, when positive numbers are followed by DR (1,000.00DR), and negative numbers are followed by CR (1,000.00CR), the expression is Sign=DR, CR.

Numbers within <> are also treated as negative. For example, when you specify (100.00) and <100.00> both are treated as negative numbers.

If positive numbers are unsigned (1,000.00), and negative numbers are followed by CR (1,000.00CR), the expression is Sign=, CR.

Displaying Debit and Credit Columns

The DRCRSplit expression is used to parse split numeric columns. By default, FDMEE assumes that numeric values in Amount fields are debits. However, you can position debit values on the left and credit values on the right.

Table 10 DRCRSplit expression example

Account Number	Description	Debit	Credit
1000-000-00	Cash-Operating Account	68,603.91	
1010-000-00	Cash-FANB-AP		177,216.16

DRCRSplit, which enables FDMEE to interpret left-right positioning and to assign the correct sign, follows the format DRCRSplit=Mid Point of the DR and CR columns.

When the file is imported, credit amounts are assigned negative signs (and thus are interpreted as positive), and debit amounts are unchanged (and thus are interpreted as negative).

Converting from European to U.S. Notation

The Fill=EuroToUS expression is used with the Amount field to trigger a number format conversion from (,,) to (,,) format.

Padding Fields with Leading Fills

The FillL=LeadingFill expression is used to fill fields with leading characters. Text values that are shorter than the specified fill expression are padded, as directed by the fill expression.

Padding Fields with Trailing Fills

The Fill=TrailingFill expression is used to fill fields with trailing characters. Text values that are shorter than the specified fill expression are padded, as directed by the fill expression. For example, when the account number is 103950- and the expression Fill=000000000 is used, the account number after import is 103950-000. Another example is an account number of 243150 with the expression Fill=1111111111. The account number after import is 243150111.

Multiplying by Whole-Number and Decimal Factors

The Factor=Value expression is used to factor source-file amounts by user-defined values.

Amounts can be multiplied by whole numbers or decimals; thus, for example, amounts can be doubled or halved.

Disabling Zero Suppression

The NZP expression is used to disable zero suppression during the data-load process. By default, FDMEE bypasses accounts in the trial balance that have zero balances. In certain circumstances, you may want to load all accounts, to ensure that values that should be zero are replaced. You enter NZP in the Expression column of the Amount field to disable zero suppression.

Stacking Import Expressions

One field can use multiple import expressions. To stack expressions, separate the expressions with a semicolon. Consider the order in which the layered expressions are processed.

For example, to stack the import expression for the General Ledger Amount column, enter the expression: Script=ParseAcct.py;Fill=0000000.

Defining a Multiple Period Data Load in Import Formats

When you define the mapping for an Amount column, you specify the number of amount columns in the Expression field. To do this, you specify the Columns=start,end values. For example when column 11 through 22 are amount columns, then specify Column=11,22.

Note: The Expression field ignores the Field Number value when you provide a column expression for a multiperiod data load.

Note: Amounts have to be in contiguous columns in the data file.

Processing Order

For all fields except the Amount field, FDMEE processes stacked expressions in the following order:

- 1. Script
- 2. Fill or FillL

For the Amount field, FDMEE processes stacked expressions in the following order:

- 1. DRCRSplit
- 2. Fill=EuroToUS
- 3. Script
- 4. Sign
- 5. Scale

Defining the Data Load Mappings

When you define an import format, you define the import format mappings for the metadata and data rule from the ERP source system.

- To define the ERP system import format mappings:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 In Import Format, if necessary, select the Data Load Mapping tab.
- 3 For Fusion and E-Business Suite source systems: Select general ledger source segments.

You can concatenate segments to map multiple segments from the source into a single EPM target dimension. The drop-down list displays all of the source system segments for the accounting entity that you selected when you defined the import format details.

Note: Verify your entries and ensure that, when you enter Segment 1, Segment 2, and Segment 3 that, the segments are displayed in sequence. For example, Segment 3 should not come before Segment 1. Entering segments out of sequence can cause an error.

Note: For Financial Management applications, concatenate segments are *not* available when your target dimension is an Entity dimension type. Typically, the intercompany segment in Fusion and E-Business Suite or affiliate in PeopleSoft is related to a company or business unit. If concatenation were allowed, you could not determine the ICP dimension value.

4 For PeopleSoft source systems: Select chartfields as necessary.

The drop-down list displays the source system chartfields for the Accounting Entity (business unit) that you selected for the import format.

- 5 Enter the target EPM dimension name.
- 6 Repeat the steps above as necessary.
- 7 When the target applications is a FDM or ARM data load type: you can set the concatenation member by specifying the concatenation character.

For non-FDM and ARM data loads and metadata loads, the underscore character "_" is used to concatenate two segments. For FDM and ARM data loads, you cannot use an underscore character with a mapping that uses the "Like" method. Therefore, for FDM or ARM mappings, specify a different concatenation character for the data load mapping details.

- 8 Click Save.
- 9 For Oracle Fusion, E-Business Suite, and PeopleSoft Enterprise Financial Management source systems: Define the EPM application import mappings or data write-back.
- 10 For PeopleSoft Human Capital Management source systems: The import format setup is complete.

Defining the Write-Back Mappings (EBS and PeopleSoft only)

When you define an import format, you can also define the import format mappings from the EPM application for the data write-back rule. In the case where you have imported data from your source system and want to write-back, the auto create option identifies the original data load mapping and creates the reversal of the mapping.

When you create the write-back mapping definition, map one dimension to one segment mapping where no data load mapping is available. If a data load mapping is available, then that is the basis for the transformation. Inbound import formats are defined using the Data Load Mapping tab, and write-backs are defined using the Write-back Mapping tab.

- To define the write-back import format mappings:
- 1 In Import Format, select the Import Name.
- 2 Select the Write-back Mapping tab.
- 3 Optional: Click Autocreate to automatically create import mappings from the EPM application dimension (source) to the target ERP source system segment or chartfield. Then, click OK.

In scenarios where data load mappings are available (metadata or data was sourced from the ERP source system using FDMEE) the same mappings can be reverse-engineered to identify the EPM target application to ERP source system mappings.

Example 1: Single Segment Mappings

Table 11 Mappings Set for the ERP Source System to EPM Target Application

Segment 1	Segment 2	Segment 3	Segment 4	Segment 5	Dimension
Company					Entity
Cost Center					Department
Account					Account
Product					Product
Location					Location

When you click the **Auto Create** button, FDMEE automatically identifies the inward mappings and populates the EPM-to-ERP mapping as shown in the following example. In the case, FDMEE splits out the department back into the related cost center and location that was mapped in the forward mapping.

Table 12 Automatic Creation of the EPM Target Application Mapping

Source Dimension	Target Segment or Chartfield
Entity	Company
Department	Cost Center
Account	Account

Source Dimension	Target Segment or Chartfield
Product	Product
Location	Location

Note: When EPM Target Application mappings are automatically created, they extract the appropriate segment from a concatenated dimension and split it back when writing back the data.

Example 2: Concatenated Segment Mappings

Table 13 Mappings Set for the ERP Source System to EPM Target Application

Segment 1	Segment 2	Segment 3	Segment 4	Segment 5	Dimension
Company					Entity
Location	Cost Center				Department
Account					Account
Product					Product

When you click the **Auto Create** button, FDMEE automatically identifies the inward mappings and populates the EPM-to-ERP mapping as shown in the following example:

Table 14 Automatic Creation of the EPM Target Application Mapping

Source Dimension	Target Segment or Chartfield
Entity	Company
Department	Cost Center
Department	Location
Account	Account
Product	Product

4 If you are not using the "Auto Create" option:

- a. Select each source dimension.
- b. For each dimension, enter the target segment or chartfield name.

Tip: For E-Business Suite, you can use the same import format for multiple locations when you have multiple source accounting entities that use the same chart of accounts and are mapped to the same target applications. For PeopleSoft, you can reuse the same import format for multiple locations when you have source business units that share the same SETID for all chartfields and are mapped to the same target applications.

5 Click Save.

Defining Import Formats for file-based Mappings

When the source system is a file-based data load, you define the settings and the import format mapping definition.

You can add a mapping row to the import format definition based on the following options:

- Skip
- Currency
- Attribute
- Description
- Dimension Row

 Table 15
 Import format definition options

Import Definition Option	Description					
Skip	The skip option is used to indicate rows in the input file that should be skipped. For example, rows with no data, negative numbers, or for specific accounts. The specification for a skip row is defined in the same way as that for a data row, and the system looks for the exact textual match for text entered in the expression field in the indicated location in the input file.					
	The system automatically skips rows in the input file that have "spaces" and "non-numeric" characters in the amount location, so a skip specification is only needed when non-amount data is present in a row of the input file in the same location as the amount. For example, the input file may contain a label named "date". An entry should be added for a skip row to indicate the starting column of the text "date", the length of the text, and the exact text to match.					
	The Skip row option is only available when the file type is fixed, and not delimited.					
Attribute	The TDATASEG table includes 13 attribute columns that you can load from the selected input file. You can either provide the location of the attribute in the input file by specifying the starting location and length, or plug the value during processing by entering the value for the attribute in the expression field. If the value for the attribute is entered in the expression field, then a starting location and length are not needed.					
	The attribute fields are generally used to help compose a drill-through URL or for history or documentation needs. You may want to populate the attribute field to support searching and filtering in the Data Load Workbench.					
Description	The TDATASEG table includes two description columns, and you can load these columns in the same way as the attribute columns. You can specify the location in the input row that contains a description, or specify an explicit value by entering it in the expression field in the mapping table.					
Currency	FDMEE supports the ability to load data that is of a currency different from the default currency of the selected location. This option allows you to specify the location in the input line that specifies the currency for the related amount field. For the file import format, either specify a currency on each row of data, or make sure that a currency is specified in the location that uses the selected import format.					
	Note: You may encounter issues with loading data when the currency is not specified correctly.					
Dimension	FDMEE supports multiple entries for a dimension in the import format when the dimension specification is spread between multiple locations on the same line. This feature enables you to concatenate fields for file-based data. To use this option, select the dimension, start and end positions, and the expression.					

- To define an import format for data files with a fixed length:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 In Import Format, from the Import Format summary grid, select the file,
- 3 In the **Import Format Detail** grid, select the type or row to add from the **Add** drop-down.

Available options:

- Skip Row
- Currency Row
- Attribute Row
- Description Row
- Dimension Row
- 4 In the **Start** field, specify where on the column the file starts.
- 5 In the **Length** field, enter the length of file.
- 6 In the **Expression** field, enter the expression that overwrites the contents of the file.

When entering a constant, you must still enter a starting position and length. Use a start position of "1" and a length of "1."

For more information, see "Adding Import Expressions" on page 90.

- 7 Click Save.
- To define an import format for delimited data files:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Import Format**.
- 2 In Import Format, from the Import Format summary grid, select the file.
- 3 In the Import Format detail grid, from Add, select the type or row to add.

Available options are:

- Currency Row
- Attribute Row
- Description Row
- Dimension Row
- 4 In Field Number, enter the field to import.

The Expression field ignores the Field Number value when you provide a column expression for a multiple period data load.

5 In the **Expression** field, enter the expression that overwrites the contents of the field.

For more information, see "Adding Import Expressions" on page 90.

6 Click Save.

Defining Locations

A location is the level at which a data load is executed in FDMEE. A Location is associated with one source system, but you can import data from multiple ledgers from that system. Each Location is assigned an Import Format, one or more Data Load Rules, and a Data Load Mapping. You define locations to specify where to load the data. Additionally, Locations allow you to use the same Import Format for more than one target application where the dimensionality of the target applications is the same.

The Location feature also enables you to specify free form text or a value using the integration option feature. Text or values entered for a location can be used with your FDMEE scripts.

Note: You can create duplicate locations with the same source system and application combination.

- To create, edit, and delete import locations:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Location**.
- 2 In Location, click Add.
- 3 Enter the location name.
- 4 Enter or click hto select the Import Format.

The source system is automatically populated based on the import format.

5 Optional: Enter or click to select the Accounting Entity. (For E-Business Suite, select the ledger. For PeopleSoft, select the business unit.)

If the accounting entity is selected here, then in the Data Load Rules, accounting entity populates automatically. See "Defining Data Load Rule Details" on page 144.

If the accounting entity is not specified here, you can specify the accounting entity in data rules. Doing so enables you to load data from multiple ledgers to business units from one location.

You can use locations with multiple import formats. For example, you can define the import format for the first location, Ledger 1. Then, define another import format for Ledger 2. In this case, you would create multiple locations with the same import format. You can also define multiple locations for target applications that are the same. In this case, you can define multiple locations for each business unit or ledger and reuse the import format.

6 Enter or click A to select the Target Application.

Note: When you create a location using a PeopleSoft Human Capital Management source system, the Functional Currency column displays "Unlike PeopleSoft Enterprise Financial Management source systems, when FDMEE human resources data loads occur, there is no break out of amount by functional, entered, or translated currencies.

7 In Functional Currency, specify the currency of the location.

8 In Parent Location, enter the parent assigned to the location.

Parent mapping used to share mappings with other locations. Enter mappings at the parent location, and the related locations can use the same mappings. Multiple locations can share a parent. This feature is useful when multiple locations use one chart of accounts. Changes to a child or parent mapping table are applied to all child and parent locations.

9 In Data Value, specify the extra dimension that is used only for integration with multi-dimensional target systems.

This dimension is associated with a data load location. In Financial Management, the data value is the value dimension. WhenFDMEE creates the load file, the dimension value is entered for every data line that is loaded by the location. For example, the Data Value dimension is associated with the Value dimension in Financial Management. By default, when no value is entered in this field, for integration with Financial Management, the Data Value <EntityCurrency> is the default value.

When Search is selected, FDMEE connects to the Financial Management to get a list of valid data values. FDMEE takes the values from Financial Management and adds rows created by FDMEE that are a concatenation of the original value and "Adjustment Data Values". FDMEE uses these newly created rows to manage journal loading to Financial Management.

The rows that FDMEE creates in the Data Value selection screen are:

- [Contribution Adjs];[Contribution Adjs]
- [Contribution Adjs];[Parent Adjs]
- [Contribution Adjs];<Entity Curr Adjs>
- [Contribution Adjs];<Parent Curr Adjs>
- [Parent Adjs];[Contribution Adjs]
- [Parent Adjs]; [Parent Adjs]
- [Parent Adjs];<Entity Curr Adjs>
- [Parent Adjs];<Parent Curr Adjs>
- <Entity Curr Adjs>;[Contribution Adjs]
- <Entity Curr Adjs>;[Parent Adjs]
- <Entity Curr Adjs>;<Entity Curr Adjs>
- <Entity Curr Adjs>;<Parent Curr Adjs>

10 **Optional**: In **Logic Account Group**, specify the logic account group to assign to the location.

Logic groups contain one or more logic accounts that are generated after a source file is loaded. Logic accounts are calculated accounts that are derived from the source data.

11 Optional: In Check Entity Group, specify the check entity group to assign to the location.

When a check entities group is assigned to the location, the check report runs for all entities that are defined in the group. If no check entities group is assigned to the location, the check report runs for each entity that was loaded to the target system. FDMEE check reports retrieve values directly from the target system, FDMEE source data, or FDMEE converted data.

12 Optional: In Check Rule Group, specify the check rule group to assign to the location.

System administrators use check rules to enforce data integrity. A set of check rules is created within a check rule group, and the check rule group is assigned to a location. Then, after data is loaded to the target system, a check report is generated.

13 Click Save.

14 **Optional**: Perform these tasks:

- To edit an existing location, select the location to modify, and then make changes as necessary. Then, click Save.
- To delete a location, click **Delete**.

When a location is deleted, the location is removed from all other FDMEE screens, such as Metadata, Data Load, and Data write-back.

Tip: To filter by the location name, ensure that the filter row is displayed above the column headers. (Use to toggle the filter row.) Then, enter the text to filter.

- To specify free form text or values for use with scripts:
- 1 Select the **Setup** tab, and then under **Integration Setup**, select **Location**.
- 2 In the Location Detail, click the Integration Option tab.
- From Integration Option, and then in Integration Option 1-4, specify the free form text or value and click OK.

The information you specify is accessible from the Integration Option fields of the Location table.

Defining Period Mappings

Subtopics

- Global Mappings
- Application Mappings
- Source Mappings

You have the flexibility to use various kinds of calendars (for example, monthly, weekly, or daily) based on your business and statutory requirements. In your EPM system, you can also use different calendars, based on your application requirements (for example, different levels of periods). Because FDMEE extracts the ERP source system data to the target EPM application, the mapping relationship should be established by defining a period mapping between the source ERP source system periods and the target EPM application periods.

Before you can define data rules, define the period mappings. Period mappings define the mapping between ERP calendars and the EPM application year or periods. You can define period mappings using:

- Global Mapping—You define a global mapping in cases where you do not have many target applications getting data from multiple source systems with different types of source calendars. Use a global mapping to ensure that various periods are accommodated in the individual mapping. As a first step, define a global mapping.
- Application Mapping—When you have multiple target applications, getting data from various source systems with complex period types, you can create application mappings in addition to global mappings. When you define an application mapping, you can modify the Target Period Month as necessary; for example, When in Financial Management the name is January and in Planning the name is JAN.
- Source Mapping—Specifies source period mapping for adapter based integrations.

Global Mapping—Sample Monthly Period Mapping

The following table shows how a monthly calendar from an ERP source maps to monthly periods in a target.

Note: You should define global mapping at the most granular level. For example, When you have a monthly calendar and a weekly calendar, define your global mapping at the lowest level of granularity. In this case, the period keys are at the week level. and you map weeks to months. You can create application mappings for the higher-level periods.

Table 16 Sample Monthly Period Mapping

Period Key	Prior Period Key	Period Name	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 1 2010	Dec 1 2009	January 1, 2010	Jan	Q1			FY10
Feb 1 2010	Jan 1 2010	February 1, 2010	Feb	Q1			FY10
Mar 1 2010	Feb 1 2010	March 1, 2010	Mar	Q1			FY10
April 1 2010	March 1 2010	April 1, 2010	Apr	Q2			FY10
May 1 2010	April 1 2010	May 1, 2010	May	Q2			FY10

Global Mapping—Sample Weekly Period Mapping

The following table shows how a weekly calendar from an ERP source system maps to monthly periods in the EPM application.

Table 17 Sample Weekly Period Mapping

Period Key	Prior Period Key	Period Name	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 26 2009	Jan 19 2009	January 26, 2010	Jan	Q1			FY09

Period Key	Prior Period Key	Period Name	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Feb 2 2009	Jan 26 2009	February 2, 2010	Feb	Q1			FY09
Feb 9 2009	Feb 2 2009	February 9, 2010	Feb	Q1			FY09
Feb 16 2009	Feb 9 2009	February 16, 2010	Feb	Q1			FY09

Application Mapping—Sample Target Application Sourcing from a Monthly Calendar Source

The following table shows a sample where the target application is sourcing from a monthly calendar. This mapping is performed on the Application Mapping tab.

 Table 18
 Sample Application Mapping—Target Application #1 with a Monthly Calendar Source

Period Key	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 1 2009	Jan	Q1			FY09
Feb 1 2009	Feb	Q1			FY09
Mar 1 2009	Mar	Q1			FY09

Application Mapping—Sample Target Application #2 Sourcing from a Weekly Calendar Source

The following table shows a sample where the target application is sourcing from a weekly calendar. This mapping is performed on the Application Mapping tab.

Table 19 Sample Application Mapping—Target Application #2 with a Weekly Calendar Source

Period Key	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 26 2009	Jan	Q1			FY09
Feb 2 2009	Feb	Q1			FY09
Feb 9 2009	Feb	Q1			FY09
Feb 16 2009	Feb	Q1			FY09

Note: To avoid double counting on Income Statement accounts, be sure not to define a mapping where the adjustment period of one year goes into the period of the next fiscal year.

Adjustment Period Mapping—Mapping the Period Key to the Adjustment Period

Note: When YTD is selected as the ERP source, then the adjustment period becomes the ending balance (replaces the period 12). When PTD, then the adjustment period gets added to period 12.

Table 20 Sample Adjustment Period Mapping-Mapping the period to the adjustment period

Period Key	Calendar	Adjustment Period	Description
Dec-2003	Accounting 13	13-03	Adj Period for 2003
Dec-2004	Accounting 13	13-04	Adj Period for 2004
Dec-2005	Accounting 13	13-05	Adj Period for 2005
Dec-2007	Accounting 13	13-07	Adj Period for 2007

Note: When the source is PeopleSoft General Ledger, set the adjustment period mapping with the related accounting year.

Global Mappings

You can define one global mapping to map various periods to the individual mapping.

- To define a global mapping:
- 1 Select the Setup tab, and then under Integration Setup, select Period Mapping.
- 2 In **Period Mapping**, select the **Global Mapping** tab.
- 3 Click Add.
- 4 Select the **Period Key**.
- 5 Select the Prior Period Key.
- 6 Enter the following:
 - a. Period Name; for example, August 2005.
 - b. Target Period Month; for example, August.
 - c. Target Period Quarter
 - d. Target Period Year
 - e. Target Period Day
 - f. Year Target

Note: Period dimension members in Performance Management Architect that have the "Data Storage" property set to "Label Only,", "Dynamic Calc", or "DynamicCalcandStore" are not displayed.

7 Click Save.

Application Mappings

You can define application mappings in cases where you want to define a special period mapping for a specific target application. The mappings apply to an individual target application.

- To create period mappings for an application:
- 1 Select the Setup tab, and then under Integration Setup, select Period Mapping.
- 2 In Period Mapping. select the Application Mapping tab.
- 3 In Target Application, select the target application.
- 4 Click Add.
- 5 Select the **Period Key**.
- 6 Enter the following:
 - a. Target Period Month
 - b. Target Period Quarter
 - c. Target Period Year
 - d. Target Period Day
 - e. Year Target
- 7 Click Save.

Source Mappings

Source mappings include explicit and adjustment period mappings. You can create explicit period mappings to ensure that the FDMEE periods map correctly to the source system calendar periods. An adjustment period mapping is used only when you select the **Include Adjustment Periods** option when creating the data load rule.

The Source Mapping tab consists of two areas:

- Master—Selects the source system and mapping type
- Grid—Defines the period mapping. The mapping can be defined only for periods defined on the Global Mapping. New FDMEE periods cannot be created on this tab.

For (EBS, Fusion, and PeoplesSoft) source systems, you can select explicit or adjustment systems. For all other systems, you can select only an explicit mapping.

Note: In Data Rules, you can choose between Default period mapping and Explicit Period mapping. When you choose Period mapping, then source periods are mapped based on the period key and previous period.

- To create source mappings:
- Select the Setup tab, and then under Integration Setup, select Period Mapping.
- 2 In Period Mapping, select the Source Mapping tab.

- 3 In Source Mapping, select the source system.
- 4 In Mapping Type, select Explicit.

For SAP and JD Edwards source systems, you must select **Explicit** period mappings For all other systems (for example, EBS and Fusion), you can select **Explicit** or **Adjustment**.

- 5 Click Add.
- 6 Enter the source system **Period Name**, and then, click **OK**.

Note: Period names cannot include spaces when used in a batch script.

- 7 Enter the source system **Period Key**, and then, click **OK**.
- 8 Enter the source system Calendar, and then, click OK.
- 9 Enter the source system **GL Period**, and then, click **OK**.

The GL Period Number is prefilled based on the Period Name.

- 10 Enter the source system **GL Name**, and then, click **OK**.
- 11 Optional: Enter a description for the mapping.
- 12 Click Save.
- To create source period mappings:
- 1 Select the Setup tab, and then under Integration Setup, select Period Mapping.
- 2 In Period Mapping, select the Source Mapping tab.
- 3 In **Source Mapping**, select the source system.
- 4 Click Add.
- 5 Click to select the source system Period Key, and then, click OK.
- 6 Click hto select the source system Calendar, and then, click OK.
- 7 Click hto select the source system Adjustment Period, and then, click OK.
- 8 For PeopleSoft source systems only: In GL Period Year, enter the general ledger period year.

The general ledger period year is required for PeopleSoft source systems because PeopleSoft Adjustment Periods definitions do not include a Year value. To properly map adjustment period data from PeopleSoft, define the source accounting period and fiscal year intersections for all PeopleSoft adjustment periods.

- 9 Optional: Enter a description for the mapping.
- 10 Click Save.
- To create budget period mappings (for PeopleSoft Commitment Control only):
- 1 Select Source Mapping.
- 2 In **Source Mapping**, select the source system.

- 3 Click Add.
- 4 In Mapping Type, select Budget.

Note: From PeopleSoft Commitment Control, only Budget Period Data can be extracted. The source calendar / period are determined based on control budget definition in PeopleSoft.

5 In **Period Name**, specify the period name.

You can also click \(\frac{1}{2} \) to search for the period name.

Note: Period names cannot include spaces when used in a batch script.

6 Enter the source system Calendar, and then, click OK.

You can also click \(\frac{1}{2} \) to search for the calendar name.

7 Enter the source system **GL Period**, and then, click **OK**. You can also or click to search for and select the GL period name.

The GL Period Number is prefilled automatically based on the Period Name.

- 8 **Optional**: Enter a description for the mapping.
- 9 Click Save.

Tip: To delete a mapping, select the mapping, and then, click **Delete**.

Defining Category Mappings

Subtopics

- Global Mappings
- Application Mappings

You define category mappings for categorizing and mapping source system data to a target EPM Scenario dimension member. For example, in a Financial Management application, you may have a Scenario dimension member called "Actuals" for storing actual balances from a source system. In a Planning application, the same source system data is stored using the Scenario dimension member "Current." In FDMEE, you can create one category mapping to give both one name to represent their respective scenarios.

Global Mappings

You can define one global mapping to map various Scenario dimensions to the individual mapping.

The global category mappings lets you define mappings that cross multiple applications. For example, you may have a case where a source category of an actual may map to a target of an

actual in the majority of the cases. But you may have a case where you have a target application where the actual may map to current. In this case, it provides the ability to override the global mapping on an application basis.

- To define a global category mapping:
- 1 Select the Setup tab, and then under Integration Setup, select Category Mapping.
- 2 In Category Mappings, select Global Mapping.
- 3 Click Add.

A blank entry row is displayed.

- 4 Enter the category name.
- 5 Enter the category description.
- 6 Select the category frequency.

The category frequency indicates the frequency defined in the period mapping. For example, Daily, Monthly, Quarterly, or Yearly.

7 Enter the target category.

The target category is the Scenario dimension in the application.

- 8 Click Save.
- 9 Optional: Perform these tasks:
 - To edit a mapping, select the mapping and then make changes as necessary. then, click
 Save.
 - To delete a mapping, click **Delete**.

Application Mappings

Unlike global mappings, application mappings can be defined for a target application.

- To define application category mappings:
- 1 Select the Setup tab, and then under Integration Setup, select Category Mapping.
- 2 In Category Mappings, select the Application Mapping tab.
- 3 From Target Application, select the target application.
- 4 Click Add.

A blank entry row is displayed.

- 5 Select the category.
- 6 Enter the target category or click A to search for a target category.
- 7 Click Save.
- 8 Optional: Perform these tasks:

- To edit an mapping, select the mapping and then make changes as necessary. Then, click **Save**.
- To delete a mapping, click **Delete**.

Loading Metadata

2

In This Chapter

Understanding General Ledger Integration	109
Requirements	109
Required Dimensions	110
Member Properties Sourced from the ERP System	111
How Dimensions are Processed	114
How Languages are Processed	115
How Currencies are Processed	116
Loading Source System Hierarchies into EPM Dimensions	117
Defining Metadata Rules	118

Understanding General Ledger Integration

FDMEE supports loading data into EPM applications from general ledger source systems and also writing back budget data from target EPM Planning applications to general ledger source systems.

- Loading data from the General Ledger Source System FDMEE supports loading data from general ledger source systems. FDMEE can load both metadata (dimension members and hierarchies from Oracle E-business Suite, Oracle Fusion Financials and PeopleSoft Enterprise Financial Management only) and data from the ERP source systems.
- Writing back budget data to the General Ledger Source System FDMEE enables you to
 extract budgets from Planning, Essbase aggregate storage, and Essbase block storage, and
 load them into the source general ledger system. This helps you maintain one single system
 of record. Budget Write-back is not available for SAP and JD Edwards.

Requirements

Before you begin using FDMEE, keep in mind the following requirements:

• Verify that you have met the EPM *dimension* requirements:

EPM applications can be built with any combination of dimensions. The combination must include required dimensions for the selected application. "Member Properties Sourced from the ERP System" on page 111 describes how member properties are sourced from the ERP source system.

- Verify that you have met the EPM *member* requirements:
 - Duplicate Members—To avoid issues with duplicate member names, as a best practice, include a unique prefix or suffix for each dimension so each member is always unique.
 - Duplicate Alias members—When your application has duplicate Alias members, it is important to remove any duplicates in the target application or validation errors occur when you deploy the application in Performance Management Architect.

Note: Source descriptions must be unique to avoid alias validation errors withPerformance Management Architect.

When moving dimensions and members from a source system into a target EPM application, it is important to understand the naming restrictions. For Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*. For Planning, see the *Oracle Hyperion Planning Administrator's Guide*. For Financial Management, see the *Oracle Hyperion Financial Management Administrator's Guide*.

Required Dimensions

Hyperion EPM applications can be built with any combination of dimensions, as long as the combination includes those required for the selected application. For example, Planning requires different dimensions to be present in an application than Financial Management.

For detailed information on dimensions required and properties for Performance Management Architect applications, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*. For Classic Planning applications, see the *Oracle Hyperion Planning Administrator's Guide*. For Classic Financial Management applications, see the *Oracle Hyperion Financial Management Administrator's Guide*. For Classic Essbase, see the *Oracle Essbase Database Administrator's Guide*.

The following Hyperion dimensions require some special considerations when integrating with FDMEE:

- Account
- Currency
- Entity
- Scenario
- Version
- View
- Year
- Period

In addition to the above list, you should review properties set by FDMEE in the Custom dimension. See "Custom" on page 114.

For information on special considerations and requirements for Public Sector Planning and Budgeting, see Chapter 7, "Loading Human Resources Data."

Member Properties Sourced from the ERP System

Subtopics

- Account
- Entity and Intercompany
- Scenario
- Version
- View
- Year and Period
- Alias
- Custom

For each required dimension, specific properties must be defined. The required dimension properties relate to Planning, Financial Management, or Essbase applications, and in some cases both.

Note: FDMEE sets some of the required properties, but not all.

Account

The Account dimension represents a hierarchy of natural accounts. Accounts store financial data for entities and scenarios in an application. Each account has a type, such as Revenue or Expense, that defines its accounting behavior. The Account dimension is mapped from the source accounting entity to the EPM Account dimension as defined in the dimension mapping definition for the selected chart of accounts or business unit. The properties set by FDMEE are shown in the following table. (Any properties not set are defaulted by either the application or Performance Management Architect).

Property	Application Type	Population Method / Value
Consolidation Account Type	Consolidation	Populated from the account type in the source accounting entity with the domain of revenue, expense, asset, or liability. When the source type is equity, it is changed to liability for use by Financial Management applications.
Account Type	Planning	Populated from the account type in the source accounting entity with the domain of revenue, expense, asset, liability or equity.
Variance Reporting	Planning, Essbase aggregate storage, and Essbase block storage	Set to "Expense" when account type is expense, otherwise set to "NonExpense." (NonExpense is the default).
Description, Display String	System	Populate from source accounting entity description.

Property	Application Type	Population Method / Value
Time Balance	Planning, Essbase aggregate storage,	For income statement accounts, (revenue and expenses) set to "Flow" for Planning applications.
	Essbase block storage	For Essbase aggregate storage and block storage applications, set to "Last."
		For balance sheet accounts (asset, liability, and equity), set to "Balance."
		These properties can be set when you create metadata rules. See "Defining Metadata Rules" on page 118.

Entity and Intercompany

The Entity dimension represents the organizational structure of the company, such as the management and legal reporting structures. Entities can represent divisions, subsidiaries, plants, regions, countries, legal entities, business units, departments, or any organizational unit. You can define any number of entities.

The Intercompany dimension represents all intercompany balances that exist for an account. This is a reserved dimension that is used in combination with the Account dimension and any custom Financial Management dimension.

Financial Management requires that members of the Entity dimension have the IsICP property set for those members that are intercompany members. When an application is populated, Performance Management Architect populates the ICP (intercompany) dimension with the appropriate members based on the Entity members that are flagged as ICP (intercompany) entities.

For E-Business Suite, there are two scenarios for mapping source segments to the Entity dimension: 1) an intercompany segment exists in the source chart of accounts, and 2) an intercompany segment does not exist in the source chart of accounts. For PeopleSoft, the business unit is mapped to the entity and the affiliate is mapped to the ICP.

The properties set by FDMEE are shown in the following table. (Any properties not set are defaulted by either the application or Performance Management Architect.

Property	Application Type	Population Method / Value
Name	Consolidation, System	Populated from the code/value in the source accounting entity.
Description	System	Populated from the name in the source accounting entity.
IsICP	Consolidation	If the intercompany segment exists in the source, then this flag is set automatically per the rules defined.
		If the intercompany segment does not exist, then you specify how this property is set. See "Entity and Intercompany" on page 112.
		Note: For ICP transaction data to load correctly, you must manually set the property ISICP ="Y" for those accounts participating in ICP. In Performance Management Architect, you can use the Property Grid to modify the property. If using Financial Management Classic application administration, extract the metadata, update, and then re-import it back. After modifying the property, you can load data correctly for ICP transactions.

Property	Application Type	Population Method / Value
Currency	Consolidation, Essbase, Planning	For Financial Management target applications: The entity currency is set based on the default defined in the mapping rule for the Entity dimension. (All members are assigned the same currency.) As the administrator, it is important to make sure that the functional currency of the source is consistent with the default Entity currency.

Note: These are the only properties that are set as part of the FDMEE integration, all others are defaulted when creating new members. If a property was originally set by FDMEE, and you later change the property, the property is overridden.

Scenario

The Scenario dimension represents a set of data, such as Budget, Actual, or Forecast. For example, the Actual scenario can contain data from a general ledger, reflecting past and current business operations. The Budget scenario can contain data that reflects the targeted business operations. The Forecast scenario typically contains data that corresponds to predictions for upcoming periods. A Legal scenario can contain data calculated according to legal GAAP format and rules.

Version

The Version dimension is specific to EPM applications and usually does not have a source in the source accounting entity. Since it is required, you must specify the necessary default value in the member mapping by using the "Like" mapping type. When defining the data rule in FDMEE, select the desired "Version" to include with the extracted data. Since the Version dimension is not extracted from the source system, it is not necessary to define specific properties.

View

The View dimension represents various modes of calendar intelligence; for example, Periodic, Year-to-Date, and Quarter-to-Date frequencies. FDMEE only extracts data that is below the quarter level. You select the view as part of the data rule definition, and when the data is extracted, it includes the View selection as the value for the dimension on each row. See "Defining Data Load Rules to Extract Data" on page 143. Since the View dimension is usually not extracted from the source system, it is not necessary to define specific properties. However, before the data extraction process, you must create all members in the View dimension manually.

Year and Period

The mapping between the source system calendar and the Year and Period dimensions is managed using the period mapping feature described in "Defining Period Mappings" on page 100. Before you perform period mapping, create the necessary Year and Period members. In FDMEE, you select the calendar periods to include in the data extraction process and on the Period Mapping page to define the appropriate target year and target period dimensions to assign

to the data. Since the Year and Period dimensions are not extracted from the source system, it is not necessary to define specific properties.

Note: For Planning applications, it is required that you must have the same number of children in each branch of the Period dimension. For example, Q4 has October, November, December children and an adjustment period in Performance Management Architect.

Alias

For Planning and Essbase, the Alias dimension or table is required to support languages. Keep in mind these special considerations:

- The Alias dimension must include a member named "Default".
- If the dimension name is not the same as the Alias name in a Performance Management Architect Planning application, the drill through landing page does not return any data.
- When creating Alias table members in a dimension, you should define them with the same name that is displayed in Oracle Fusion, E-Business Suite, or PeopleSoft. This is the value of the NLS_LANGUAGE column.

Custom

The properties set by FDMEE are shown in the following table. (Any properties not set are defaulted by either the application or Performance Management Architect)

Property	Application Type	Population Method / Value	
Name	System	In E-Business Suite, this value is populated from Segment Name. In PeopleSoft Enterprise Financial Management, this value is populated from the chartfield value.	
Description	System	In E-Business Suite, this value is populated from Segment Value. In PeopleSoft Enterprise Financial Management, this value is populated from the chartfield value.	

How Dimensions are Processed

For Performance Management Architect applications, the dimension extract process populates the interface tables with dimension members based on the mapping rule details created in FDMEE. For Classic Planning and Financial Management applications, the dimension extract process populates dimension members directly into the application based on the mapping rule details created in FDMEE.

As part of the extract process, the dimension members are directly loaded into the target Performance Management Architect dimension, with specific properties defined as described in "Member Properties Sourced from the ERP System" on page 111. In addition to loading dimension members, the related alias entries are also loaded to provide the appropriate language support.

Shared dimensions from EPMA are not selected directly from the shared library for loading. FDMEE presents the dimensions in the target application for inclusion in a metadata rule, and then updates the shared library automatically when updating the target application. Dimensions in the shared library cannot be directly loaded, but must be included in a target application to enable loading from an ERP source.

The FDMEE dimension extract process:

- 1. Extracts the general ledger segment or chartfield value sets from the source system.
 - Only general ledger segment value set members or chartfield members that are relevant to the source chart of account segments or chartfields mapped to Classic Financial Management, or Planning are extracted.
 - The members are loaded into a staging table on the target instance. Before loading them
 into the staging table, FDMEE assigns the segment values a prefix defined for the
 corresponding EPM application dimension.
- 2. Processes dimensions mapped to single segments or chartfields.

For Classic applications, dimensions are loaded directly into the target application. The interface tables for dimensions map to a single general ledger segment or chartfield. This consists of filtering the data from the staging table loaded in step 1, on the basis of the segment value set mapped to a particular dimension, and loading the corresponding dimension member interface table and dimension member property array table (for aliases).

In most cases, dimensions are mapped as a single segment in Fusion and E-Business Suite source systems or single chartfield in PeopleSoft source systems from the source chart of accounts to a target dimension and you select the starting node in the source dimension as the basis for the new dimension.

3. Processes the dimensions mapped to multiple segments or chartfields.

For Performance Management Architect, the member interface tables are populated for the dimensions mapped from more than one E-Business Suite general ledger chart of accounts segment or PeopleSoft chartfield. This requires the individual segment values to be concatenated to create the dimension member values.

The dimension extract process creates the required member entries, properties, and alias entries when they exist in the source system, and then applies defaults to those properties when they do not exist. Users should update member properties when the source system value was not available or in cases where a different value is desired.

In some cases, you can create target dimension members based on the concatenation of one or more source segments. When dimensions are mapped as a concatenated segment, the new dimension is created based on a user-defined traversal order of the source hierarchies into the concatenated member target hierarchy.

How Languages are Processed

As part of the source system registration, FDMEE gets the list of available source system languages, in addition to the base language. The base language is typically the language selected

when the ERP source system is installed. Additional available languages that are not the base language are referred to as the "enabled languages."

Languages from the source system that are mapped to the languages defined in the target application are independent of the languages available for selection via the FDMEE browser selection. It is possible that the languages available in the browser differ from the languages available in the ERP source system and the target EPM application. For information on languages that FDMEE supports, see the *Oracle Hyperion Enterprise Performance Management System Certification Matrix*.

When you register a target application for use with FDMEE, the Default Language column on the Target Application Registration page is used as follows:

- The languages displayed in the Default Language drop-down list are FDMEE supported languages. These languages are mapped behind the scenes to the ERP source system languages.
- The Alias dimension in Essbase and Planning applications has a required "Default" member. The FDMEE language you select when registering a target application is automatically mapped to the "Default" member. Since the FDMEE language is mapped to the source language for the member description, you map either the base or enabled source language in the source system to the "Default" alias member. During processing, all other languages are mapped to the other alias members as long as the alias member matches the FDMEE source language for the member description exactly.

Note: Language processing is the same for Essbase and Planning applications.

Note: Financial Management languages are processed based on the Default Language you select on the Target Application Registration page.

For information on registering target applications, see "Registering Target Applications" on page 63.

How Currencies are Processed

When you define a data rule, you can specify how to extract exchange rates from the ERP source system. If your target application has the multi-currency option enabled, you can specify how to process exchange rates.

All rates are extracted and inserted into the AIF_HS_EXCHANGE_RATES table. This table is populated using the ISO currency code for each currency from the source system. The ISO numeric code is not used in this processing.

Exchange rates are pushed into Planning based on a match between the ISO currency code in the AIF_HS_EXCHANGE_RATES table and the currencies defined in the multi-currency Planning application. (It is important to set up the currencies in the Planning application with ISO currency codes.) Then, perform any currency conversions with those rates, as nothing is recalculated as part of this process.

Any data that is coming in with the default currency of the Planning application is loaded to locale.

For Financial Management, data is loaded to value dimension specified in the Location. Typically it is set to <Entity Currency>.

Note: Exchange rates are not interfaced directly into Financial Management. You should manually access them from the AIF_HS_EXCHANGE_RATES table and insert them into Financial Management.

Loading Source System Hierarchies into EPM Dimensions

Metadata within the source system changes over time, as well as the metadata and hierarchies in the target system.

The management of hierarchies is an ongoing process, with frequent changes due to updates in business functions and organizations. When managing hierarchies between source and target systems, users generally create new hierarchies, replace old hierarchies with new hierarchies or update hierarchies.

Managing hierarchies between systems becomes difficult because of the size of the hierarchies, the latency between system updates, and the needs of operational systems versus analytical systems. When managing hierarchies as part of the general ledger integration process, keep in mind the following assumptions:

- The only operations between hierarchy management in the source system and target application is to create or update the hierarchies by merging in the target. FDMEE never deletes hierarchies or members in a target application. If additional members or hierarchies are not specified, they are ignored by FDMEE.
- When you integrate a hierarchy from the source system to the target system, select the node from the source that serves as the root node in the target.
- The integration pushes the hierarchy into the target system, and reports any errors encountered during the process.
 - You can use the Process Details page to view errors logged in FDMEE. You can also select the Log link to review the Oracle Data Integrator log file. For Performance Management Architect applications, you can also view profile creation errors in the Job Console. See "Viewing Process Details" on page 180 or the appropriate product documentation for additional information.
- A hierarchy selection in the dimension mapping is optional. However, you must at least determine how to handle members not in a hierarchy. For example, create children of a selected node, as orphans, (for Performance Management Architect only) or to not bring orphans over.

Defining Metadata Rules

Subtopics

- Defining the Metadata Rule Details
- Defining Dimension Attributes
- Managing Metadata Rules

You can create metadata rules once and rerun the rules as necessary.

For general ledger source systems:

- For Fusion and E-Business Suite source systems, the chart of accounts is the collection of general ledger segments with various value sets, which are mapped to the dimensions to pull the dimension members and hierarchies.
- Similarly for PeopleSoft Enterprise Financial Management, the chartfields are mapped to the dimensions to pull the dimension members and hierarchies.

Note: Metadata rules are not used in FDMEE integrations with human resources source systems.

Before you define metadata rules:

- Ensure that your source system data does not include special characters. Special characters are not supported in Financial Management target applications.
- Register your source systems and target applications for use with FDMEE. See "Registering Source Systems" on page 49 and "Registering Target Applications" on page 63.
- Select the source accounting entities in the registered source system. See "Selecting Source Accounting Entities" on page 80.
- Define the import format. See "Working with Import Formats" on page 83.
- Define the location. See "Defining Locations" on page 98.

Note: Performance Management Architect supports Shared and Local dimensions. FDMEE also supports Shared and Local dimensions in applications.

At a high level, follow this process to define metadata rules:

- 1. Create the metadata rule.
- 2. Select the dimension.
- 3. Define the mapping details and specify how to handle intercompany segments. See "Defining the Metadata Rule Details" on page 119.
- 4. Define dimension attributes. See "Defining Dimension Attributes" on page 122.
- 5. Optional: Define the segment hierarchies or chartfield trees to be extracted.
- 6. Save and run the metadata rule.

- 7. Optional: Check the status of the rule. See "Viewing Process Details" on page 180.
- To create metadata rules:
- 1 Select the Workflow tab, and then under Metadata, select Metadata Rule.

Note: You cannot create multiple metadata rules for the same ledger or business unit for each target application.

- 2 In **Metadata**, click ³ to select the **Location**.
- 3 Select the location, and then, click OK.

Note: A location can be created without an accounting entity. This feature allows users to load data from multiple accounting entities from a single location. In this case, accounting entities are associated with the data rules. If no accounting entity is specified for the location, the accounting entity from the import format is used.

4 Click Add.

A blank line is displayed at the top of the Dimension Mappings summary grid.

5 In the Mapping details area, from Dimension, select the dimension.

The dimensions listed are based on the import format.

When a Dimension is selected, the Dimension Classification field prefills.

- Define the mapping details for each dimension you select. See "Defining the Metadata Rule Details" on page 119.
- 7 Repeat steps 4-6 for each dimension.
- 8 Click Save.

Defining the Metadata Rule Details

Single segment or chartfield mappings define a simple one-to-one mapping between source and target dimension members. When you create single segment or chartfield mappings, you can optionally define:

- A member prefix or suffix type and value
- Segment hierarchies to extract from the source system
- Orphan member handling
- Statistical Account handling
- To define the mapping details:
- 1 Optional: In Prefix/Suffix Type, select either Prefix or Suffix.
- 2 In Prefix/Suffix Value, enter the member prefix or suffix.

Member prefixes are inserted before the source member code. Although optional, it is important to prefix the segment values with a prefix defined for the corresponding dimension when those members do not exist in the target application.

Member suffixes are inserted after the source member code.

Note: When you perform the next steps to define the hierarchy region starting parent, consider that Planning, Essbase, and Financial Management do not allow members to roll up to the parent under the same root. When extracting, specify hierarchies where every node has one parent. Fusion and E-Business Suite support instances where a segment value can roll up to two parents.

- 3 Select Concat Name to Alias to concatenate the Name and Alias.
- 4 From **EPMA Option**, select the EPMA metadata load option.

Available metadata load options:

• Merge as Primary—Merge as Primary processes all sections and adds new dimensions, members, relationships, properties and associations that exist in the source, but do not exist in the Shared Library or target application. No dimensions, members, relationships, properties, or associations are removed. If a dimension, relationship, member, or property specified in the source exists in the Shared Library or target application it is overwritten with the value specified in the source. (Members are not moved, but overwritten.) Properties not included in the source are unchanged in the Shared Library or target application.

Note: If a member has multiple instances in a dimension, the IsPrimary column specifies which instance is primary and which instance(s) are shared. The IsPrimary parameter is optional; however, we strongly recommend that you use it. If left undefined, it defaults to "True", which assumes that the member is primary. If you do not define the IsPrimary parameter or when there are duplicate members defined as primary, a warning is displayed in the Import Results file.

- Merge as Move—This mode only processes members with IsPrimary set to true, and
 ignores any others. During processing, the first line specifying a primary location for a
 member is executed, any subsequent lines specifying a different primary location are
 ignored, and a warning message is displayed.
- Replace—All new elements are added and all property updates are made. Then, any members or member relationships that are not specified in the source are deleted from the Shared Library or target application. You can also use replace mode to reorder members under a parent. Properties not included in the source are unchanged in the Shared Library or target application.

This mode does not create new Shared members unless they are defined in the import file (IsPrimary=false). This mode can detect a primary member under a new parent and process it as a move.

Note: The Replace option does not display the Reorder Type and Reorder Existing Members options.

5 For Fusion and E-Business Suite source systems:

 For Planning and Essbase applications—Account dimension mapping details. Select the Time Balance Property for Balance Sheet Accounts and Time Balance Property for Income Statement Accounts.

The time balance property specifies how the value of summary time periods is calculated. If set to Flow, it is an aggregate of all values for a summary time period as a period total. If the time balance property is set to "Balance," it is considered an ending value in a summary time period for the period total.

- b. From the **Hierarchy Region** tab, click **Add** to define the hierarchy region.
- c. Click or enter the Starting Parent.
- d. Select Base Hierarchy.

The base hierarchy indicates the part of the hierarchy that is the base and that other parts of the hierarchy that share the same parent are shared. All non-shared members of base hierarchies have the "Store Data" property set. However, shared members cannot be set to "Store Data."

e. Select a prefix or suffix for the hierarchy, and then, enter a prefix/suffix value.

The parent prefix is applied only to the parent. To enable alternate rollup hierarchies, the Prefix/Suffix value applies only to parent members. Parent members cannot be shared and must have a unique name. In addition, parent members do not store data.

6 For PeopleSoft source systems:

a. For Planning and Essbase applications—Account dimension mapping details. Select the Time Balance Property for Balance Sheet Accounts and Time Balance Property for Income Statement Accounts.

The time balance property specifies how the value of summary time periods is calculated. If set to "Flw", it is an aggregate of all values for a summary time period as a period total. If set to "Balance", it is an ending value in a summary time period as the period total.

- b. From the **Hierarchy Region** tab, click **Add** to define the hierarchy region tree.
- c. Click or enter the tree name.
- d. Click or enter the effective start date.
- e. Select Base Hierarchy.

The base hierarchy indicates the part of the hierarchy that is the base and that other parts of the hierarchy that share the same parent are shared. All non-shared members of base hierarchies have the "Store Data" property set. However, shared members cannot be set to "Store Data."

f. Select a prefix or suffix for the hierarchy. Then enter a prefix/suffix value.

The parent prefix is applied only to the parent. To enable alternate rollup hierarchies, the Prefix/Suffix value applies only to parent members. Parent members cannot be shared and must have a unique name. In addition, parent members do not store data.

- 7 For Planning only: Select the **Plan Type** for the Accounting and Entity dimensions.
- 8 Financial Management only: If you are mapping an Entity dimension, enter the following details for intercompany segments, depending on your source system:
 - Intercompany Segment Value
 - For Intercompany Default, select **Yes** or **No**.

In the scenario where the Intercompany segment is defined, the Intercompany transactions are identified based on the Intercompany segment. Typically, one set of natural accounts is required for accounting the receivable and payable among the Intercompanies.

In the scenario where there is no Intercompany segment, the Intercompany transactions should be identified based explicitly on the natural account segment. Typically, all combinations of receivable and payable accounts exist among the transacting partner companies. Only with these natural accounts are the receivable and payable position between the Intercompanies known.

9 Click Save.

Defining Dimension Attributes

The Attribute tab provides a table with a list of attributes for dimension. For each attribute, you can specify a default value. Note that the attributes vary by application type and dimension as shown below.

Note: The default values are not validated. Refer to your application documentation for valid values.

Table 21 Dimension Attributes by Application Type

Application Type	Dimension	Attribute
Planning	Account	Time Balance for Balance Sheet
		Time Balance for Income Statement
		Data Storage
		Expense Reporting
		Account Type
HFM	Account	Consolidation Account Type
		Custom 1 Top Member
		Custom 2 Top Member
		Custom 3 Top Member
		Custom 4 Top Member

Application Type	Dimension	Attribute
HFM	Entity	ISICP
		Currency
Essbase	Account	

- To specify a custom default value:
- 1 Select the Attribute tab.
- 2 Enter the default value for the attribute.
- 3 To use the custom default value, deselect the Use System Default field.
- 4 Click Save.

Managing Metadata Rules

You can perform the following tasks to manage your metadata rules:

- Edit metadata rules—See "Editing Metadata Rules" on page 123.
- Run metadata rules—See "Running Metadata Rules" on page 123.
- Check the metadata rule process details—See "Deleting Metadata Rules" on page 124.
- Delete dimension mappings or hierarchies in metadata rules—See "Checking the Metadata Rule Status" on page 124.

Editing Metadata Rules

If the metadata rule is not running, you can modify the rule.

- ➤ To edit metadata rules:
- 1 Select the Workflow tab, and then under Metadata, select Location.
- 2 Enter the Location Name or click to select the location.
- 3 Add or modify the dimension mappings or mapping details as necessary.
- 4 Click Save.

Running Metadata Rules

You can run the metadata rule to load updates and push the metadata into the target application. All submitted rules are processed by Oracle Data Integrator.

- To submit the metadata rule:
- 1 Select the Workflow tab, and then under Metadata, select Metadata Rule.

- 2 In **Metadata**, enter the **Location Name** or click $\stackrel{\P}{\longrightarrow}$ to select the location.
- 3 Select the metadata rule.
- 4 Click Execute, and then, click OK.

Checking the Metadata Rule Status

After you run a metadata rule, you can check the status on the Process Details page. You can click the Status icon on the Metadata page to link to the Process Details page and view the process details. See "Viewing Process Details" on page 180.

Tip: You can also check the status of the rule in Oracle Data Integrator.

Deleting Metadata Rules

You can delete dimension mappings or hierarchies in the mapping details for metadata rules created in FDMEE.

- To delete dimension mappings or hierarchies in a metadata rule:
- 1 Select the Workflow tab, and then under Metadata, select Metadata Rule.
- 2 In **Metadata**, enter the **Location Name** or click 🦠 to select the location.
- 3 Select the row in the Dimension Mappings or Hierarchy Region area.
- 4 Click Delete.

3

Loading Data

In This Chapter

Creating Member Mappings	126
Defining Data Load Rules to Extract Data	143
Managing Data Load Rules	162
Using the Data Load Workbench	168
Viewing Process Details	180
Using Excel Trial Balance Files to Import Data	182
Using Journal Templates to Import Data	184
Processing Journal Files	189

You define data load rules to extract data from your ERP source system. You begin by creating member mappings.

Creating Member Mappings

Subtopics

- Creating Mappings Using the Explicit Method
- Creating Mappings Using the Between Method
- Creating Mappings Using the In Method
- Creating Mappings Using the Like Method
- Using Special Characters in the Source Value Expression for Like Mappings
- Creating Mappings Using Multi-Dimensions
- Automap Wildcarding
- Using Special Characters in Multi-Dimensional Mapping
- Using Special Characters in the Target Value Expression
- Format Mask Mapping for Target Values
- Ignoring Member Mappings
- Importing Member Mappings
- Downloading an Excel Template (Mapping Template)
- Importing Excel Mappings
- Exporting Member Mappings
- Deleting Member Mappings
- Restoring Member Mappings

Member mappings are used to derive the target members for each dimension based on source value. Member mappings are referenced during the data load, enabling FDMEE to determine how to dimensionalize the data that is loaded to the target application. They define relationships between source dimension members and target dimension members within a single dimension. You must create a member mapping for each target dimension.

There are five types of member mappings:

- Explicit—The source value is matched exactly and replaced with the target value.
- Between—The range of source values is replaced with one target value.
- In—Enables a list of nonsequential source accounts to be mapped to one target account.
- Multi-Dimension—Target value is assigned for a combination of source segment/ chartfields.
- Like—The string in the source value is matched and replaced with the target value.

The following table is an example of a member mapping, where three segment members, Cash-101, 102, and 103 map to one EPM member, Cash.

Segment / Chartfield Member	EPM Member
Cash-101	Cash
Cash-102	Cash
Cash-103	Cash

Segment / Chartfield Member	EPM Member
Expense-1	Expense
Expense-2	Expense

You can use special characters for the source and target values. See "Using Special Characters in the Source Value Expression for Like Mappings" on page 131 and "Using Special Characters in the Target Value Expression" on page 135.

- To define member mappings:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
- 2 In Data Load Mapping, select the Location.
- 3 Select the Dimension.
- 4 Choose the type of mapping by selecting either the Explicit tab, Between tab, Multi-Dimension, or Like tab.

Note: If you have defined a metadata mapping for the dimension, FDMEE automatically creates a "Like" member mapping. If you entered a member prefix, the same member prefix is automatically entered as the target value in the member mapping. "DEFAULT" displays in the rule name and description field for system-generated mappings. When data is extracted, user-defined mappings are extracted first, and then system generated mappings.

Type options:

- Explicit—The source value is matched exactly and replaced with the target value. For example, the source value, "ABC," is replaced with the target value, "123." See "Creating Mappings Using the Explicit Method" on page 128.
- Between—The range of source values are replaced with one target value. For example, a range from "001" to "010" is replaced as one value: "999." See "Creating Mappings Using the Between Method" on page 129.
- In—In mappings enable a list of nonsequential source accounts to be mapped to one target account. In this case, multiple accounts are mapped to one account within one rule, eliminating the need to create multiple rules (as is required for an Explicit map). For example, you could have source accounts 1503, 1510, and 1515 map to the target account 15000010.
- Multi-Dimension—For the specified combination of multiple segment / chartfield values a target value is assigned See "Creating Mappings Using Multi-Dimensions" on page 133.

For example in the case where Source value Entity-001,002 Department-ABC, XYZ Account-1222, 1333, for this source value combination, the target value assigned for Account Dimension is 1200.

• Like—The string in the source value is matched and replaced with the target value. For example, the source value, "Department" is replaced with the target value, "Cost CenterA." See "Creating Mappings Using the Like Method" on page 130.

When processing the source values for transformations, multiple mappings may be applicable to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimension, and Like. Within Between and Like types, mappings can overlap.

The rule name determines precedence within a mapping type. Rules are processed in alphabetical order of the rule name within a mapping type. Numbers may also be used to help with ordering. For example it is a good practice when using numbers to number by 10's or 100's to easily insert new rules between existing rules. For example, when rules are numbered 10,20,30, you may need to add a rule between 20 and 30 and may do so by creating a new rule that starts with 25, without need to rename the other rules.

Tip: You can click **Refresh Values** to refresh the list of segment or chartfield values that appear in the drop-down list from the source system. This is especially helpful when creating "Explicit" mappings.

Creating Mappings Using the Explicit Method

Explicit mappings enable you to enter a source value to be matched exactly and replaced with a target value. Use an explicit mapping to explicitly map the members from the source ledger to an EPM target application. For example, you can map Account1 to the Account100 in your EPM target application. This allows you to explicitly define how to dimensionalize the data file that is loaded into the target application.

- To create an Explicit mapping:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
 - The Data Load Mapping screen is displayed.
- 2 In the **Dimensions** drop-down, select the dimension name.
- 3 Select the **Explicit** tab.
- 4 Click Add.
- 5 Enter the **Source Value** or click 🥄 to select a value.

See "Using Special Characters in the Source Value Expression for Like Mappings" on page 131.

- 6 **Optional**: Enter a description for the mapping.
- 7 Enter the **Target Value** or click 🦠 to select a member.

The target value is the EPM dimension member name. See "Using Special Characters in the Target Value Expression" on page 135.

8 To reverse the sign of the target account specified, select Change Sign.

- 9 In **Description**, specify a description of the mapping.
- 10 Select Apply to Rule to apply the mapping only to the specific data rule in the location.

For other data rules in the location the mapping are not applied.

By default, mappings specified at a location are applicable to all data rules in a location.

11 Click Save.

Creating Mappings Using the Between Method

Between mappings enable you to enter a range of source values, separated with a comma. The range of source values are replaced with one target value. Use a between mapping to consolidate several accounts from your ledger to a single account in the plan.

- To create a Between mapping:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.

The Data Load Mapping screen is displayed.

- 2 In the **Dimensions** drop-down, select the dimension name.
- 3 Select the Between tab.
- 4 Click Add.
- 5 Enter source values in the **Source Value** range.

The source value is the ERP segment value. Separate source values with a comma. For example 100, 199. "Between" mappings do not support special characters, such as an asterisk.

6 Enter the **Target Value** or click to select a member.

The target value is the dimension member name. See "Using Special Characters in the Target Value Expression" on page 135.

- 7 To reverse the sign of the target account specified, select **Change Sign**.
- 8 Enter the Rule Name.
- 9 In **Description**, enter a description for the mapping.
- 10 Select Apply to Rule to apply the mapping only to the specific data rule in the location.

For other data rules in the location the mapping are not applied.

By the default, mappings specified at a location are applicable to all data rules in a location.

11 Click Save.

Creating Mappings Using the In Method

In mappings enable a list of nonsequential source accounts to be mapped to one target account. In this case, multiple accounts are mapped to one account within one rule, eliminating the need to create multiple rules (as is required for an Explicit map).

- To create an In mapping:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
- 2 In Data Load Mapping, click Add.

A blank row is added.

- 3 In the Dimensions drop-down, select the dimension name.
- 4 Enter source values in the **Source Value** range.

The source value is the ERP segment value. In mappings enable you to specify nonsequential source values. Separate source values with a comma. For example 100, 199.

- 5 Enter the Target Value or click to select a member.
- 6 To reverse the sign of the target account specified, select Change Sign.
- 7 Enter the Rule Name.
- 8 Enter a description of the In mapping in the **Description**.
- 9 Select Apply to Rule to apply the mapping only to a specific data rule in the location.

For other data rules in the location the mapping are not applied.

By the default, mappings specified at a location are applicable to all data rules in a location.

10 Click Save.

Creating Mappings Using the Like Method

Like mappings enable you to enter a string in the source value that are matched and replaced with the target value.

- To create a Like mapping:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
- 2 In Data Load Mapping, in the Dimensions drop-down, select the dimension name.
- 3 Select the Like tab.
- 4 Click Add.

A blank row is added.

5 Enter the Source Value string.

The source value is the ERP segment value. "Like" source write-back mappings support special characters. See "Using Special Characters in the Source Value Expression for Like Mappings" on page 131.

6 Select the Target Value or click to select a member.

The target value is the EPM dimension member name. "Like" target write-back mappings support special characters. See "Using Special Characters in the Target Value Expression" on page 135.

- 7 To reverse the sign of the target account specified, select **Change Sign**.
- 8 Enter the Rule Name.
- 9 In **Description**, enter a description of the Like.
- 10 Select Apply to Rule to apply the mapping only to a specific data rule in a location.

For other data rules in the location the mapping are not applied.

By the default, mappings specified at a location are applicable to all data rules in a location.

11 Click Save.

Using Special Characters in the Source Value Expression for Like Mappings

The Source and Target Value expressions can have one or more special characters. Special characters are supported for "Like" mappings only.

Asterisk (*)

An asterisk (*) represents the source value. The asterisk (*) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix. The wild card or strips (write back only) takes whatever is present in the source and puts it in the target column, usually adding a prefix.

Question Mark (?)

The question mark (?) strips a single character from the source value. You can use one or more question marks (?) in the expression. You can also use question marks in combination with other expressions. For example: A?? (finds members that start with A and have any two characters following and selects the members or strips off the two characters. You see these examples in the table below.)

• <1>,<2>,<3>,<4>,<5>

Processes rows that have concatenated values and extracts the corresponding segment value (identified by the segment number). Each segment is separated by an underscore character (_). Only one segment value can be extracted in a source value expression. The source member must use the "_" character as the separator.

Note: <1>, <2>, <3>, <4>, <5> can be used with a question mark (?) but cannot be used with an asterisk (*).

<BLANK>

Processes only rows that contain the blank character (space).

Note: <BLANK> cannot be used with other source expression or characters.

This is true for both single and concatenated segment or chartfield dimension mappings.

 Table 22
 Examples of Expressions Using Special Characters

Special Character(s) Used	Mapping Type	Source Value	Target Value	Result	Notes
*	Data Load	*	1000	1000 returns 1000 WXYZ returns 1000	In this example, FDMEE processes all rows and overrides the source value with a default value of 1000. In this expression, WXYZ also returns 1000. Because you entered an asterisk for the source value it replaces any source value with the target value of 1000.
*	Data Load working off Stripping	*	*	1000 returns 1000 WXYZ returns WXYZ	In this example, FDMEE process all rows and replaces the source value as is.
*	Stripping	*	A*	101 returns A101	Processes all source members, and adds an "A" as a prefix.
*	Stripping	*_DUP	*	1000_DUP returns 1000	Processes and strips off only source values ending with "_DUP."
?	Stripping	?*	*	A1000 returns 1000 B2000 returns 2000	This result processes only source values of one or more characters in length. Strips off the first character
?	Stripping	*????	*	1000_DUP returns 1000 A1000 returns A	This result processes only source values of four or more characters in length. This strips off the last 4 characters
<1>, <2>, <3>, <4>, <5>	Data Load	<1>	*	01_420 returns 01	
<1>, <2>, <3>, <4>, <5>	Data Load	<2>	*	01_420 returns 420	
<1>,<2>,<3>,<4>,<5>	Data Load	<3>	*	01_420_AB_CC1_001 returns AB	
<1>,<2>,<3>,<4>,<5>	Stripping	?<1>	*	A01_420 returns 01	
<blank></blank>	Data Load	<blank></blank>	[None]	' ' returns [None]	Single quotes are shown for illustration only.

Creating Mappings Using Multi-Dimensions

Multi-dimension mapping enables you to define member mapping based on multiple source column values. This functionality provides you with the ability to load data into dimensions not available in the target application. For example, the mapping for Account dimension can be based on source values of Entity, Product, and Project. This allows for the derivation of target values based on the combination of source values. In addition, Lookup dimensions are added to Target Application registration. These dimensions contain source dimension but do not exist in the target application. They provide even more flexibility in creating multi-dimension filters. It is a way to facilitate conditional data loading,

- To create mappings using multiple dimensions:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
- 2 In the Data Load Mapping, in the Dimensions drop-down, select the dimension name.
- 3 Select the Multi-Dimension tab.
- 4 In Multi Dimension, click Edit.
- 5 In **Rule Name**, specify the name of the rule.
- 6 Enter the **Target Value** or click hto select a target value.
- 7 To reverse the sign of the source account value, select **Change Sign**.
- 8 In **Description**, enter a description of the mapping.
- 9 Click Add to create blank rows for specifying mapping conditions.
- 10 In **Dimension**, select the dimension to add.

The Dimension drop-down shows all dimensions mapped in the Import Format.

As a context, the source Segment/Chartfield column is shown in addition to the Data Table Column.

11 In Condition, select the method for mapping values.

Available conditions:

- Explicit
- Between
- Like
- In
- 12 In Value, specify the dimension member name.
- 13 Repeat steps 9-12 to specify multiple conditions.
- 14 Select Apply to Rule to apply the mapping only to a specific data rule in the location.

For other data rules in the location the mapping are not applied.

By the default, mappings specified at a location are applicable to all data rules in a location.

15 Click Save.

Automap Wildcarding

FDMEE enables target-account or target-entity derivation by permitting the placement of wildcard characters (* and ?) in both source and target members. Mapping-table records that have wildcard characters in both the source and target column are considered to be Automap.

FDMEE does not validate the target value.

Example Automap

Rule Name	Rule Description	Source Value	Target Account
w0011	Cash Accts	0011??	Cash.??

Example General Ledger Trial Balance Records

GL Account	Center	Description	Amount
001100	0160000	Cash In Bank	1000.00
001101	0000000	Cash Corp LB	2000.00
001116	0001000	Petty Cash	1000.00
223500	0160000	AP	5000.00

Resulting Record Conversion

GL Account	Hyperion Account
001100 0160000	Cash.00
001101 0000000	Cash.01
001116 0160000	Cash.16

Explanation of the Conversion Process

The criteria for the Automap entry (Like 0011??) retrieves the first three records from the general ledger trial balance. Because the Automap entry contains wildcard characters in the target account column, FDMEE must search the source account to replace the wildcard characters within the target account with actual characters from the source account.

The source-account characters represented by ?? marks (under Source Value) are used to replace the two question marks that follow Cash (under Target Account).

Conditional Mapping

With conditional mapping, source members are mapped to script expressions, rather than to hard-coded target members. Conditional mapping is valid only for rule-based mapping (Between, In, and Like). You can activate conditional mapping by placing #SCRIPT or #SQL in the Target value column. #SCRIPT should be used for Jython Script and #SQL for SQL Script. Conditional mapping, in conjunction with dimension processing order, enables mapping that is based on the results of dimension mappings that were processed previously, prior to the processing of the current mappings. For more information, see "Using Mapping Scripts" on page 280.

Using Special Characters in Multi-Dimensional Mapping

The Source and Target Value expressions can have the asterisk special character. An asterisk (*) represents the source value. The asterisk (*) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix. The wild card or strips (write back only) takes whatever is present in the source and puts it in the target column, usually adding a prefix.

Table 23 Using the * Special Character in Like and Multi-Dimensional Mapping Types

Туре	Source Value	Map Source	Map Target	Result
Like	E1000	E*	*	1000
Like	E1000	E*	F*	F1000
MD	E1000	E*	*	E1000
MD	E1000	E*	F*	FE1000

Using Special Characters in the Target Value Expression

You can use one special character in the target expression: an asterisk (*). You can prefix or suffix any number of characters to the asterisk (*) character. When you run the rule, the asterisk (*) character is replaced by the resulting source value (which may or may not have its own source expression), and is concatenated to any prefix or suffix you may have specified in the target expression. For example:

Target Value:
A*
Result:
1000 = A1000
Target Value:
*_DUP
Result:
1000 = 1000 DUP

Note: <BLANK> is supported in the target value expression in data mappings and can be used in all mapping types (Like, Between, and Explicit).

Format Mask Mapping for Target Values

FDMEE supports the ability to specify a format mask for a target member. The format mask defines the target member based on a combination of the source member details, and optional user defined text. This functionality is useful when designating the target member based on some part of the source member, an additional prefix, suffix, or replacement text for the target.

The format mask is available for the target member specification for all mapping types except "Explicit." Common usage of this mapping type falls into three categories – replacing segments from the source, replacing segments with string operations, and replacing segments with string operations using a prefix or a suffix.

#FORMAT Mapping Type Components

The #FORMAT mapping type consists of the following components:

Table 24 #FORMAT Mapping Type Components

Component	Description Indicates that a mapping type of FORMAT is specified in the target member.	
#FORMAT		
<format mask></format 	 User defined format mask with the following characters used to define the format: "?"—Include a character from a specific position in the source member or segment within a member. "#"—Skip or drop a character from the source when creating the target member. The "#" and "?" character used interchangeably. "character"—Include the user defined character on the target "as- is". Used for prefixing, suffixing or any fix or required character. This can be used in conjunction with the special format mask characters. 	
	 "*"—Include all characters from the source segment or source. When "*" is used as the only format mask character in a segment, then the entire segment value is copied from the source. When "*" is used in conjunction with "#" or the "?" character, then all remaining and unused characters are brought over. "*" is a wildcard character that takes the remaining characters not specified by "?" or "#". For example, when the source is "abcd" and "*" is used, then the target is "abcd". When the target is "?#*", then the result is "acd". If FDMEE encounters a "*" within a segment, then anything specified after the "*" is ignored other than the "character" specified on the format. 	
<segment delimiter></segment 	The optional segment delimiter defines the character that is used to delimit the segments in the source and target member. For this rule type, the source and target delimiter must be the same. When the segment delimiter is not specified, then the format mask is applied to the entire member independent of any segment specification or delimiter.	

#FORMAT Mapping Example

The following is an example that uses all options provided by #FORMAT:

Table 25 #FORMAT Mapping Example

Source	Target	Result
12345-6789-012-3456ABC-001	#FORMAT("???-*-GROUP-AA##?#*X-GROUP","-") Explanation: Take the first three characters of the first segment, take the entire second	123-6789- GROUP-
	segment, replace the third segment with the text "GROUP", prefix the fourth segment with AA, drop the third and fourth characters, keep the fifth character, drop the sixth character, keep ABC and add suffix "X, replace the fifth segment with the text "GROUP".	AA5ABCX- GROUP

Replacing Segments

You may want to use the format of the source member as the definition of the target member, but replace some of the source values rather than reuse the segments from the source. For example, you may have a requirement to filter the source by the value of the 4th segment, then replace the 7th segment with an explicit value, and then retain the values of the other segments as in the following:

Source:

Target:

```
??????-??????-?-012000000-??????-???-GROUP-???????-??????-??????
```

Replacing Segments with String Operations

You may also need to perform a string operation on a segment that is being replaced. For example, you may have a segment 11002293, but when the segments are written, you want to only take the last four digits, or the first six digits. Examples using the member 11002293 are:

- Ignore the first two characters and provide the result: 002293. Use #FORMAT("##*").
- Truncate the last three characters provide the result: 11002. Use #FORMAT("?????").
- Ignore the first two and truncate the last three with the result: 002. Use #FORMAT("##???").

Replace Segment with String Operations and Using an Prefix or Suffix

You may want to use the segment from the source "as-is" in the corresponding segment in the target. For example, the source may be A100, and you can map this segment in the target. In this case, you use a wildcard on the source, and then specify the explicit value for that value in the target based on the source.

Note: If any other string operation is desired, then use scripting.

Replacing with Segment Value from Source

You may want to use the segment value from the source "as is" in the corresponding segment in the target. For example, the segment value for the fifth segment from the source my be A100, and you can map this value as the fifth segment value in the target. In this case, you use a wildcard on the source, and then specify the explicit value for that segment in the target based on the source.

Using the #FORMAT Mapping Type

- ➤ To use the #FORMAT mapping type:
- 1 Select Workflow, and then under Data Load, select Data Load Mapping.
- 2 In the **Data Load Mapping**, in the **Dimensions**, drop-down, select the dimension name.
- 3 Select either the **Between**, **In** or **Like** tab.
- 4 Click Add.
- 5 In **Source Value**, enter the segment to map.

For example, enter: 12345-6789-012-3456ABC-001

In the **Target Value**, select the format map for the target member using this format: #FORMAT(<format mask>, <segment delimiter>).

For example, enter #FORMAT("???-*-GROUP-AA##?#*X-GROUP","-")

For information on the format map components, see "#FORMAT Mapping Type Components" on page 136.

- 7 To reverse the sign of the target account specified, select **Change Sign**.
- 8 Enter the Rule Name.
- 9 In **Description**, enter a description of the mapping.
- 10 Select **Apply to Rule** to apply the mapping only to a specific data rule in a location.

For other data rules in the location the mapping are not applied.

By the default, mappings specified at a location are applicable to all data rules in a location.

11 Click Save.

The result of applying format map created in steps 5 and 6 is: 123-6789-GROUP-AA5ABCX-GROUP

Ignoring Member Mappings

You can ignore loading data to a particular dimension member.

- To ignore member mappings:
- 1 From the task bar, select **Data Load Mapping**.

The Data Load Mapping screen is displayed.

2 Select a source value, and in its **Target Value**, enter **IGNORE**.

Importing Member Mappings

You can import member mappings from a selected .CSV and .TXT. You can create new mappings in a text file and import them. Import member mappings support merge or replace modes, along with validate or no validate options for target members.

- To import member mappings:
- 1 From the Data Load Mapping task bar, select Import.

The Import drop-down is displayed.

2 From the **Import** drop-down, select either **Current Dimension** or **All Dimensions**.

The Select file to import screen is displayed.

- 3 Navigate to the file to import and click OK.
- 4 Optional: If necessary, click Upload to navigate to the file to import and then, click OK.

In the member mapping import files FDMEE supports one of the following characters as column separator:

- •
- •
- •

The order of the columns are as follows: Source Value, Target Value, Rule Name, Rule Description

Note: If you add a minus sign in front of a target account value, then it is imported with the "Change Sign" selected.

The mapping details are as follows:

Table 26 Supported Column Separators

Column	Mapping
100, Cash, R1, Explicit Mapping	Explicit Mapping
100>199, Cash, R2, Between Mapping '	">" indicates its BETWEEN mapping.
1*, Cash, R3, Like Mapping	"*" indicates its LIKE mapping.

Column	Mapping
#MULTIDIM ACCOUNT=[4*] AND UD3=[000],Cash,R4,Multi Dimension Mapping	"#MULTIDIM" indicates it is a multiple dimension mapping. The actual column name used for the mapping is the Data Table Column Name. The easiest way to create a multiple dimension mapping is to create a mapping through the user interface and export it to the file. You can then modify the file by additional mapping.
10, 20, In Mapping	Source values are enclosed with "" and separated by a comma (,) for the In mapping. For example IN 10, 20 is defined as "10,20" in the source column of the import file.

Downloading an Excel Template (Mapping Template)

In the Data Load Mapping, using the import feature, you can select and import an Excel mapping, and specify whether to merge or replace the mappings. Excel map templates with correct formatting are included in the EPM_ORACLE_HOME/products/FinancialDataQuality/templates directory.

The mapping template also includes a macro script that pulls Financial Management dimensions directly from the target application you are connected.

When working with a mapping template in Excel:

- Do not have any blank lines in the map template.
- Do not insert lines in the map template.
- One mapping template is allowed per workbook.
- To download an Excel template:
- 1 From the Workflow tab, select Data Load Mapping.
- 2 Select the All Mapping tab.
- 3 From the Import drop-down, select Download Excel Template.

A Maploader.xls file is downloaded. Copy or save the file to your hard drive.

- 4 Open the Maploader.xls file.
- 5 Select the **Map** tab.
- 6 Enter the FDMEE Location name in cell B1, and the Location ID in cell B2.
- 7 Complete the following column fields:
 - a. In **Source**, enter the source dimension value.

You can specify wildcards and ranges when entering the source dimension.

- Wildcards for unlimited characters—Use asterisks (*) to denoted unlimited characters. For example, enter 548* or *87.8.
- Wildcards for single character place holders—Use questions marks (?) to denote single character place holders. For example,
 - o 548??98

- o ??82???
- o ??81*
- **Range**—Use commas (,) to denote ranges (no wildcard characters are allowed). For example, specify a range as 10000,19999.

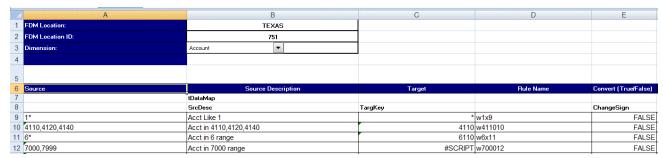
(this would consider all values from 10000 to 19999 inclusive of both start and end values)

In this case, FDMEE considers all values from 10000 to 19999 to include for both start and end values.

- **In map**—Use commas (,) to separate entries (no wildcard are characters allowed). You must have at least three entries or the map shows as a between map. For example, specify an "In" map as 10,20,30.
- **Multi-Dimension map**—Use #MULTIDIM to indicate its multi-dimension mapping. Enter the DIMENSION NAME=[VALUE] and the value. The Value follows the logic as Wildcard, Range, and In map. In the following example the search criteria is all ACCOUNT starting with 77 and UD1 = 240. For example, #MULTIDIM ACCOUNT=[77*] AND UD1=[240].

3

- b. In **Source Description**, enter a description of the source value.
- c. In Target, enter the target dimension value.
- d. In **Change Sign**, enter **True** to flip the sign of the Account dimension. Enter **False** to not flip the sign of the Account dimension. This setting is only used when mapping the Account dimension.
- e. In **Data Rule Name**, enter the data rule name when the mapping is applicable for a specific data rule name.

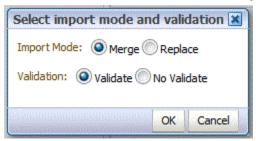


Importing Excel Mappings

You can import Excel mappings by selecting the Import option and selecting an Excel mapping.

Note: The import of mapping rules using an Excel template does provides a place to specify a mapping script.

- To import an Excel mapping:
- 1 From the Workflow tab, select Data Load Mapping.
- 2 Select the All Mapping tab.
- 3 From the **Import** drop-down, select **Import from Excel**.
- 4 From Select a file to import, select the Excel file to import and click OK.
- 5 From Select mode and validation, and then Import mode, select the import mode



- Merge—Overwrites the data in the application with the data in the Excel data load file.
- Replace—Clears values from dimensions in the Excel data load file and replaces them with values in the existing file.
- 6 Click Validate to validate the mappings.
- 7 Click OK.

The mapping inherits the default data load rule, and the description of "System Generated Mappings."

Exporting Member Mappings

You can export member mappings to a selected file, or an Excel file.

- To export member mappings:
- 1 From the **Data Load Mapping** task bar, select **Export**.
- 2 From the **Export** drop-down, select the export method:

Export options:

- Current Dimension
- All Dimensions
- Export to Excel
- From the Specify file location for Current Dimension and All Dimensions export methods, specify the file name in the File Name field, or navigate to the file to export and click OK.

For the **Export to Excel** method, mappings are exported to a Microsoft Excel spreadsheet. Open or save the XSL file as desired.

When you export to Excel, you cannot re-import in that format.

When the file has been exported, FDMEE displays the message: "File exported successfully".

4 Optional: Click Upload or Download and navigate to the file to export and then click OK.

Deleting Member Mappings

You can delete either all member mappings or only those mappings for which there is a tab in which mappings have been added

- To delete member mapping from a selected tab:
- 1 From **Data Load Mapping**, select the tab from which to delete mappings.

For example, select the **Explicit** tab to view explicit type mappings.

To view all mappings, select the **All Mappings** tab.

2 Select the mapping and click Delete Mappings.

To delete multiple mappings, use the **Shift** key to select multiple mappings.

To delete all mappings, use CtI + A key.

- 3 In Are you sure you want to delete the selected data load mapping(s), click OK.
- 4 Click Save.

Restoring Member Mappings

Restoring member mappings deletes any mappings made in the current session and restores mappings based on the point of view.

You can only restore member mappings from the last data load for the POV.

- To restore a member mapping:
- 1 From the Data Load Mapping task bar, select Restore Mapping.
- 2 In Restore Mapping Confirmation, click OK.

Defining Data Load Rules to Extract Data

Subtopics

- Defining Data Load Rule Details
- Defining Data Load Rule Details for a file-based Source System
- Using a Multiple Period Data Load with a Data Load Rule

After you define member mappings for the data load rule, define data load rules for ledgers or business units in your source system. Data load rules are defined for locations that you have set up. Data load rules are specific to:

locations

- Ledgers for Fusion and E-Business Suite source systems
- Business units for PeopleSoft Enterprise Financial Management source systems

You can create multiple data load rules for a target application so that you can import data from multiple sources into a target application. Use the following high level process to create a data load rule:

- 1. Create the data load rule.
- 2. Define data load rule details. See "Defining Data Load Rule Details" on page 144.
- 3. Execute the data load rule. See "Running Data Load Rules" on page 163.

You can create multiple data load rules for a target application so that you can import data from multiple sources into a target application. Use the following high level process to create a data load rule:

- 1. Create the data load rule.
- 2. Define data load rule details. See "Defining Data Load Rule Details for a file-based Source System" on page 161.
- 3. Execute the data load rule. See "Running Data Load Rules" on page 163.

Defining Data Load Rule Details

You create and modify data load rules on the Data Load screen. The Data Load Rule screen has three sections:

- Data Rule Summary
- Data Load Details
- Source Filters

Note: Before you create data load rules, ensure that your source system data does not include special characters, which are not supported in Financial Management target applications.

- To define the data load details for a source system:
- 1 From Workflow, and then under Data Load, select Data Load Rule.
- In Data Load Rule, and then from the Data Load summary task bar, click Add.
- 3 In Name, enter the data load rule name.
- 4 Select a Category.

The categories listed are those that you created in the FDMEE setup. See "Defining Category Mappings" on page 106.

By default only rules that belong to the category selected in POV are displayed. To see all rules, select **Show** and then **All Categories**.

5 Optional: Enter a description.

6 In Accounting Entity: If the Location has no accounting value, enter the accounting entity.

If the Location was created with an accounting entity, this field is already populated.

7 In the *Include Adjustment Periods drop-down select either yes or no from the drop-down.

Adjustment periods ensure that the FDMEE adjustment periods map correctly to the source system adjustment periods.

This option is only available for EBS, Peoplesoft, and Fusion.

8 In the *Period Mapping Type drop-down, select the period mapping type for each data rule.

Valid options are:

- Default—The Data Rule uses the Period Key and Prior Period Key defined in FDMEE to determine the Source General Ledger Periods mapped to each FDMEE period included in a Data Rule execution.
- Explicit—The Data Rule uses the Explicit period mappings defined in FDMEE to
 determine the Source GL Periods mapped to each FDMEE Period included in a Data
 Rule execution. Explicit period mappings allow support of additional GL data sources
 where periods are not defined by start and end dates.
- None—With source Adaptors use this option to ignore source period mappings. Thus, all imported data rows are mapped to the FDMEE period selected in a Data Rule execution.

These options are not available for a file-based definition.

- 9 Select a Plan Type.
- 10 In the *Include Adjustment Periods drop-down select either yes or no from the drop-down.

Adjustment periods ensure that the FDMEE adjustment periods map correctly to the source system adjustment periods.

When you explicitly map period 13 to December/Period 12, and select the Include Adjustment Period option, then the following occurs:

- For YTD balances, then period 13 becomes the ending balance.
- For PTD balances, then period 13 and December/Period12, are added.
- 11 In *Period Mapping Type drop-down, select the period mapping type for each data rule.

Valid options are:

- Default—The Data Rule uses the Period Key and Prior Period Key defined in FDMEE to determine the Source General Ledger Periods mapped to each FDMEE period included in a Data Rule execution.
- Explicit—The Data Rule uses the Explicit period mappings defined inFDMEE to determine the Source GL Periods mapped to each FDMEE Period included in a Data Rule execution. Explicit period mappings allow support of additional GL data sources where periods are not defined by start and end dates.
- None—Source period mappings are ignored, thus all imported data rows are mapped to the FDMEE period selected in a Data Rule execution.

The following table outlines the available Period Mapping Types depending on the Location Type and the Import Format Type.

Table 27 Period Mapping Types

Location Type	Import Format Type	Period Mapping Default	Explicit Period Mapping Explicit	Period Mapping	Include Adjustment Periods
With Accounting Entity	Standard	Yes	Yes	N/A	Yes
Without Accounting Entity	Standard	Yes	Yes	N/A	Yes
With Accounting Entity	Source Adapter	N/A	N/A	Yes	Explicit: Yes None: N/A
Without Accounting Entity	Source Adapter	N/A	N/A	Yes	Explicit: Yes None: N/A

12 In Accounting Entity, select the accounting entity from the list of values of the source system.

You can specify the accounting entity on this screen, or when entering location detail. Data Rules in Locations without an Accounting Entity require you to select an Accounting Entity. This functionality is only applicable to Data Rules in a Location using a Standard Import Format.

The Accounting Entity cannot be modified once the Data Rule has been executed.

13 To use the PeopleSoft's Commitment Control feature, complete the following:

• *Period Mapping Type Select Budget—If commitment Ledger Group is the source, then the period mapping type must be Budget.

Note: From PeopleSoft Commitment Control, only Budget Period Data can be extracted. The source calendar / period are determined based on control budget definition in PeopleSoft.

- **As of Date**—Select the effective date of records from the source system.
- **Target for Blank Period**—Select the target period to use when the source period is blank.

14 Click Save.

Source filter options by source systems:

- For Fusion and E-Business Suite source systems, see "Defining Source Filter Options for E-Business Suite Source Systems" on page 147.
- For SAP source systems, see "Defining Source Filter Options for SAP Adapters" on page 152.
- For PeopleSoft Enterprise Financial Management source systems, see "Defining Source Filter Options for PeopleSoft Enterprise Financial Management Source Systems" on page 150.

• For PeopleSoft human resource source systems, you create human resource data load rules. See "Defining Human Resource Data Load Rules" on page 240.

Defining Source Filter Options for E-Business Suite Source Systems

When defining data load mapping details, you can define the data to extract, including whether or not to extract:

- The amount type—Only monetary, statistical, or both, monetary and statistical amounts
- Zero balance accounts where the debits and credits for an account total zero and there is not any period activity.
- Adjustment periods—Determines whether to extract balances in the adjustment period
- Standard or Average balances—Average balances only contain balance sheet data.
- Source balance type—Actual, Budget, or Encumbrance

In FDMEE you classify the data to transfer with the valid types in the source accounting entity of Actual, Budget, and Encumbrance. Typically, you do not map a segment from the chart of accounts to the Scenario dimension, so you choose a default member as part of the data rule definition. When the data is extracted, it is defined with the Scenario column defined in accordance with the default selected.

You can extract functional balances, which are stored in the base currency of the selected ledger or business unit. For example, when transactions are entered in multiple currencies, the total of all transaction balances is expressed in the functional currency.

You can also extract entered balances, which are balances associated with a currency attached to a primary ledger other than functional/local currency.

Additionally FDMEE can import the data in a currency specified by the user. In this case, the balances must be translated to the specified currency in the source system. This can be achieved by running the Currency Translation process in the ERP system. (FDMEE does not perform any currency translations.)

To define the data load source filter for Fusion and E-Business Suite source systems:

1 Select the Amount Type:

- Monetary
- Statistical—The balance selection of entered or functional currency does not apply.
- Monetary and Statistical
- 2 From **Currency Type**, select the currency type by which to extract balances:
- 3 In the Include Zero Balance drop-down, select whether to include zero balances.

Select Yes to extract zero balances.

If you select *No*, accounts with zero balances are not extracted. In the following circumstances, note how FDMEE determines when to exclude zero balances:

- Balance Sheet account type—FDMEE excludes the balance when the beginning debit, beginning credit, period debit and period credit are all zero.
- Income Statement account type—FDMEE excludes the balance when the period debit and period credit are all zero.
- If you select the balance type as YTD in the data rule, then the method for excluding zero balances follows the Balance Sheet account type.

Note that balances are excluded only when there is no activity for that period. For example, when the Beginning Balance debit/credit are zero in addition to the period debit/credit.

In the case when there is activity and balances are zero, amounts are extracted. If there is some activity that results in a zero balance, for example a debit of 5 and a credit of 5, then the zero amount is extracted.

Note: The Include Zero Balance option is not applicable when pulling data from Peoplesoft.

In **Signage Method**, select the method for flipping the sign of amounts when data is loaded.

Available methods:

• Absolute—Loads the data based on the following rules:

Table 28 Absolute Signage Rule	les
--------------------------------	-----

Account Type	GL (GAAP)	EPM (Absolute)
Revenue	naturally negative	signage flipped
Liability	naturally negative	signage flipped
Equity	naturally negative	signage flipped
Expense	naturally positive	signage unchanged
Asset	naturally positive	signage unchanged

- Same as source—Loads the same sign as recorded in the source system.
- Reverse from source—Loads the reverse of the sign as recorded in the source system.
- 5 Select whether to include adjustment periods.
- 6 Select the Amount for Balance Sheet Accounts and Amount for Income Statement Accounts:
 - YTD—Year-to-date account balance, where account balances are accumulated from the beginning of the year to the current period. Typically, balance sheet accounts (assets, liabilities, and equities) are specified with a YTD balance. (The default is YTD.)
 - Periodic—Account balance for the specific period. Typically, income statement accounts (revenues and expenses) are specified with a periodic balance. (The default is PTD.)
- 7 From **Currency Type**, select the currency type by which to extract balances:
 - Functional—Balances stored in the base currency of the selected ledger or business unit (local currency)

- Entered—Balances associated with a currency attached to primary ledger other than the functional/local currency that you specify in the Currency Code field.
- Translated—FDMEE can import the data in a currency specified by the user. In this
 case, the balances must be translated to the specified currency in the source system. This
 can be achieved by running the Currency Translation process in the ERP system.
 (FDMEE does not perform any currency translations.) Additionally, you need to specify
 the translated currency code in the Currency Code field
- From **Currency Code** (Entered and Translated currency types only), select the ISO 4217 currency code to use with either an entered or translated currency type.

For example, enter **EUR** to select the EURO currency code.

9 Select the Balance Method:

- Standard—In Oracle General Ledger, accounting transaction balances are stored as-is, also known as standard balances.
- Average—Average balances only contain balance sheet data. If you selected Statistical as the amount type, the Balance Method is ignored.

10 Select the balance type to extract:

- Actual
- **Budget**—If you select the Budget source balance type, click Add to select budget types to include in the extraction.
- Encumbrance—If you select the Encumbrance source balance type, click *Add* to select encumbrance types to include in the extraction.

11 Select the segment values to extract.

If you do not want to extract all data from the source general ledger, filter data by the balancing segments of the source. Options include:

- All
- Selected

In Oracle E-Business Suite, the balancing segment ensures that at this level, balancing debits equal credits. When you create a data load rule, you can extract the general ledger balances relating to all the members of the balancing segment or for specific members of it.

To select the balancing segment values, click , and then select segment values and click *OK*.

To deselect a value, click , and then in the Select Balancing Segment Values dialog box, clear any values and click *OK*.

12 Select the Beginning, Ending, and Average exchange rate options.

ERP source systems maintain comprehensive exchange rate information for transaction processing. This information can be used by the target applications by extracting the exchange rates. You can select a beginning, ending, and average rate type from the source

system. (The types in the source system may not explicitly define those rates types but are mapped to the rates types in the FDMEE interface table.)

Note: For Planning applications, exchange rates are loaded only when the "Classic" data load method is selected.

Note: You define exchange rate options only when the target application is multi-currency.

Note: For Account Reconciliation Manager applications, you must not choose more than one rate option.

13 Click Save.

14 Define the target filter options.

After you define the target filter options, run the data rule. See "Running Data Load Rules" on page 163.

Defining Source Filter Options for PeopleSoft Enterprise Financial Management Source Systems

When defining data load rule details, you can specify various data extraction options as described below.

➤ To define the source filter options for PeopleSoft Enterprise Financial Management source systems:

1 Select the Amount Type:

- Monetary
- Statistical—The balance selection of entered or functional currency does not apply.
- Monetary and Statistical
- In Signage Method, select the method for flipping the sign of amounts when data is loaded.

Available methods:

• Absolute—Loads the default debit or credit sign.

Table 29 Absolute Signage Rules

Account Type	GL (GAAP)	EPM (Absolute)
Revenue	naturally negative	signage flipped
Liability	naturally negative	signage flipped
Equity	naturally negative	signage flipped
Expense	naturally positive	signage unchanged

Account Type	GL (GAAP)	EPM (Absolute)
Asset	naturally positive	signage unchanged

- Same as source—Loads the same sign as recorded in the source system.
- Reverse from source—Loads the reverse of the sign as recorded in the source system.

3 Select the Amount for Balance Sheet Accounts and Amount for Income Statement Accounts:

- YTD—Year-to-date account balance, where account balances are accumulated from the beginning of the year to the current period. Typically, balance sheet accounts (assets, liabilities, and equities) are specified with a YTD balance.
- Periodic—Account balance for the specific period. Typically, income statement accounts (revenues and expenses) are specified with a periodic balance.

4 From **Currency Type**, select the currency type by which to extract balances:

- Functional—Balances stored in the base currency of the selected ledger or business unit (local currency)
- Entered—Balances associated with a currency attached to primary ledger other than the functional/local currency that you specify in the Currency Code field.
- 5 From **Currency Code**, select the ISO 4217 currency code to use with an enteredcurrency type.

For example, enter **EUR** to select the EURO currency code.

- 6 Select the Ledger Group.
- 7 Select the Ledger.

In PeopleSoft, a business unit may have multiple ledger groups. In this list, FDMEE displays only the ledger groups associated with the ledger.

8 **Optional**: To select book code values, click 🖶, select book code values, and then, click **OK**.

To clear a book code, click . Then, in the Select Book Code dialog box, clear book codes, and then, click *OK*.

9 To select budget values, click 🕕, select budget values, and then, click **OK**.

You specify the budget values when the ledger that you selected has the data table, "Ledger_Budg."

To deselect budget values, click . Then, in the Select Budget Scenario Values dialog box, clear values, and then, click *OK*.

10 Click Save.

11 Define target filter options.

After you define target filter options, run the data rule. See "Running Data Load Rules" on page 163.

Defining Source Filter Options for JD Edwards GL Source Systems

When defining data load mapping details, define the data to extract, including whether to extract the company code, ledger type, amount type.

- To define the source filter options for a JD Edwards GL source system:
- Select the Amount Type.

Select **PTD** for Period to Date, or **YTD** for Year to Date balances.

2 Select the Company Code.

Specify the company code in four characters or less using alphanumeric characters.

3 Select the Ledger.

Specify the ledger in two characters or less using alphanumeric characters from the JD Edwards source system. For example, ledger types include:

- AA—Actual
- BA—Budget
- CA—Original Currency Transaction
- 4 Click Save.

Defining Source Filter Options for SAP Adapters

Subtopics

- SAP Integration Process Background
- Defining Source Filter Options for the SAP_FDM_GLBALANCES_CLASSICS Adapter
- Defining Source Filter Options for the SAP_FDM_GLBALANCES_NEW Adapter
- Defining Source Filter Options for the SAP_FDM_COST_CENTER Adapter
- Defining Source Filter Options for the SAP_FDM_PROFIT_CENTER Adapter
- Defining Source Filter Options for the SAP_FDM_CUSTOMER_BALANCES Adapter
- Defining Source File Options for the SAP_FDM_VENDOR_BALANCES Adapter

SAP Integration Process Background

Oracle Data Integrator creates an ABAP program to extract the data from SAP into a file and then transfers the file to a FTP Server or Shared File system. The ABAP program is generated based on the mappings in the import format and options defined in the rule. The ABAP program is created and uploaded to SAP system when you run a data load rule. In the SAP deployment, ABAP programs are created and modified in a development environment, and locked from modification in the production environment.

In order to support this requirement, ODI provides an option UPLOAD_ABAP_CODE in the SAP Load Knowledge Module. This option is set to "Yes" in a development environment, and "No" in a production environment. The ABAP programs are transferred from development to production using SAP Transport requests.

Because the ABAP code that is run in a production environment is not modified, users should create the import format in the exact format as the development environment. In addition there are options in the data rule that impact the logic of the ABAP program. These options must be set to the same value as they are set in the development environment. Changing these options in the data rule in a production environment does not have any impact on the extraction process. If changes are required to these options, make them in the development environment and the ABAP code transported to the production environment. The options that impact the ABAP code generation are noted in the SAP adapter sections.

Defining Source Filter Options for the SAP_FDM_GLBALANCES_CLASSICS Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- To define the source filter options for the SAP_FDM_GLBALANCES_CLASSICS adapter:
- 1 In Include Account Description, select one of the following:
 - Yes—include the GL Account description
 - **No**—include all other conditions

Impacts ABAP program

- 2 In Amount Type, select one of the following:
 - **PTD**—Period to Date
 - **YTD**—Year to Date

No impact on ABAP program

3 In Company Code, specify the company code in four characters or less using alphanumeric characters.

No impact on ABAP program

- 4 In Currency Type, specify one of the following types:
 - **00**—Transaction currency
 - **10**—Company code currency
 - **30**—Group currency

No Impact on ABAP program

5 In Language, specify the language code in two characters or less, using uppercase characters.

For example, specify "EN" for English.

Refer to the SAP documentation for the language code.

No impact on ABAP program

6 In Ledger, specify the ledger code in two characters or less using alphanumeric characters.

No impact on ABAP program

- 7 In **Record Type**, select one of the following record types:
 - **0**—Actual
 - **1**—Plan

Defining Source Filter Options for the SAP_FDM_GLBALANCES_NEW Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- To define the source filter options for a SAP_FDM_GLBALANCES_NEW adapter:
- 1 In Amount Type, select one of the following:
 - **PTD**—Period to Date
 - **YTD**—Year to Date

No impact on ABAP program

2 In Company Code, specify the company code in four characters or less using alphanumeric characters.

No impact on ABAP program

- 3 In Currency Type, select one of the following types:
 - **00**—Transaction currency
 - **10**—Company code currency
 - **30**—Group currency
 - **40**—Hard currency
 - **50**—Index based currency
 - **60**—Global company currency

No impact on ABAP program

- 4 In Include Account Description, select one of the following:
 - **Yes**—include the GL Account description
 - No—include all other conditions

Impacts ABAP program

5 In Language, specify the language code in two characters or less, using uppercase characters.

For example, specify "EN" for English.

Refer to the SAP documentation for the language code.

No impact on ABAP program

6 In Ledger, specify the ledger code in two characters or less using alphanumeric characters.

No impact on ABAP program

- 7 In **Record Type**, select one of the following record types:
 - **0**—Actual
 - **1**—Plan

No impact on ABAP program

Defining Source Filter Options for the SAP_FDM_COST_CENTER Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. When the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- ➤ To define the source filter options for a SAP_FDM_COST_CENTER adapter:
- 1 In Activity Type, select one of the following:
 - **Yes**—include the activity type
 - **No**—exclude the activity type

Impacts ABAP program

- 2 In Activity Type Description, select whether to include or exclude the activity type description:
 - **Yes**—include the activity type description
 - **No**—exclude the activity type description

Impacts ABAP program

- 3 In Amount Type, select one of the following:
 - **PTD**—Period to date balances
 - **YTD**—Year to date balances

No impact on ABAP program

4 In Controlling Area, specify the controlling area in four characters or less using alphanumeric characters.

No impact on ABAP program

- 5 In Include Cost Element Description, select whether to include the cost element description:
 - **Yes**—include the cost element description
 - **No**—exclude the cost element description

Impacts ABAP program

- 6 In Currency Type, select one of the following:
 - **20**—Controlling Area currency

- **00**—Transaction currency
- **70**—Cost Center currency
- (blank)—leave blankwhen the **Quantity** filter is **Yes** or the **Activity Type** filter is **Yes**.

- 7 In Flow Check, select one of the following:
 - **External**—load external balances
 - **Internal**—load internal allocations

Impacts ABAP program

8 In Language Code, specify the language code in two characters or less, using uppercase characters.

For example, specify "EN" for English.

No impact on ABAP program

9 In Ledger Code, specify the ledger code in two characters or less using alphanumeric characters.

No impact on ABAP program

- 10 In Statistical Key Figure, select to extract statistical key figure totals:
 - Yes—extract the data for Statistical key figure totals
 - **No**—extract data for activity type totals

Impacts ABAP program

11 In Group Code, select one of the following group codes:

For external balances, select:

- **0101**—data extracted is for the Cost Center Group
- **0102**—data extracted is for the Account Group

For internal balances, select:

- **0101**—data extracted is for the Cost Center Group
- **0102**—data extracted is for the Account Group
- **0104**—data extracted is for the Statistical Key Figure Group
- **0105**—data extracted is for the Account Group
- (Null)—no grouping is required

Impacts ABAP program

- 12 In Value Type, select one of the following:
 - **04**—Actual
 - **01**—Plan

Defining Source Filter Options for the SAP_FDM_PROFIT_CENTER Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- ➤ To define the source filter options for a SAP_FDM_PROFIT_CENTER adapter:
- 1 In Amount Type, select one of the following:
 - **PTD**—Period to date balances
 - **YTD**—Year to date balances

No impact on ABAP program

2 In **Controlling Area**, specify the value for the controlling area.

No impact on ABAP program.

- 3 In **Currency Type**, select one of the following:
 - **10**—Company code currency
 - **00**—Transaction currency
 - **70**—Profit Center currency
 - (blank)—Select blank when **Quantity** is set to **YES**.

No impact on ABAP program

- 4 In **Dummy Prctr**, select:
 - **Yes**—include balances related to the dummy profit center.
 - **No**—include other conditions.

No impact on ABAP program

5 In Language, select the language code in two characters or less, using uppercase characters.

For example, specify "EN" for English.

No impact on ABAP program

6 In Ledger, select the ledger code in two characters or less using alphanumeric characters.

No impact on ABAP program

- 7 In Statistical Key Figure, select to extract statistical key figure totals:
 - **Yes**—extract the data for Statistical key figure totals
 - **No**—extract data for activity type totals

Impacts ABAP program

- 8 In Group Code, select one of the following group codes:
 - **0106**—Profit Center Group
 - **0109**—Account Group
 - (blank)—no group is required

- 9 In **Record Type**, select one of the following types:
 - **0**—Actual
 - **1**—Plan

No impact on ABAP program

Defining Source Filter Options for the SAP_FDM_CUSTOMER_BALANCES Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- ➤ To define the source filter options for a SAP_FDM_CUSTOMER_BALANCES adapter:
- 1 In Amount Type, select one of the following:
 - **PTD**—Period to date balances
 - **YTD**—Year to date balances

No impact on ABAP program

2 In Company Code, specify the company code in four characters or less using alphanumeric characters.

No impact on ABAP program

- 3 In Currency Type, select one of the following:
 - **10**—Company code currency
 - **00**—Transaction currency

No impact on ABAP program

In Customer, select the customer code in ten characters when the balance is required for a specific customer.

Otherwise, set to blank.

No impact on ABAP program

- 5 In **Customer Details**, select to include customer details:
 - **Yes**—include customer details
 - **No**—exclude customer details

- 6 In Flow Check, select one of the following:
 - **Open**—load open item balances
 - **Cleared**—load cleared item balances
 - **All**—load all item balances

Impacts ABAP program

- 7 In Special GL Transactions, select one of the following:
 - **Yes**—load special General Ledger balances
 - **No**—load other balances

Impacts ABAP program

- 8 In **Trading Partner**, select to include trading partner balances:
 - **Yes**—load trading balances
 - **No**—load other conditions

No impact on ABAP program

Defining Source File Options for the SAP_FDM_VENDOR_BALANCES Adapter

Before executing a data load rule using a SAP source adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the data load rule any number of times.

- To define the source filter options for the SAP_FDM_VENDOR_BALANCES adapter:
- 1 In Amount Type, select one of the following:
 - **PTD**—Period to date balances
 - **YTD**—Year to date balances

No impact on ABAP program

2 In **Company Code**, specify the company code in four characters or less using alphanumeric characters.

No impact on ABAP program

- 3 In Currency Type, select one of the following:
 - **10**—Company code currency
 - **00**—Transaction currency

No impact on ABAP program

- 4 In Flow Check, select one of the following:
 - **Open**—load open item balances

- **Cleared**—load cleared item balances
- All—load all item balances

- 5 In Special GL Transactions, select one of the following:
 - Yes—load special General Ledger balances
 - No—load other balances

Impacts ABAP program

- 6 In Trading Partner, select to include trading partner balances:
 - **Yes**—load trading balances
 - **No**—load other conditions

No impact on ABAP program

7 In Vendor, specify the vendor code in ten characters when the balance is required for a specific vendor.

Otherwise, set to blank.

No new ABAP code is uploaded for this filter.

- 8 Specify Vendor Details, select whether to include vendor details:
 - Yes—include vendor details
 - **No**—exclude vendor details

No impact on ABAP program

Defining Source Filter Options for the Open Interface Adapter

Before executing a Data Rule using open interface adapter, you are required to have generated the Import Format's ODI Scenario. Once the ODI Scenario exists in the Execution Repository, you can execute the Data Rule any number of times.

- To define the source filter options for an open interface adapter:
- 1 In **Batch Name**, enter the name of the batch used to identify the data in the interface table.
- 2 In Record Type, specify whether to delete data after importing the data by selecting Y (for yes) or N.

To delete the data, select **Y**. To retain the data, select **N**.

3 Click Save.

Defining Data Load Rule Details for a file-based Source System

When defining data load detail for a file-based data load system, load data to a single period or a range of periods. For a single period, enter the file name in the data rule and run the rule for a single period. To load multiple periods, create a file for each period and append the period name or period key to the file name. When the rule is executed for a range of periods, the process constructs the file name for each period and uploads the appropriate data to the POV.

- To define the data load details for a file-based source system:
- 1 In Name, enter the data load rule name.
- 2 In Category, select a category.

The categories listed are those that you created in the FDMEE setup, such as "Actual." See "Defining Category Mappings" on page 106.

- 3 Optional: Enter a description.
- 4 In **Directory**, enter the relative path where the file is located.
- 5 From the **Plan Type** drop-down, select the plan type.
- 6 In the File Name field, enter the static name of the file.

If only the file name is provided, then data must be entered for a single period on the Rules Execution window.

To load multiple periods users can create a file for each period and append a period name or period key to the file name. When the rule is executed for a range of periods the process constructs the file name for each period and upload the appropriate to the POV.

Note: Period names cannot include spaces when used in a batch script.

To navigate to a file located in a FDMEE directory, click **Select** and then choose a file on the **Select** screen. You can also select **Upload** on the **Select** screen, and navigate to a file on the **Select a file to upload** screen.

If you do not specify a file name, then FDMEE prompts you for the file name when you execute the rule.

7 To load data into multiple periods, in the File Name Suffix Type drop-down, select either Period Description or Period Key.

A suffix is appended to the file name, and FDMEE adds the file extension after adding the suffix. If you leave the file name blank, then FDMEE looks for a file with Suffix. When the file name suffix type is provided, then the file name is optional in this case, and it is not required on the Rule Execution window

If the file name suffix type is a period key, the suffix indicator and period date format are required (as the suffix set) in the file name, and must be validated as a valid date format. For example, you might specify:

- a. 1 Jan-2013.txt
- b. 1 Feb-2013.txt
- c. 1 Mar-2013.txt

In this case, when you run the rule, enter 1_.txt in the file name field and select "Period Name" for the suffix indicator. Then run the rule for January to March periods

- 8 In Period Key Date Format, specify the data format of the period key that is appended to the file name in JAVA date format (SimpleDateFormat).
- 9 From the Import Format drop down, select the format when importing the file for the location.
 If the import format is not specified, then the import format from the Location is used.
- 10 Click Save.

Using a Multiple Period Data Load with a Data Load Rule

In the Data Load Rule, you can specify an import format for single and multiple period data rules, which enables you to load single or multiple period files from the same location. The file can have a single period or multiple data values that correspond to different periods. ExIn this case, the import format selected must have the same target as the Location selected in the POV. If the import format is not specified, then the import format from the Location is used.

The starting and ending period selected for the rule are used to determine the specific periods in the file when loading a multiple period data file.

In the data file, when amounts are not available for contiguous periods, then you can explicitly map the respective amount of columns to required periods in the data rule. When you execute the rule, the data is loaded to the periods as specified in the explicit mapping.

Managing Data Load Rules

Subtopics

- Editing Data Load Rules
- Running Data Load Rules
- Checking the Data Load Rule Status
- Deleting Data Load Rules
- Creating Integration Information

You can perform the following tasks to manage your data load rules:

- Edit data load rules—See "Editing Data Load Rules" on page 163.
- Run data load rules—See "Running Data Load Rules" on page 163.
- Delete data load rules—See "Deleting Data Load Rules" on page 168.
- View data load rules before executing them—See "Using the Data Load Workbench" on page 168.

• Check the data rule process details—See "Viewing Process Details" on page 180.

Editing Data Load Rules

If the data load rule is not in the process of running, you can modify rule details.

- To edit data rules:
- 1 From the Workflow tab, and then under Data Load, select Data Load Rule.
- 2 Select the data load rule.
- 3 Modify any of the data load rule details as necessary.
- 4 Click Save.

Running Data Load Rules

You run the data load rule to load updates and push the data into the target application. All submitted rules are processed by Oracle Data Integrator. When you submit a data load rule, specify the data extract options.

Data Load Rules can be executed by selecting one of the methods below:

- Execute command on the Data Load Rule screen.
- Import Source option in the Data Load Workbench option.
- Executing a batch. See "Executing Batches" on page 255.
- Running a batch script. See "Working with Batch Scripts" on page 260.

When a data load rule is run, it loads the data and a drill region (optional) is created to allow users to drill through to the source data.

When you run a data load rule, you have several options.

• Import from Source—FDMEE imports the data from the source system, performs the necessary transformations, and exports the data to the FDMEE staging table.

Select this option only when:

- O You are running a data load rule for the first time.
- Your data in the source system changed. For example, when you reviewed the data in the staging table after the export, and it was necessary to modify data in the source system.

In many cases, source system data may not change after you import the data from the source the first time. In this case, it is not necessary to keep importing the data when it has not changed.

When the source system data has changed, you need to recalculate the data.

• Export to Target—Exports the data to the target application.

Select this option after you have reviewed the data in the staging table and you want to export it to the target application.

Note: Select both options only in cases where the data has changed in the source system *and* you want to export the data directly to the target application.

- To submit the data load rule for a file-based source system:
- 1 From the Workflow tab, and then under Data Load, select Data Load Rule.
- 2 In **Data Load**, select the data load rule.
- 3 Click Execute.

When the data rule is run for Financial Management target applications, the Exchange Rates from the source are populated only up to the FDMEE interface table AIF_HS_EXCHANGE_RATES. The Core Exchange Rates table in Financial Management are not updated.

- From Execute Rule, to extract data and/or metadata from the source system and push it into target applications, select Import from Source, and then select the Start Period and End Period.
 - **Tip:** You can use a utility outside of FDMEE to view the data in the staging table. After you review the exported data, you can return to FDMEE, make modifications, and run the rule again. If you are sure that the information in the staging table is what you want to export to the target application, you can run the rule again and select "Export to Target." For information on staging tables, see "Staging Tables Used for Import from Source" on page 357.
- 5 Select Recalculate to remap all imported source data using the current mapping table and to recreate all logic accounts.
- 6 Select **Export to Target** to create the export file to export data to the target application.
- 7 From Start Period, select the beginning period of the POV from which to import the data from the source system.
- From End Period, select the ending period of the POV to which to import the data from the source system.
 - The following steps 9 and 10 are only relevant when the rule definition does not specify the file name to load.
- 9 For a file-based source system, in File Name, specify the file name of the source system.
 - To browse for a file, click **Select** and navigate to the file, or click Download or Upload on the Select Screen.
- 10 For a file-based source system. from File Name Suffix Type, select to affix either the period name or period key after the file name.
 - Period Key—A unique identifier for the period. The period key is a date value. When
 the file name suffix type is period key, specify the date format in the Period Key Data
 Format field.

 Period Name—A secondary identifier for the period. The value is unique, and may contain alpha-numeric characters.

Note: Period names cannot include spaces when used in a batch script.

- 11 For a file based source system, in Period Key Data Format, specify the date format when a period key suffix is selected.
- 12 For a file-based source system, In **Import Mode**, select the method for loading data.

Available load methods:

• Append—Existing rows for the POV remain the same, but new rows are appended to the POV (that is, appends the new rows in TDATASEG).

For example, a first time load has 100 rows and second load has 50 rows. In this case, FDMEE appends the 50 rows to TDATASEG. After this load, the row total for the POV is 150.

• Replace—Replaces the rows in the POV with the rows in the load file (that is, replaces the rows in TDATASEG).

For example, a first time load has 100 rows, and a second load has 70 rows. In this case, FDMEE first removes the 100 rows, and loads the 70 rows to TDATASSEG. After this load, the row total for the POV is 70.

- 13 In Export Mode, select
- 14 Click Run.

After you click Run, the rule is locked from any updates to ensure that the drill through path is intact. To check the status of the rule, see "Checking the Data Load Rule Status" on page 167.

- To submit the data load rule for a Planning system:
- 1 From the Workflow tab, and then under Data Load, select Data Load Rule.
- 2 In **Data Load**, select the data load rule.
- 3 Click Execute.

When the data rule is run for Financial Management target applications, the Exchange Rates from the source are populated only up to the FDMEE interface table AIF_HS_EXCHANGE_RATES. The Core Exchange Rates table in Financial Management are not updated.

- 4 From Execute Rule, to extract data and/or metadata from the source system and push it into target applications, select Import from Source, and then select the Start Period and End Period.
 - **Tip:** You can use a utility outside of FDMEE to view the data in the staging table. After you review the exported data, you can return to FDMEE, make modifications, and run the rule again. If you are sure that the information in the staging table is what you want to export to the target application, you can run the rule again and select "Export to Target."

- 5 Select Recalculate to remap all imported source data using the current mapping table and to recreate all logic accounts.
- 6 Select Export to Target to export data to the target application.
- 7 From Start Period, select the beginning period of the POV from which to import the data from the source system.
- 8 From End Period, select the ending period of the POV to which to import the data from the source system.
- In Import Mode, select the mode to extract data all at once for an entire period or incrementally during the period.

The three data extract types:

Note: The snapshot import mode is the only way data can be extracted from SAP and JD Edwards source system.

• Snapshot—Extracts everything for the selected source set for an entire period.

Note the following behavior of Snapshot mode:

- When the source data for the selected period *has never been run*, data is extracted from the source.
- When the source data for the selected period *has been run*, data is extracted from the FDMEE staging table, and not from the source. This means that if a user has two locations that extract from the same ERP source, data is extracted from the source only once (and the first time only).

For example, if you load data to Financial Management from the EBS for a selected period, but you run the integration to ARM for the same source and period, use what is in the interface tables since data is only extracted the first time from the source. This result in a significant performance gain for any subsequent data loads. The first extraction take the longest, but any other subsequent extractions are fast.

- Incremental—Extracts those records that were added after the prior data extract
- Full Refresh—Performs a clean extraction from the source system, thereby clearing any existing data rows in the appropriate FDMEE staging tables for a given source Ledger (or Business Unit) and source period.

Note: The import mode options (Snapshot, Incremental and Full Refresh) are only applicable to Data Rules in a Location using a Standard Import Format. Data Rules in a Location with a Source Adapter Import format always perform a full data extraction (similar to Full Refresh) directly into the TDATASEG_T table.

10 In **Export Mode**, select the mode for exporting whedata:

Available export modes for Planning and Essbase:

• Store Data—Inserts the data from the source or file into the target application. This replaces any value that currently exists.

- Add Data—Adds the value from the source or file to the value that exists in the target application. For example, when you have 100 in the source, and 200 in the target, then the result is 300.
- Subtract Data—Subtracts the value in the source or file from the value that exists in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
- Override All Data—Clears all data in the target, and then loads from the source or file. For example when you have a year of data in your planning application, but are only loading a single month, this option clears the entire year before performing the load.

Available export modes for Hyperion Financial Management:

Merge—Overwrites the data in the application with the data in the load file. For each
unique point of view that exists in the data file and in the application, the value in the
data file overwrites the data in the application.

Note: If the data load file includes multiple values in the file for the same point of view, the system loads the value for the last entry.

Note: Data in the application that is not changed by the data load file remains in the application.

- Accumulate—accumulate the data in the application with the data in the load file. For
 each unique point of view in the data file, the value from the load file is added to the
 value in the application.
- Replace—Replaces the data in the application with the data in the load file. For each unique combination of Scenario, Year, Period, Entity, and Value in the data file, the Replace option clears all account values from the application, then loads the value from the data file.
- Replace by Security—Performs a data load in Replace mode in which only the members
 to which you have access are loaded. This option enables you to perform a data load in
 Replace mode even when you do not have access to all accounts. When you perform the
 Clear operation for a period in a sub-cube, only the cells to which you have access are
 cleared. Data, cell text, and line item detail are cleared, but cell attachments are not
 cleared.

11 Click Run.

After you click Run, the rule is locked from any updates to ensure that the drill through path is intact. To check the status of the rule, see "Checking the Data Load Rule Status" on page 167.

Checking the Data Load Rule Status

After you run a data rule, you can check the status on the Process Details page. See "Viewing Process Details" on page 180.

Deleting Data Load Rules

You can delete data load rules created in FDMEE. You cannot delete data load rules when they are in the process of running.

When you delete a rule, all data loaded using the data rule are also deleted.

Note: After you delete data load rules, you can delete a source system. After you execute a deletion, users cannot use to drill through.

- To delete a data load rule:
- 1 From the Workflow tab, and then under Data Load, select Data Load Rule.
- 2 Enter the Location Name or click $\stackrel{>}{\sim}$ to select the location.
- 3 Select the data load rule.
- 4 Click Delete.

Creating Integration Information

You can specify free form text or a value about a location or data load using the integration option feature. Text or values entered can be used with your FDMEE scripts.

- ➤ To specify integration information:
- 1 From the Workflow tab, and then under Data Load, select Data Load Rule.
- 2 Click Edit Integration Options.
- 3 In Integration Option 1-4, specify the free form text or value and click OK.

The information you specify is accessible from the Integration Option fields of the Location table.

Using the Data Load Workbench

The Data Load Workbench feature provides a framework to import, view and verify, and export data from source systems in the FDMEE.

Key features of the Workbench include

- Interactive Load Process with options for Import, Validate, Export, and Check.
- Provision to view Source (All) / Source (Mapped) / Target / Source and Target values
- PTD / YTD Value display for ready reference
- Display options for Valid, Invalid, Ignored and All Data
- Online and Offline Load process
- Query option for Historical Loads

- Historical Loads Export to Excel
- Drill-back to source from Workbench
- Load, check and post journals for Financial Management applications

Note: The Data Load Workbench and Write-Back Workbenches share an almost identical user interface. For this reason, both features are documented in this section. Exceptions are noted when applicable.

When you login with the Run Integration role, these links are visible in the Tasks pane: Data Load Workbench, Data Load, Member Mapping, HR Data Load, Metadata, and Process Detail.

The Data Load Workbench consists of four sections:

- Workflow Grid
- POV Bar
- Status
- Data Grid

Workflow Grid

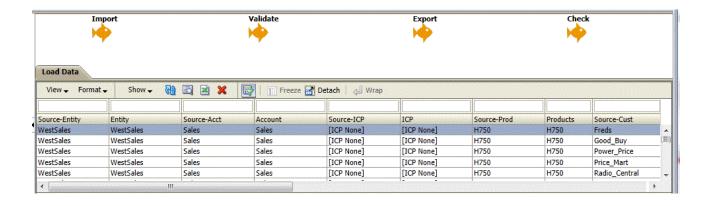
The Workflow grid enables users to process data from start to finish in FDMEE. The Workflow grid items are displayed as headers in the FDMEE Workspace display and correspond to a Workflow step. The steps consists of Import (loading data from source), Validate (ensures that all members are mapped to a valid account), Export (loads the mapped members to the target application), and Check (verifies accuracy of data by processing data with user-defined check rules).

When you select a Workflow step, the following occurs:

- Import—Initiates the Import action.
- Validate—Initiates the Validate action (even when the Import process has not been run for the current POV) but does not validate the data.
- Export—Initiates the Export action (even when the current POV has not validated its data) but does not initiate the Export process.
- Check—Displays the Check report for the current POV (when there is no check report data for the current POV, a blank page is displayed.)

FDMEE uses fish icons to indicate the status of each step. When a Workflow step has been completed successfully, the fish is shown in the color orange. If the step was unsuccessful, the fish is displayed with a gray color.

Note: You can customize the icons that show a "successful process" and a "failed process" by replacing the ProcessSucceeded and ProcessFailed icons located in the %EPM_ORACLE_HOME%\epmstatic\aif\images\general folder.



Processing Data

FDMEE process flow consists of four main operations:

- 1. Import—Imports the source data (GL).
- 2. Validate—Ensures that all data in the imported GL is has a corresponding mapping. Unmapped items must be assigned to a target account before proceeding to the Export step.
- 3. Export—Loads the mapped GL data to the target application.
- 4. Check—Validates the data loaded to the target application using the validation rules (when applicable).

Step 1: Importing Source Data

The Import from Source feature enables the Oracle Data Integrator to import the data from the source system, performs the necessary transformation, such as import, map and validate the data. The Import from Source features also allows you to import the source either online (immediate processing) or offline (runs in background).

Select this feature only when:

- You are running a data load rule for the first time.
- Your data in the source system changed. For example, when you reviewed the data in the staging table after the export, and it was necessary to modify data in the source system.

In many cases, source system data may not change after you import the data from the source the first time. Therefore, it is not necessary to keep importing the data when it has not changed.

- To import source data:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- Optional: When you import a source file, FDMEE uses the current POV to determine location, category, and period.

To import another source file, you must change the POV. For information on changing the POV, see "Using the POV Bar" on page 34.

3 At the top of the screen, click Import.

- 4 In Execution Mode drop-down, select the mode of importing the source.
 - online—ODI processes the data in sync mode (immediate processing).
 - offline—ODI processes the data in async mode (runs in background).
 - Click to navigate to the Process Detail page to monitor the ODI job progress.

5 Click OK.

The color of the Import fish changes to orange.

Step 2: Validating Source Data

FDMEE forces the validation of the source data against the mapping tables. Validation compares the dimension mapping to the source file and identifies unmapped dimensions. The process flow cannot continue until all dimensions are properly mapped.

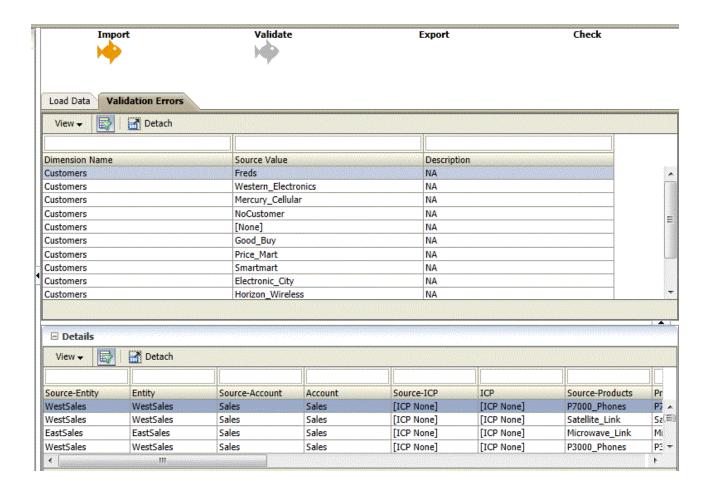
- To run the validation process,
- 1 Select the Data Load Workbench.
- 2 At the top of the main FDMEE screen, select Validate.

Validation with No Mapping Errors

When validation is successful, the orange Validate fish is displayed in the header of the FDMEE screen.

Validation with Mapping Errors

Because newly added dimension members may be unmapped, the addition of dimension members to source can produce validation errors. If a dimension member is unmapped, the Validate fish is grey and a Validation Error screen is launched in the Workbench, which shows the number of dimension members that are not mapped (that are, therefore, undefined). You must correct any unmapped dimension members before running the validation again.



- ➤ To correct conversion-table errors:
- 1 In the top region of the **Validation** screen, highlight a row that requires correction.
- 2 Correct any unmapped accounts.

In the top region of the Validation Error screen, the unmapped item is inserted into the Source value, and the bottom region shows all row with that source value. For example when the top region displays a value of Entity dimension with Source Value 01, then bottom region should show all rows where ENTITY = '01'.

For information on how to define member mappings, see "Defining the Data Load Mappings" on page 93.

3 Click Validate to refresh the validation form.

Source data that passes the validation process can be loaded to the target system.

Validating Financial Management Data Intersections

When running the validation step, Financial Management users can check the data intersections (cell status of account, entity, and so) from the Financial Management target application by running the Intersection Check Report.

Intersection check reports are generated as part of the data validation step in the data load workflow.

This feature is enabled in the Check Intersection option in Application Options. See "Defining Application Options for Financial Management" on page 74.

- To run an intersection check:
- 1 In the top region of the Validation screen, click Generate Intersection Check Report.
- 2 When prompted, save or open the Intersection Check Report.
- 3 Correct any validation errors and run the Validation step.

Step 3: Exporting Data to Target

After the source data has passed the validation process, use the Export option to export data to a target application. Select this option after you have reviewed the data in the data grid and you need to export it to the target application.

When exporting data for Planning and Essbase, you can store, add, and subtract data. For Planning and Essbase, you can override all data.

For Financial Management, you can merge, accumulate, replace, and replace by security data.

When you use Lifecycle Management to export mapping rules, any related mapping scripts are included.

The export of mapping rules to a CSV or Excel format does not include any scripting.

- To submit the data load rule:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- Optional: When you import a source file, FDMEE uses the current POV to determine location, category, and period and conducts the following process: To import another source file, you must change the POV. For information on changing the POV, see "Using the POV Bar" on page 34.
- 3 At the top of the screen, click Export.
- 4 In Execution Mode drop-down, select the mode for exporting the source data to the target application.
 - online—ODI processes the data in sync mode (immediate processing).
 - offline—ODI processes the data in async mode (runs in background).
 - Click to navigate to the Process Detail page to monitor the ODI job progress.
- 5 Click OK.

Step 4: Checking the Data

After exporting data to the target system, execute the Check step to display the Check report for the current POV (when there is no check report data for the current POV, a blank page is displayed).

Users can select the default report type that is used when Check reports are run. By default, the Publish Type field on the Reports page is set to the selected report-type value. Selections for this field are PDF, Excel, Word, Rich Text, and HTML.

Using the Workbench Data Grid

The data grid includes two tabs in the main grid:

- Load Data/Load POV—Use to import, view and verify and export data from source systems.
- drill through to the source data
- view mapping details

You perform tasks on the data grid by selecting options on the Table Action including:

- "Viewing Data" on page 174
- "Formatting Data" on page 175
- "Showing Data" on page 176
- "Opening Loaded Data in Microsoft Excel" on page 177
- "Querying by Example" on page 178
- "Freezing Data" on page 178
- "Detaching Data" on page 178
- "Wrapping Text" on page 179
- "Attaching Cell Text and Documents to a Data Cell" on page 179

Viewing Data

The following View drop-down options provides multiple ways to view data.

Table 30 View Options and Descriptions **Description** View Option Customizes views. Options include: View ▼ • Table—Selects the source or target data to display in the grid including: Source (All)-Shows both mapped and unmapped source dimensions (ENTITY, ACCOUNT, UD1, UD2,... AMOUNT). O Source (Mapped)—Shows only mapped source dimensions. Target—Shows only target dimensions (ENTITYX, ACCOUNTX, UD1X, UD2X,....AMOUNTX). Source and Target—Shows both source and target dimensions (ENTITY, ENTITYX, ACCOUNT, ACCOUNTX, UD1, UD1X, AMOUNT, AMOUNTX). Columns—Selects the columns to display in the data grid including: Entity Account Version Product Department STAT Amount Source Amount Freeze/Unfreeze—Locks a column in place and keeps it visible when you scroll the data grid. The column heading must be selected to use the freeze option. To unfreeze a column, select the column and from the shortcut menu, select Unfreeze. • Detach/Attach-Detaches columns from the data grid. Detached columns display in their own window. To return to the default view, select View, and then, click Attach or click Close. Sort— Use to change the sort order of columns in ascending or descending order. A multiple level sort (up to three levels and in ascending and descending order) is available by selecting Sort, and then Advanced. From the Advanced Sort screen, select the primary "sort by" column, and then the secondary "then by" column, and then the third "then by" column. The search fields that display in the advanced search options differ depending on what artifact you are selecting. Reorder Columns—Use to change the order of the columns. When you select this option, the Reorder Columns screen is displayed. You can select a column and then use the scroll buttons on the right to change the column order. Query by Example—Use to toggle the filter row. You can use the filter row to enter text to filter the rows that display for

a specific column. You can enter text to filter on, when available, for a specific column, and then press [Enter]. To clear

a filter, remove the text to filter by in the text box, then press [Enter]. All text you enter is case sensitive.

Formatting Data

You can resize the width of a column either by the number pixel characters or a percentage. You can also wrap text for each cell automatically when text exceeds the column width.

- To resize the width of a column:
- Select the column to resize. 1
- From the table action bar, select **Format**, and then **Resize**.

The Resize Column screen is displayed.

3 In the first Width field, enter the value by which to resize.

You can select a column width from 1 to 1000.

- 4 In the second Width field, select either pixel or percentage as the measure to resize by.
- 5 Select OK.
- To wrap the text of a column:
- 1 Select the column with the text to wrap.
- 2 From the table action bar, select **Format**, and then **Wrap**.

Showing Data

You can select the type of data to display in the data grid including:

- Valid Data—Data that was mapped properly and is exported to the target application.
- Invalid Data—One or more dimensions that was not mapped correctly and as a result, the data is not exported to target.
- Ignored Data—User defined explicit IGNORE maps to ignore while exporting to target.
 IGNORE maps are defined in the member mapping by assigning a special target member of IGNORE.
- All Data—Shows all valid, invalid and ignored data.
- To show a type of data:
- 1 Select Show.
- 2 Select either Valid Data: Invalid Data, Ignored Data, or All Data.

Drilling Through to Source Data and Viewing Mappings

When data has been displayed in the Data Load Workbench, you can drill through to the source, view mappings and open the source document.

Note: If the source system is EBS/Peoplesoft and you have metadata rules, then the drill region is created based on the metadata rule. Otherwise, it is created based on the target members in the data load mappings. For Year, Period, and Scenario, FDMEE uses audit information to create the drill region.

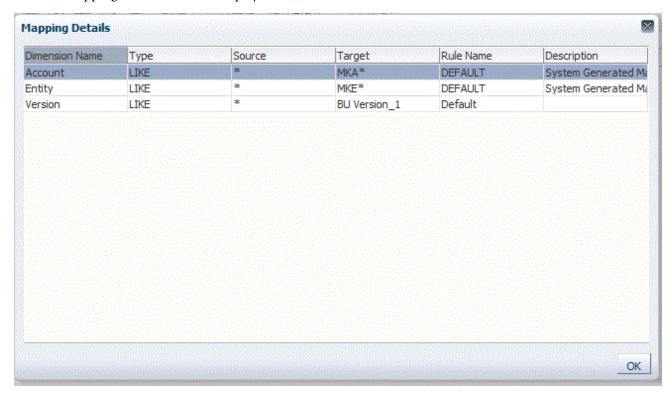
- To drill through to the source mapping:
- Select the type of data to display in the data grid.
 - See "Showing Data" on page 176.
- 2 From the Source Amount column, select an amount.

- 3 Click the source amount link and select Drill through to source.
- To view the source mapping detail:
- 1 Select the type of data to display in the data grid.

See "Showing Data" on page 176.

- 2 From the **Source Amount** column, select an amount.
- 3 Click the source amount link and select View Mappings.

The Mapping Details screen is displayed.



Opening Loaded Data in Microsoft Excel

When reviewing data in the workbench, users can drill down from the amount to ERP source system. In the source system the data is displayed in the granularity with which it was loaded.

You can open loaded data for a source system application in Microsoft Excel and review how the data is defined.

> To open loaded data in Microsoft Excel, from the table action bar, click .

You are prompted to open Microsoft Excel.

Querying by Example

Use the Query by Example feature to filter rows that display for a specific column. You can enter text to filter on, when available, for a specific column, and then press [Enter]. To clear a filter, remove the text to filter by in the text box, then press [Enter]. All text you enter is case sensitive.

- To query by example:
- 1 From the table action bar, click to enable the filter row.

The filter row must appear above the columns to use this feature.

2 Enter the text by which to filter the values in the column and press [Enter].

Note: When entering text to filter, the text or partial text you enter is case sensitive. The case must match exactly. For example, to find all target applications prefixed with "HR," you cannot enter "Hr" or "hr."

Freezing Data

Use the Freeze feature to lock a column in place and keeps it visible when you scroll the data grid.

- To freeze a column:
- 1 Select the column to freeze.
- $\mathbf{2}$ from the table action bar, click \mathbf{III} .
- To unfreeze a column:
- 1 Select the frozen column.
- 2 On the shortcut menu, select Unfreeze.

Detaching Data

Use the Detach feature to detach column from the data grid, When you detach the grid, columns display in their own window. To return to the default view, select *View*, and then, click *Attach* or click the Close button

- To detach columns:
- 1 Select the column.
- 2 From the table action bar, click \square .

The data grid is displayed in a separate window.

- To reattach a column:
- 1 Select the column.
- 2 From the table action bar, select View, and then Attach.

Wrapping Text

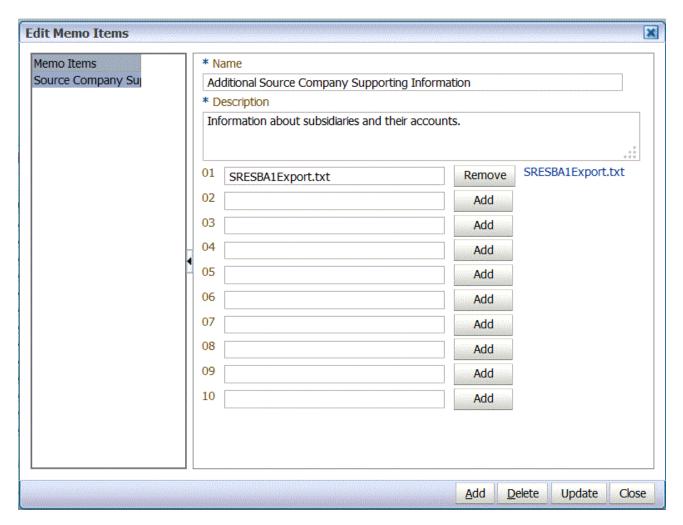
You can wrap text for each cell automatically when text exceeds the column width.

- To wrap text in a column:
- 1 Select the column.
- 2 Click 🔑.

Attaching Cell Text and Documents to a Data Cell

The cell text feature enables you to attach text and documents to a data cell. Multiple instances of cell text can be added as needed. FDMEE archives documents in EPM_ORACLE_HOME/products/FinancialDataQuality/data directory. Cell text can only be exported in Financial Management applications.

- To assign cell text and attach documents:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 In Data Load Workbench, select the data cell.
- From the memo column () in the column heading, click the memo link () to the left the data cell.
- 4 From Edit Memo Items, click Add.
- 5 In the Name field, enter a name of the memo.
- 6 In the **Description** field, enter a description of the memo.
- 7 Click Add (to the right of an Attachment field).
- 8 From the **Select** screen, browse and select an attachment, and then click **OK**.



- 9 Click Update.
- 10 Click Close.
- 11 Optional: To remove an attachment, click Remove (to the right of an Attachment field).

Viewing Process Details

You use the Process Details page to view submitted rule status and logs.

- To view data rule process details:
- 1 Select the Workflow tab, and then under Monitor, select Process Details.

The Process Details page is displayed, showing processes for all source systems. The following columns are displayed for each process:

- Process ID—An automatically generated identification number
- Status—Displays a visual indicator for the status of the process. You can rest the cursor over the icon to view a Screen Tip. Available statuses:
 - **《** —Rule Processed Successfully

- **②**—Rule Execution did not complete successfully
- Log—Click Show to display the log file.
- Location—Displays the location name
- Process Name—Type of process

Types of processes include:

- o Data Load—Initiated when you run a data load rule.
- o Metadata Load—Initiated when you run a metadata load rule.
- o HR Load—Initiated when you run an HR data load rule.
- o **Data Write-Back**—Initiated when you run a data write-back rule.
- Purge Process—Initiated when you remove an artifact, such as a target application or source system.
- Initialize Source System—Initiated when you initialize a source system.
- Rule Name—Name of the rule
- Source System—Name of the source system
- Accounting Entity—Name of the source accounting entity
- Target Application—Name of the target application
- ODI Session Number—The session number in Oracle Data Integrator. You can use this to look up a session in Oracle Data Integrator.

Note: The ODI Session number is present in Process Details only when the data is processed during an offline execution.

Job ID—The Performance Management Architect job ID

Process By—The user ID who initiated the process

- Reset Status—Resets the status to failed when a process continues to stay in a running status for a long period of time
- Link—Shows the log information for the process step. In the case of File Import, it shows skipped rows, and in the case of export to Planning, it shows rejected rows and so on.
- 2 Select a process to display the details. The following details are displayed:
 - **Status**—For each process step, the status is displayed. You can troubleshoot a problem by viewing at which point the process failed.
 - Process Step—Displays the steps in the process
 - Process Start Time—Time that the process step started
 - Process End Time—Time the process step ended
 - Log—If a log is available, you can click **Show** to display the log contents.
- Optional: To filter the rows that display, ensure that the filter row appears above the column headers.

 (Use the to toggle the filter row.) Then, enter the text to filter.

You can filter:

- Status—Enter SUCCESS, FAILED, or WARNING.
- Process ID
- Location
- Rule Name
- Source System
- Accounting Entity
- Target Application

Note: When entering text to filter, the text or partial text that you enter is case sensitive. For example, to find all target applications prefixed with "HR," you cannot enter "Hr" or "hr." For additional information on filtering, see "FDMEE User Interface Elements" on page 32.

Using Excel Trial Balance Files to Import Data

An Excel trial-balance file is an Excel spreadsheet that, through the import screen, is formatted to one period, category, and location.

Text Trial Balance Files Versus Excel Trial Balance Files

Text trial-balance files and Excel trial-balance files are similar in two ways: They are both loaded to the current POV (category and period), and, on the import form, they use the same Append and Replace options.

Text trial-balance files and Excel trial-balance files differ in one way: text files can use only the standard import format, but Excel data files do not use any import formats.

When you an Excel Trail Balance template, the template can contain one or more periods. When doing a multiple period load, you need to create a dummy import format that indicates multiple periods. If only a single data value is in the Excel file, then you don't need an import format.

Downloading an Excel Trial Balance Template

- To download an Excel trial balance template:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 From the **Download Template** drop-down, select **Trial Balance**.
- 3 From the Open screen, open or save the template and click OK.

Defining Excel Trial Balance Templates

To define an Excel trial-balance template, you define the first row of the named region, which contains the metadata tags. Other information may be defined outside of the region, but the system only considers what is within the named region. For example, the template includes a title and an amount summary. These are outside the named region and are not processed when loading data.

To load data using a template, the system uses a named range definition to find the dimensions and the related data. For the trial balance template, the predefined range is called upsTB, and it can be seen using the "Name Manager" option in Excel.

The following template contains one line of metadata (row 1) and three lines of imported data (rows 5–7).

Dimension Values and Amount should be populated in the respective columns as per the Tags defined in row 1. To add additional dimension tags, columns can be added. Data can be added by way of additional rows.

When adding rows or columns, add them within the named region. Excel updates the region definition automatically. If you add rows outside of the region, update the region to include these new rows or columns. When adding dimension columns, add a dimension tag to specify when the column is an account, entity, intercompany transaction, amount or user defined (UD) dimension. Note that the entity dimension is represented by the tag for "Center".

FDMEE Dimension	Valid Tags
Account (Required)	A, Account, SrcAcctKey
Center (Required)	C, Center, SrcCenterKey
Description (Optional)	D, Description, SrcAcctDesc
IC Counter Party (Optional)	I, IC, ICCoParty
User Defined 1 - User Defined 20 (Optional)	1-20, UD1 — UD20, UserDefined1 —UserDefined20
Amount (Required)	V, Amount, SrcAmount

In the template that is provided with FDMEE, some of the rows are hidden. To update the columns and the column tags, you need to unhide these rows. To do this, select the row above and below the hidden rows, and then update the cell height. A setting of 12.75 is the standard height for cells, and this should show all of the hidden rows for the selected range in the sheet. You may want to re-hide the rows after you have made the necessary changes.

4	Α	В	C	D
1	Trial Bala	nce Templat	е	
2		-		
3				
4				
5	Account	Center	Description	Current Month
8				
9				
10				
11				

Adding a Multiple Period Data Load using Excel

You may also use the Excel Trial Balance template to load data to multiple periods. To do this, create a data rule using a multiple period import format. The import format does not have to contain any detail mappings, only the definition must have a multiple period. Using the multiple period data rule you can import the Excel Trial Balance File. You create a dummy import format and only select the specification for the multiple period. To load data for multiple periods, the column header must be in the format V1:PeriodKey, V2:Periodkey, etc. The period key must be specified in YYYY/MM/DD format. You do not have to define the source period mapping in the data rule. You also need to update the range to make sure any additional columns are included in the range. Below is a sample of an Excel file.

5	Account	Center	Description	Jan	Feb
6	A	C	D	V1:2013/1/1	V2:2013/3/1
8	Revenue		100	110	1,300
9	Expense		100	500	500
10	A CONTRACTOR OF THE PROPERTY O		and the services of the servic		

Using Journal Templates to Import Data

In Financial Management, you use journals to adjust data after it has been entered or loaded into base level entities. Journals provide an audit trail of changes made in the application and indicate, which users made adjustments and which accounts, entities, and time periods are affected.

FDMEE enables you to load Financial Management journal entries with journal templates. These templates are Excel spreadsheets that are formatted as journal entry input screens.

FDMEE journal templates are typically used for the following types of adjustments:

- GAAP adjustments to general ledger files
- Transformations of gross balance accounts into roll-forward accounts (Beg, Add, Del, End)
- Supplemental data entries (Head Count, Ratios, and so on)

Integrating Financial Management Journals

The integration Financial Management journal feature with FDMEE involves setting up the FDMEE application options, and integrating data values.

Financial Management Application Options

The following Financial Management application options must be configured before using the Journal feature in FDMEE:

- Enable Journal Load
- Journal Enable JV ID per Entity
- Journal Balancing Attribute
- Iournal Status

For information on setting up any of the above options, see "Defining Application Options for Financial Management" on page 74.

Data Values

Data value is an extra dimension that is only used when integrating with a Financial Management multi-dimension target system. The name of the dimension is "Value". The members in this dimension are: [Contribution Adjs], and [Parent Adjs]. When data is loaded to Financial Management, specify a member of the value dimension to indicate where the data is loaded. In the Location definition in FDMEE, specify an entry for the value dimension in the Data Value field. The Data Value is set on the Location screen by selecting the Search link.

When FDMEE creates the load file, this dimension value is entered for every data line loaded by this location. By default, when no value is entered in this field, and an integration with Financial Management is executed, the Data Value <Entity Currency> is the default value.

If you load journals to Financial Management, you can specify the value dimension member for data loads and for journal loads. The first ";" is the value member used for data loads, and the second field by ";" is the value member for journal loads.

When using the template, the system will pick up the value member by looking for the second field delimited by ";" in the value member field in the location

When Search is selected, FDMEE connects to the Financial Management to get a list of valid data values. FDMEE takes the values from Financial Management and adds rows created by FDMEE that are a concatenation of the original value and "Adjustment Data Values". FDMEE uses these newly created rows to manage journal loading to Financial Management.

The rows that FDMEE creates in the Data Value selection screen are:

- [Contribution Adjs];[Contribution Adjs]
- [Contribution Adjs]; [Parent Adjs]
- [Contribution Adjs];<Entity Curr Adjs>
- [Contribution Adjs];<Parent Curr Adjs>

- [Parent Adjs];[Contribution Adjs]
- [Parent Adjs];[Parent Adjs]
- [Parent Adjs];<Entity Curr Adjs>
- [Parent Adjs];<Parent Curr Adjs>
- <Entity Curr Adjs>;[Contribution Adjs]
- <Entity Curr Adjs>;[Parent Adjs]
- <Entity Curr Adjs>;<Entity Curr Adjs>
- <Entity Curr Adjs>;<Parent Curr Adjs>

Downloading a Journal Template

- ➤ To download a Journal template:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- 2 From the **Download Template** drop-down, select **Journal**.
- 3 From the Open screen, open or save the template and click OK.

Defining Journal Templates

You define an FDMEE journal template by specifying the metadata header that FDMEE uses to interpret the data contained in the template. Metadata consists of a series of tags that tell FDMEE what column contains the account number or what period in which to load. A named range is used to tell the system where to look for the header information and the data to load. For the journal template, the named region is called upsJournal, and the following details relate to the relative rows within the named range. The template that comes with FDMEE has a range that starts at row 16, but the metadata starts at row 1 within the range. See below for an example of the template that is provided with FDMEE.

The sample journal template below has two lines of actual imported data, and five lines of metadata. Rows 1-5 contain metadata and rows 6 and 7 contain the data values.

	А	В	С	D
1	ID-Texas100		1_Texas	
2			ACTUAL05	
3			1/31/2005	
4			Α	
5	А	С	V	D
6	1100	Texas	500.00	Reclass Cash
7	1210	Texas	(500.00)	Reclass Cash

Metadata Structure

The metadata header (Row 1-5) instructs FDMEE on how to find the relevant segments of data that it handles in this template. The following Row 1-5 topics explain how each piece of metadata is used by FDMEE.

Row 1 (Journal ID and Location Tag)

The tag in row 1 is used to set the Journal ID and the FDMEE location that the data should be loaded into. The Journal ID must be placed in row 1 of the Account column. The Location tag must be placed in row 1 of the Amount column.

Note: The journal ID is limited to ten characters.

Row 2 (FDMEE Category Tag)

The tag in row 2 sets the FDMEE category into which the journal loads. The category must be a valid FDMEE category. The FDMEE Category tag must be placed in the Amount column.

Row 3 (FDMEE Period Tag)

The tag in row 3 sets the period that the data should be loaded into. The period must be a valid FDMEE period. This tag must be placed in the Amount column.

Row 4 (Load Method Tag)

The tag in row 4 sets the journal load method within FDMEE. To append to an existing journal with the same Journal ID, type the letter **A**. If a journal exists with the same Journal ID within the same FDMEE point of view, then the new journal is appended to the previously submitted journal. To replace an existing journal with the same Journal ID, type the letter **R**.

If a journal exists with the same journal ID within the same FDMEE point of view, then the new journal replaces the old journal. This tag must be placed in the Amount column. The table below defines the possible tags. The "AZ" and "RZ" settings work the same as the "A" and "R" methods except that all zero value amounts are suppressed.

Method	Valid Tags
Append Journal	A, Append
Replace Journal (default)	R Replace
Append journal - Zero Suppress	AZ
Replace journal - Zero Suppress	RZ

Row 5 (Dimension Tags)

The tags in row five define the dimension that the amounts are loaded into. The table below defines the possible dimension tags.

FDMEE Dimension	Valid Tags
Account (Required)	A, Account, SrcAcctKey
Center (Required)	C, Center, SrcCenterKey
Description (Optional)	D, Description, SrcAcctDesc
IC Counter Party (Optional)	I, IC, ICCoParty
User Defined 1 - User Defined 20 (Optional)	1-20, UD1 — UD20, UserDefined1 —UserDefined20
Amount (Required)	V, Amount, SrcAmount

Creating Range Names Within Journal Templates

To define a journal template, you create a range name that includes all metadata and data cells and that begins with the prefix *ups*. For example, for a standard template, create the range name [upsStandardJV (A1 to D7)].

A journal template requires a named range. You can specify as many named ranges as needed. To do this, duplicate the original range included with the template. FDMEEFDMEE processes as many named ranges as the template file contains.

The following template references upsJournal, which starts at row 16. Therefore, rows 16–20 are the first five rows (the metadata header) of ups.journal. And, therefore, the metadata of the template is placed in rows 16–20.

Rows 4–14 provide an interface that facilitates creation of the metadata header. Metadata information is input in rows 4–14 and referenced by the metadata header.

(Journal data should be entered against the respective columns and by adding more rows within the range. The easiest thing to do is to add rows to the existing range and just use a single range, and use the default upsJournal. You add columns to the spreadsheet based on the dimensionality of the target application.)

A B	C	D	E	F	G	Н	1	J
1 HFM Journ	nal Entry							23-Jul-13
2								
2 3 Journal Voucher								
4								
5 Journal ID: 6		Enter the Journal ID. It must 10 or le	ess characters					
6								
7 Location:	RESBA2_EP900SE	B. Enter the Location Name. It must n	natch the value in POV in t	he User Interface				
8								
9 Category: 10	Current	Enter the Category Name. It must n	natch the value in POV in t	he User Interface				
	3/1/2003			440000000000000000000000000000000000000				
11 Period: 12 Period:	3r #2003	Enter the Period Name. It must mat	tch the value in PUV in the	User Interface				
13 Load Method:	R	A = Append, R = Replace						
14	311	A - Appello, N - Neplace						
15 Account	Center	Amount	Description	IC	UD1	UD2	UD3	UD4
16	0	SRESBA2_EP900SBA	İ			İ	İ	
17		Current						
18		3/1/2003						
19		R						
20 A	С	V	D	1	1	2	3	4
22 23								
24								
25								
26								
27								
28 29								
30								
	(0.)	1 +0	00					
34	Balance:	\$0.	00					

Processing Journal Files

The process for processing journals is:

- 1. Load a journal file in Excel format from the inbox directory.
- 2. Check whether the POV entered in journal matches the current POV in FDMEE. The ups range is also checked.

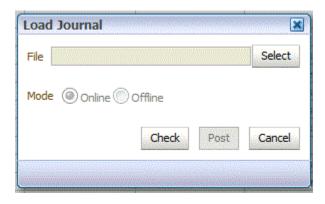
When a journal is checked in, FDMEE examines the template for all ranges with names beginning with ups. It then examines and validates the metadata tags found in each ups range. FDMEE does not check in metadata segments that include an invalid range.

3. Post the journal.

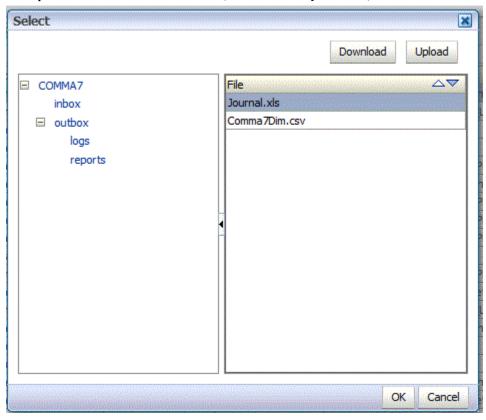
Loading Journals

- To load a journal:
- 1 From the Workflow tab, and then under Data Load, select Data Load Workbench.
- Optional: When you load a journal, FDMEE uses the current POV to determine location, category, and period. To use another POV, select another POV. For information on changing the POV, see "Using the POV Bar" on page 34.
- 3 Click Load Journal.

When a journal has been successfully loaded, the **Check** button is enabled. Complete the steps described in the "Checking In Journals" on page 190.



4 Optional: On the Load Journal screen, to browse for a journal file, click Select.



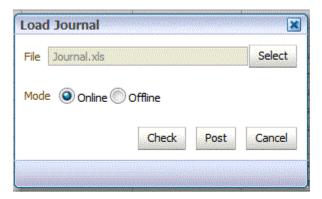
- a. On the **Select** screen, navigate to the journal file to load.
 - The journal file must be an Excel file (.xls).
- b. Optional: To download a journal file, click **Download** and open or save the journal file.
- c. **Optional**: To upload a journal file, click **Upload**, then navigate to the file to upload, and click **OK**.

Checking In Journals

Before journals can be posted, they must be checked in. This process verifies whether the POV entered in the Excel file for the journal matches the current POV. It also ensures that the ups range is valid. If the validation is successful, the Post button is enabled.

- To check in journals, from the Load Journal screen:
- 1 Make sure that a successfully loaded journal file is in the File field.

The journal file must be an Excel (.xls) file type.



- 2 Click Check,
- 3 Select Online or Offline for the processing method.

Online checking runs immediately, and offline checking runs in the background.

4 Click Check.

When a journal is checked in, FDMEE examines the journal file for all ranges with names beginning with *ups*. It then examines and validates the metadata tags found in each ups range. FDMEE does not check in metadata segments that include an invalid range.

Posting Journals

After a journal has been checked in successfully, you can post the journal. Posting a journal appends or replaces the data displayed in the Import Format screen (as determined by the load method specified in the journal). To post the journal, click **Post**.

Journal Security

If the POV Lock option is enabled, FDMEE administrators and end users are restricted to posting journals to the FDMEE global POV.

Logic Accounts

4

ln.	Th	c	Ch	or	itei

Overview of Logic Accounts	193
Creating a Logic Group	193
Creating Accounts In A Simple Logic Group	194
Creating Complex Logic Accounts	200

Overview of Logic Accounts

Logic accounts are dynamically generated accounts, and are used to calculate supplemental values that are not provided in source files. Logic groups are associated with a target applications. (The logic group list of values is filtered in the location user interface based on the target application in was created.) Like all other source accounts, logic accounts can be mapped and loaded into target systems. Logic accounts are used for various functions:

- Statistical loading—Map one source account to multiple target accounts
- Conditional mapping—Map a source account based on its value
- Arithmetic mapping—Perform arithmetic operations on source values

Creating a Logic Group

The first step in creating logic accounts is to create a logic group. The logic group is then assigned to one or more locations. When a source file or source system is loaded to a location, logic accounts are generated when the logic group is assigned to a location.

A logic group must be defined as simple or complex. A simple logic group enables you to derive logic items only from the source account dimension. A complex logic group enables you to derive logic items from any combination of dimensions.

- To create a logic group:
- 1 On the Setup tab, under Data Load Setup, select Logic Group.
- 2 In Target Applications, select the target application for this logic group.
- 3 In Logic Group, select Add.

A row is added to the grid.

- 4 In **Logic Group Name**, enter a unique name.
 - Optional: In **Description**, enter a description of the logic group.
- 5 In Logic Type, select Simple Logic or Complex Logic.
- 6 Click Save.

Creating Accounts In A Simple Logic Group

Within a simple logic group, you can create individual logic accounts.

- To create accounts in a simple logic group:
- 1 On the Setup tab, under Data Load Setup, select Logic Group.
- 2 From the Logic Group summary grid, select the logic group.
 - The logic accounts currently contained in with the selected logic group are listed.
- 3 From the Logic Items grid, click Add.
- 4 Provide the requested information.

See "Logic Group Fields" on page 194.

Logic Group Fields

Logic accounts consist of the following fields:

- Item
- Description
- Criteria Type
- Criteria Value
- Operator
- Value/Expression
- Seq
- Export

Item

Specify the name of the logic account using the item field. The logic account that is named in the item field is displayed in the Workbench grid as the source account. This same account can be used as a source in a mapping rule. Oracle recommends that you prefix the names of logic accounts with an "L" or some other character to indicate that an account came from a source file, or was generated from a logic rule. Logic accounts can only be loaded to a target application when they are mapped to a target account.

Description

The description that you enter in the Description field is displayed in the Account Description field in the Workbench.

Criteria Type and Criteria Value

The operator in the Criteria Type field works with the source account that is specified in the Criteria Value field to determine from which source accounts the logic account is derived. One logic account can be derived from multiple source accounts.

Valid values for the Type field:

- Between
- Like
- In

Specifying Source Accounts

Between (Criteria Type)—Used when a range of source accounts is specified in the Criteria Value field. Separate the accounts that specify the range by a comma.

Type Field	Criteria Value Field
Between	1000,1999

Like (Criteria Type)—Used when the source accounts in the Criteria Value field contain wildcard characters. Use question marks (?) as placeholders and asterisks (*) to signify indeterminate numbers of characters.

Type Field	Criteria Value Field
Like	1??0
Like	10*

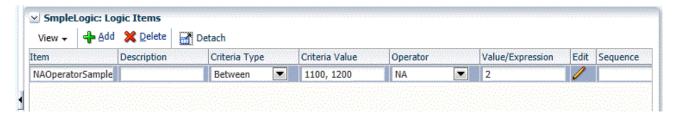
In (Criteria Type)—Used to include one source account or a list of nonsequential source accounts.

Type Field	Criteria Value Field
In	1000
In	1000,1005,2001

Operator and Value/Expression

NA Operator

NA (no operator)—If NA is specified, the source accounts that are specified in the Criteria Value field are summed. For example when the Type = Between and the Criteria Value = "1100,1200," then FDMEE creates one new account summarizing the values for source accounts that are between 1100 and 1200 for each entity where the accounts exist in the source file.

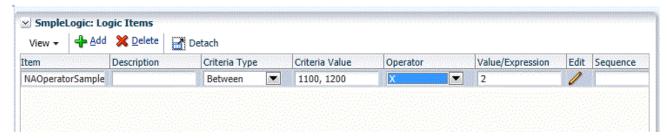


Math Operator

Math Operators (+, -, x, /)—If a math operator is selected, then the new logic records has an amount that equals the original amount is calculated with the specified Value/Expression. For example, when the operator "x" was selected and 2 is entered in the Value/Expression field, then the new record has an amount two times the original amount.

Use a numeric operator to perform simple mathematical calculations:

- NA (no operator)
- + (addition)
- - (subtraction)
- X (multiplication)
- / (division)
- Exp (expression operators)
- Function—see "Function" on page 197



In this example, one logic account is created because one Entity had a row meeting the account criteria.

Expressions and Functions

An expression enables you to perform the following in the Value/Expression field:

- Execute a simple math equation.
- Use a CURVAL parameter to specify the value of a logic account operation. The CURVAL parameter can be used within expressions, as it can within logic functions, except that, with expressions, CURVAL must be enclosed in pipes. For example, the CURVAL expression includes the specification of an account using the notation of |Account| and the specification of a value using POV details of entity, category, period and account.

Functions enable the use of simple logic with if/else using Jython syntax. Jython commands can be used in a function and can be more complex than an expression.

Exp

Use Expression operators to execute custom logic expressions, which are defined in the Value/ Expression field. Logic expressions, which cannot use variables or If statements, are simpler than logic functions. Except for |CURVAL|, expressions do not have built-in parameters. For expressions, you do not need to assign a value to RESULT.

Expressions execute faster than logic functions. You can use the FDMEELookup function within expressions, as it is used within logic functions. To write a custom expression, double-click the Value/Exp field to open the expression editor.

```
|CURVAL| + |810| + |238|
```

The function above uses the FDMEE Lookup function to add two source accounts to the value of the logic account. Notice that the CURVAL parameter can be used within expressions, as it can within logic functions, except that, with expressions, CURVAL must be enclosed in pipes.

```
(|CURVAL| + |000,10,09/30/01,810|) * 100
```

The function above uses the FDMEE Lookup function to add a source account (810) and a source account from a specified center, FDMEE category, and FDMEE period to the value of the logic account, and then multiplies the resulting sum by 100.

Function

Use function operators to execute a custom logic function defined in the Value/Expression field.

To write a function, select Function from the Operator drop-down list in the Logic Item line, and then click the edit icon to open the edit window. Logic functions are usually used for conditional mapping and other complex operations that involve multiple source accounts. Logic functions enable the use of Jython commands including variables, if/elif/else statements, numeric functions, and other Jython constructs.

The logic function enable the use of predefined function parameters, and also requires that you assign a value to the RESULT variable so that a value can be updated for the newly created logic account. The following function parameters can be used in a logic function, and these do not require using the "|" notation:

Table 31 Function Operators and descriptions

Function Operator	Description
CURVAL	Source value of the logic account operation
StrLocation	Active location name
StrCenter	Logic account entity
StrCatKey	Active category key, not the name. You need too lookup the category key in the database to use this parameter.
StrPerKey	Active period
Entity, Category, Period, Account	Lookup notation may be used in a logic function. This is the same notation provided in the logic expression.
Skip	If "Skip" is assigned to the keyword RESULT, then the logic account is not created.

You can define function parameters in uppercase, lowercase, or mixed case letters. However, the keyword RESULT must be in uppercase letters.

Assigning Function Return Values

The result of a Logic Function must be assigned to the keyword RESULT. If a return value is not assigned to the RESULT keyword, then the logic engine automatically sets the value of result to zero. This causes the calculation to be skipped and the logic account is not created.

The following function assigns the result of the logic account calculation (using the CURVAL parameter) to the logic account (RESULT) when the logic account calculation returns a value greater than zero. If the first condition is not met, then the logic account is not created because of the keyword "Skip."

```
if CURVAL > 0:
    RESULT = CURVAL
else:
    RESULT = "Skip"
```

Note: You must use the Jython notation and indentation for the logic function.

The following function only assigns the result of the logic account calculation to the logic account when "10" is the active FDMEE category key.

```
if StrCatKey == "10":
    RESULT = CURVAL
else:
    RESULT="Skip"
```

This function assigns the result of the logic account calculation to the logic account only when the Criteria Account Entity is "000."

```
if StrCenter == "000":
```

```
RESULT = CURVAL * 100
else:
    RESULT="Skip"
```

This function uses the FDMEE Lookup function to add a source account (810) to the value of the logic account if the current FDMEE period is "Dec 2013."

```
if StrPerKey == "12/31/2013":
    RESULT = CURVAL + |810|
else:
    RESULT="Skip"
```

This function uses the FDMEE Lookup function to add another source account from a different Entity, FDMEE category, and FDMEE period to the value of the logic account when the active location is "Texas."

```
If StrLocation == "Texas":
    RESULT = CURVAL + |000,10,09/30/13,810|
else:
    RESULT="Skip"
```

Value/Expression

To perform calculations and thereby, to derive values for a logic account, you select an operator, from the Operator field, to work with the Value/Expression value.

Seq

This field specifies the order in which the logic accounts are processed. Order specification enables one logic account to be used by another logic account, provided that the dependant account is processed first.

Export

A Yes-No switch determines whether a logic account is considered an export account and therefore is subjected to the conversion table validation process. If the switch is set to Yes, then you must map the logic account.

Creating Summarized Logic Accounts

By default, a logic account is created for each center in the trial balance. For example, when the Criteria Value field is 12300, the result is a logic account created for each source center that is associated with account 12300.

You can create a logic account that summarizes multiple source centers by, in the Criteria Value field, placing a semicolon after the account name and entering the number that identifies the number of characters to group by.

For example, when the value in the Criteria Value field is 12300;4, the result is a summarized account that includes all source centers that have the same first four characters in the source

center name. The source center assigned to account 12300 is the four characters that start at position 1. In addition, when the value in the Criteria Value field is 12300;3;4, the result is a summarized account that includes all source centers that have the same three characters in the source center, starting at position 4. The source center assigned to account 12300 is the three characters that start at position 4.

To create a logic account that summarizes all source centers by, in the Criteria Value field, place a semicolon after the account name and enter a text value. This hard-coded text value becomes the center for the summarized logic account. For example, when the value in the Criteria Value field is 12300;Dept100, then the result is a summarized account that includes all source centers. The source center assigned to account 12300 is Dept100.

Creating Complex Logic Accounts

Individual logic items are defined within a complex logic group. Each of the fields for a complex logic rule operates the same as a simple logic rule except for the Criteria Value and Include Calc fields. Complex logic accounts enable the user to enter a criteria value that includes other dimensions besides the account dimension. In addition, you can specify a "group by" and "group level" to alter how the generated logic accounts are displayed within the Workbench.

Criteria Value

To enter criteria for each dimension, click the Criteria Value icon to open the criteria form. The logic item is created only from the source line items that meet the specified criteria for each dimension. Descriptions of each complex logic criteria field is as follows:

Dimension

This field enables the selection of any enabled source dimension. You can select each dimension only once.

Criteria Type

This field works in conjunction with the Source Dimension and Criteria Value fields to determine from which source values the logic items are derived. Criteria types available are In, Between, and Like. The Criteria Type determines how the criteria value is interpreted.

Criteria Value

The criteria type uses this field to determine to determine what members to include in the logic calculation for any given logic dimension.

Group By

When viewing the derived logic item in the Workbench, the Group By field enables the logic item to override the displayed member in the appropriate dimensions field. You can override to group the dimension based on the value entered in the Group By field. Use this field to hard code the returned member, or append hard-coded values to the original members by entering a hard-coded member and an asterisk (*) in the Group By field.

For example, by placing the word "Cash" in the row with account selected for dimension, the Import form displays "Cash" in the Account field for the logic item. If you place "L-*" in the Group By field, the import form displays "L-1100" where 1100 is the original account that passed the logic criteria.

If you enter no value in the Group By field, no grouping occurs for this dimension, and a separate logic item is created for each unique dimension member.

Group Level

When viewing the logic item in the Workbench, the Group Level field works with the Group By field to override the displayed member in the appropriate dimensions field. This field accepts only numeric values.

When you enter a value of 3 in the Group Level field, the left three characters of the Group By field are returned. If no value is entered in the Group By field, then when you specify 3 in the Group Level field, first three characters of the original source dimension member are returned. The logic items displayed on the Import form can be grouped to the desired level.

For example, when you enter L-* in the Group By field, the logic item displays in the Import form as "L-1100," where 1100 is the original account that passed. When viewing the logic item in the Workbench, the Group Level field works with the Group By field to override the displayed member in the appropriate dimensions field. This field accepts only numeric values.

+ displays "L-11." If you enter the Group level1 for this row, then the Import form displays "L-1."

Include Calc

If it meets the logic item criteria, the Include Calc field enables the logic item to include previously calculated FDMEE values in its calculations.

Note: Each logic item has a sequence attached, and the logic items are calculated in this sequence. If the second, or later, logic item has this field enabled, then any previously calculated logic items are included, provided they meet the logic criteria.

Complex Logic Example 1: CashTx

Table 32 Complex Logic Example 1: CashTx

Dimension	Criteria Type	Criteria Value	Group By	Group Level
Account	Like	11*	Cash	0
Entity	Like	Тх	Texas	0
ICP	Between	00,99	ICP	0
UDI	In	00,01,02	UD1	0

The first row specifies that any accounts that begin with "11" are included in the calculated result for "Calc Item: CashTx."

The second row further qualifies the results by specifying that the source record must also have the entity like "TX."

The third line reduces the results to only those source records that have an ICP value between 00 and 09.

The last line reduces the results to only those source records that have a Custom 1 (UD1) of either: 00, 01 or 02. Imported lines that do not meet the listed criteria are excluded from the calculated results.

In the following table, only one new logic item is derived from multiple source records. Using the preceding graphic example as the logic criteria, and the first grid that follows as the source line items, you can see how FDMEE derives the value of a single logic item. Note the Group By field. Each Group By field includes a hard-coded value. Therefore, for every line that passes the specified criteria, the original imported member is replaced with the member listed in the Group By field.

Sample Imported Values

Account	Entity	ICP	UD1	Amount	Include or Exclude
1150	Tx	07	01	50,401.07	Include
1176	Tx	04	02	10,996.00	Include
1201	Тх	01	00	500.00	Exclude

Sample Imported Account Names

Account	Entity	ICP	UD1	Amount
Cash	Texas	ICP	UD1	50,401.07
Cash	Texas	ICP	UD1	10,996.00

FDMEE groups and summarizes the rows that include identical member combinations and thus creates the following result:

Final Result

Account	Entity	ICP	UD1	Amount
Cash	Texas	ICP	UD1	61,397.07

Complex Logic Example 2: CashTx

Table 33 Complex Logic Example 2

Dimension	Criteria Type	Criteria Value	Group By	Group Level
Account	Like	11*	Cash	0
Entity	Like	Tx	Texas	0
ICP	Between	000,100	*	2
UDI	In	00,01,02	UD1-*	0

The first row in the preceding table specifies accounts that begin with "11" are to be included in the calculated result for "Calc Item: CashTx."

The second row further qualifies the results by specifying that the source record must also have the entity like "TX."

The third line reduces the results to only those source records that have an ICP value between 000 and 100.

The last line reduces the results to only those source records that have a Custom 1 (UD1) of either: "00," "01." or "02." Any imported line that does not meet all of the listed criteria is excluded from the calculated results.

In the following tables, two logic items are derived from the source records because of the values entered in the Group By and Group Level fields. Two of the Group By fields have hard-coded values listed and two have an asterisk. Therefore, for every line that passes the specified criteria, the original imported members for the Account and Entity dimensions are replaced with the member listed in the Group By field. The other dimensions return all or part of the original members based on the Group Level entered.

Sample Imported Values

Account	Entity	ICP	UD1	Amount	Include or Exclude
1150	Тх	070	01	50,401.07	Include

Account	Entity	ICP	UD1	Amount	Include or Exclude
1176	Tx	040	02	10,996.00	Include
1121	Tx	045	02	9,050.41	Include
1201	Tx	100	00	500.00	Exclude

Logic Members

Account	Entity	ICP	UD1	Amount
Cash	Texas	07	UD1-01	50,401.07
Cash	Texas	04	UD1-02	10,996.00
Cash	Texas	04	UD1-02	9,050.41

FDMEE groups and summarizes the rows that include identical member combinations and thus creates the following result.

Final Result

Account	Entity	ICP	UD1	Amount
Cash	Texas	07	UD1-01	50,401.07
Cash	Texas	04	UD1-02	20,046.41

Check Rules

5

Th			

Overview of Check Rules	205
Creating Check Rule Groups	205
Creating a Check Rule	206
Creating Check Entity Groups	217

Overview of Check Rules

System administrators use check rules to enforce data integrity. A set of check rules is created within a check rule group, and the check rule group is assigned to a location. Then, after data is loaded to the target system, a check report is generated.

If a check entities group is assigned to the location, then check report runs for all entities that are defined in the group. If no check entities group is assigned to the location, the check report runs for each entity that was loaded to the target system. FDMEE check reports retrieve values from the target system, FDMEE source data, or FDMEE converted data.

FDMEE analyzes the check report and inserts a status entry in the process monitoring table. The location associated with the report shows a status of True only when all rules within the check report pass. For rules used only for warning, no rule logic is assigned.

Check reports run as data is loaded. You can also run the reports manually.

Note: Check rules are not applicable when loading to Accounts Reconciliation Manager.

Creating Check Rule Groups

- To create check rule groups:
- 1 On the Setup tab, under Data Load Setup, select Check Rule Group.
- In the Check Rule Group summary grid, click Add.
 - A row is added to the top grid.
- 3 In Check Rule Group Details, enter a name for the group in the Name field.
 - Optional: In **Description**, enter a description of the group.

4 Click Save.

Creating a Check Rule

Each line of a Check Rule report represents a check rule.

- To create check rules:
- On the Setup tab, under Data Load Setup, select Check Rule Group.
- Optional: In Check Rules, select the POV Location, POV Period, or POV Category.

```
See "Using the POV Bar" on page 34.
```

- 3 In the Check Rule Group summary grid, select the check rule group.
- 4 In the Rule Item details grid, click Add.

A row is added to the grid.

- 5 In each field, enter check rule information:
 - **Display Value**—See "Display Value" on page 206.
 - **Description** (optional)—See "Description" on page 207.
 - Rule Name—See "Rule Name" on page 207.
 - **Rule Text**—See "Rule Text" on page 207.
 - **Category**—See "Category" on page 207.
 - **Sequence**—See "Sequence" on page 208.
 - Rule Logic (optional)
- 6 Click Save.

Display Value

The Display Value field, which controls how FDMEE formats the data rows of check reports, is used to select target accounts or report format codes, or to create custom expressions.

Rules used to process Display Value field:

- For fields that contain report format codes, no value lookup is attempted.
- For fields that contain data other than report format codes, the result of the custom expression (rule logic) is displayed in the Value column. The Value column is limited to 75 characters.

Browse for Target Account

This option, which displays the **Search and Select: Target Value** screen, enables you to search and insert a target account (from a list of target-system application accounts) into the check rules form.

Select Format Code

This option enables you to enter format codes into the Target Account column.

Format codes determine the display of check reports.

Format Code	Action Performed on Check Reports		
#ModeList	Sets the report to display the Display Value, Description, and Amount column values. The system defaults to #ModeRule if nothing is specified.		
#ModeRule	(Default) Sets the report to display the Rule Name, Rule Text, and Amount column values.		
	The report evaluates each expression of the Rule Logic column and tests the True or False condition of each rule. The status of each rule (OK or Error) is displayed in the report.		
#Title	Inserts the text of the associated Description field as a title line on the check report.		
#Subtitle	Inserts the text of the associated Description field as a subtitle line on the check report.		

Description

Displayed only for check reports in #ModeList mode, the Description column displays account descriptions (which may be designated as titles or subtitles).

Example—Description

Out-of-Balance Account

Rule Name

Displayed only for check reports in #ModeRule mode, the Rule Name column, stores identifiers for check rules. Rule Name values should be unique and easy to identify.

Example—Rule Name

Out-of-Balance Check

Rule Text

Displayed only for reports in #ModeRule mode, the Rule Text column defines the logic behind rules. In check reports, the primary statement for a rule is the text of the Rule Text field associated with the rule.

Example—Rule Text

This account must be between [+10 and -10].

Category

In the Category column, select an FDMEE category to restrict a check rule to one FDMEE category. The rule is displayed in the check report only when the FDMEE category that is selected

in the Category field associated with the rule is the FDMEE category set in the POV. To display the check rule in the check report regardless of the category set in the POV, you must select All.

Sequence

Sequence column values (numbers) determine the order in which format codes and rules are processed. It is good practice to increment sequence numbers by 10—to provide a range for the insertion of format codes and rules.

Rule Logic

The Rule Logic column is used to create multidimensional lookups and check rule expressions. Rule Logic columns are processed for reports only in #ModeRule or #ModeList mode. After a rule logic is processed for a rule in the check report, FDMEE flags the rule as passing or failing.

Check Rule Expressions

Check rule expressions are used primarily to validate target-system account balances when performing a multidimensional lookups. The expressions return a True or False result.

For example, the following returns true (OK) if Cash (a target account) has a positive balance, and false (Error) when it does not:

```
|,,,YTD,<Entity Currency>,,Cash,[ICP None],[None],[None],[None],
[None],,,,,,,,,,,,,,,,,,
```

In this example, the check rule expression returns true (OK) when the value of Cash (a target account) plus \$1000 is greater or equals the value of AccruedTax (another target account), and false (Error) when it does not:

```
|,,,YTD,<Entity Currency>,,Cash,[ICP None],[None],[None],
[None],,,,,,,,,,,,,,,+1000>=|,,,YTD,<Entity Currency>,,AccruedTax,
[ICP None],[None],[None],[None],[None],,,,,,,,,,,,,,,,,
```

Using the Rule Logic Editor to Create Check Rules

The Rule Logic Editor facilitates creation of check rules. It helps you develop rule logic and enables you to create rules from the Rule Logic Editor, rather than from the **Check Rules** screen. You can also use the **Rule Logic Editor** to modify check rules.



- To open the Rule Logic Editor:
- 1 On the Setup tab, under Data Load Setup, select Check Rule Group.
- 2 From Check Rules, in the Check Rule Group summary grid, select a check rule group.
- 3 From the Rule Item Details grid, click Add.

A row is added to the grid.

- 4 In each field, enter check rule information:
 - **Display Value**—See "Display Value" on page 206
 - **Description**—(optional) See "Description" on page 207.
 - Rule Name—See "Rule Name" on page 207.
 - **Rule Text**—See "Rule Text" on page 207.
 - **Category**—See "Category" on page 207.
 - **Sequence**—See "Sequence" on page 208.
- 5 Click

The **Rule Logic** screen includes two tabs:

- Rule Logic Add/Edit
- Rule Logic Add/Edit as Text

Adding Rule Logic

Use the Rule Logic Add/Edit tab to add each row for the rule logic statement with a list of member values.

The Rule Logic Add/Edit tab consists of these elements:

- Rule Summary—Provides the logic for the check rule.
- Display Summary—Enables the specification of a multiple-dimensional lookup to be used as the display value when you want to display an amount different from the display value on the previous screen. The display summary is optional.
- Add—Adds the contents of the Rule and Display areas to the check rule being created or modified.
- Delete—Closes the Rule Logic Editor without saving changes.
- To add a rule logic statement:
- 1 On Rule Logic Editor, select the Rule Logic Add/Edit tab.
- 2 From the Rule summary grid, click Add.

A blank line is displayed.

3 Enter the rule to test.

Note: When using the equal sign for evaluating amounts, use double equal signs (==).

- 4 Optional: Click
- From Rule Logic in the Intersection Type field, select the intersection type for the multidimensional lookup.

Available intersection types:

- Source intersection—Values are enclosed by the "~" character.
- Converted source intersection—Values are enclosed by the 'character.
- Target intersection—Values are enclosed by the "|" character.

See "Multidimensional Lookup" on page 212.

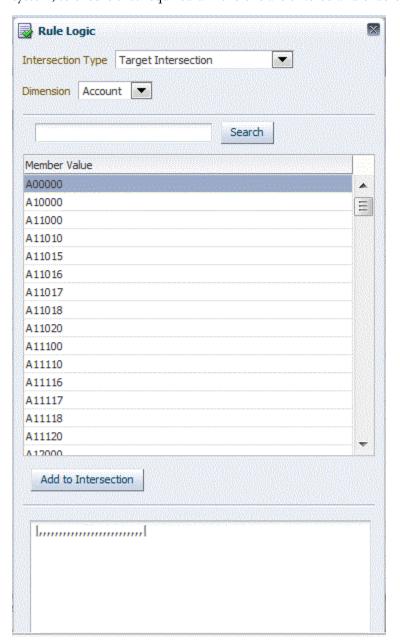
- 6 From **Dimension**, select the dimension from which to retrieve values.
- 7 From **Member Value**, select a value from the dimension.
- 8 Click Add to Intersection.

The member value is added to the Display area.

9 Click OK.

Rule Logic Tab

Use the **Rule Logic** tab, from which you can select "retrieve" dimensions directly from the target system, to ensure that required dimensions are entered and ordered correctly.



Display the Rule Logic tab by clicking from the Rule Summary or Display summary grid on the Rule Logic Add/Edit screen.

The Rule Logic tab contains the following elements:

- Rule and Display—Areas that, respectively, display the rule and the display rule that are being created.
- **Intersection Type**—Enables you to select the type of retrieval format for the target dimensions.

FDMEE uses the intersection type when multidimensional lookups are selected for a rules logic statement. The multidimensional lookup retrieves account values from the target system, FDMEE source data, target data or FDMEE source converted data. See "Multidimensional Lookup" on page 212.

Multidimensional Lookup

The multidimensional lookup retrieves account values from the target system, FDMEE source data, or FDMEE converted data. You can use multidimensional lookups in Rule Logic columns and in the Display columns of check rules forms.

Rule Data Sources

FDMEE can retrieve data from three sources:

- Target-system data
- FDMEE source data
- FDMEE converted data

Target System Data

The following format, which begins and ends the rule with the pipe character (|), enables FDMEE to retrieve target-system values for any dimension.

Unless otherwise specified, parameters are optional.

```
|Scenario, Period, Year, View, Value, Entity, Account (Required), ICP, Custom1, Custom2, Custom3, Custom4, Custom5, Custom6, Custom7, Custom8, Custom9, Custom10, Custom11, Custom12, Custom13, Custom14, Custom15, Custom16, Custom17, Custom18, Custom19, Custom20|
```

The following examples illustrate ways that target-system values can be retrieved. In each example, Balance is a target account. For dimensions that are not referenced, you must use commas as placeholders.

Example 1

Look up the value of Balance for the target period and scenario (category) set in the POV and for each entity of the FDMEE check entity group that is assigned to the location. The example rule passes the check when the target account is less than \$10.

Example 2

Look up the value of Balance for the specified dimensions.

Example 3

Look up the value of Balance for the specified dimensions and the previous period.

```
|Actual,-1,2002,YTD,Ohio,Balance,Michigan,Engines,Ford,Trucks,[None],,,,,,,,,,,,,,USD| > 0
```

Example 4

Look up the value of Balance for the target scenario (category) set in the FDMEE POV, the previous target period, and each entity of the FDMEE check entity group that is assigned to the location.

All missing custom dimensions default to [None]. The ICP dimension defaults to [ICP-None]. The Year dimension defaults to the year set in the POV. The Currency dimension defaults to 0. The View dimension defaults to YTD.

FDMEE Source Data

The following format, which begins and ends the rule with the tilde character (\sim), retrieves values from data that was mapped to a target member, and then loaded into FDMEE.

Unless otherwise specified, parameters are optional. Parameters designated UD# are user-defined.

~FDMEE Category, FDMEE Period, Year (Field Not Applicable), FDMEE View, FDMEE Location, Source Entity(Required), Source Account(Required), Source ICP, Source UD1, Source UD2, Source UD3, Source UD4, Source UD5, Source UD6, Source UD7, Source UD8, Source UD9, Source UD10, Source UD11, Source UD12, Source UD13, Source UD14, Source UD15, Source UD16, Source UD17, Source UD18, Source UD19, Source UD20~

FDMEE Converted Data

The following format, which begins and ends the rule with the grave accent character (`), retrieves pull values from data that was loaded into FDMEE. Unless otherwise specified, the parameters are optional.

`FDMEE Category, FDMEE Period, Year (Field Not Applicable), FDMEE View, FDMEE Location, Entity(Required), Account(Required), ICP, Custom1, Custom2, Custom3, Custom4, Custom5, Custom6, Custom7, Custom8, Custom9, Custom10, Custom11, Custom12, Custom13, Custom14, Custom15, Custom16, Custom17, Custom18, Custom19, Custom20`

Math Operators

Math Operators (+, -, x, /)—If you select a math operator, then the check rule has an amount that equals the original amount calculated with the specified expression. For example, when you select the operator "x" and enter 2 in the rule field, then the new record is an amount two times the original amount. The math operators available in the expressions:

- + (addition)
- - (subtraction)
- * (multiplication)
- / (division)
- math.abs ()

If/Then/Else

Check rules accept If/Then/Else statements after you perform a multidimensional lookups. This statement provides a primary path of execution when the if statement evaluates to "true," and a secondary path of execution when the if statement evaluates to "false."

Using the If/Then/Else statement, you can use custom-field values within reports as warning messages and flags.

In the following example, when the Result is between 100 to 1500, the Check Report with Warning prints "Amount between 100 and 1500." The example references three data accounts:

```
    24000050: 1000
    24000055: 500
    24000060: 10
```

This calculation for this example is 1000 + 500/10, with the result of 1050.

The script is written using Jython code:

```
def runVal():
                           dbVal=abs((|,,,,,BERLIN,24000050,[ICP None],[None],[None],[None],
 [None], , , , , , , , , , , , , | ) / (|, , , , , BERLIN, 24000060, [ICP None], [None], [Non
 [None],,,,,,,,,))
                           PstrCheckMessage1=''
msg2=''
msg3=''
                           if (dbVal<100):
                                      RESULT=True
                                       PstrCheckMessage1='Amount < 100.'
                           elif(dbVal>100 and dbVal<=1500):
                                       RESULT=True
                                       PstrCheckMessage1='Amount between 100 and 1500.'
                            elif(dbVal>1500 and dbVal<=9999):
                                       RESULT=True
                                       PstrCheckMessage1='Amount between 1501 and 9999.'
                           else:
                                       RESULT=False
                                       PstrCheckMessage1='Amount greater than 9999!'
                           return [RESULT, PstrCheckMessage1, msg2, msg3]
```

Note: You must include three message parameters with the return statement to write data to the status table. Regardless you are only writing a single message, the other two message parameters are required.

The result of running this script is shown in the Check Report with Warnings:

Location: MultiPer_SSN
Category: Actual
Period: 2008-05-01

Validation Group: Check

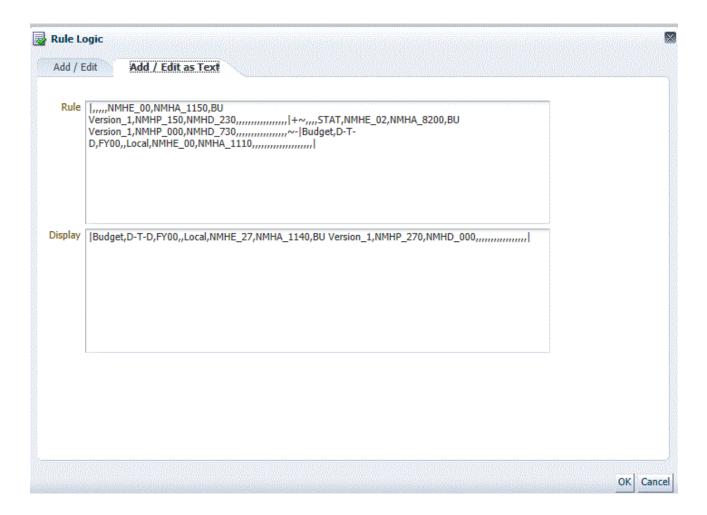
Pass BERLIN

	Account	Rule Definition	Value
Ok	24000050	24000050	1,050.00 Amount between 100 and 1500

Adding a Rule Logic Statement as Free Form Text

Use the **Rule Logic Add/Edit as Text** screen to add the rule to test manually as free-form text. This feature enables you to instruct the system how to display the actual value from the check rule logic. If you do not specify a display summary, then the report shows an unrelated value.

When the Display area is active, all interaction takes place in the Display area. For example, pasting a tree copies all tree members to the Display area. When the Rule area is active, all interactions take place in the Rules area. Changes made on the Lookup tab are reflected on the Rule tab in related editors.



- To add a rule logic statement:
- 1 On the Rule Logic Editor, select the Rule Logic Add/Edit as Text tab.
- 2 In Rule, enter the rule.

Do not use a semicolon (;) in check rules. The semicolon is a reserved as the separator between the rule value and the display value.

When using the equal sign for evaluating amounts, use double equal signs (==) instead of the single equal sign (=). For example, use a - b == 0, and not a - b =0.

3 Click OK.

Running Check Reports for Profitability and Cost Management BSO Essbase Cube

Before running a Check Report for a Oracle Hyperion Profitability and Cost Management BSO Essbase cube, make sure that a fully qualified name is used to differentiate any duplicate members used within dimensions or across dimensions. To construct a fully qualified name from a duplicate member name, add the fully qualified name as a rule logic statement on the Rule Logic Add/Edit as Text screen (see "Adding a Rule Logic Statement as Free Form Text" on page 215).

Note that a fully qualified member name comprises the duplicate member or alias name and all ancestors up to and including the dimension name. Each name must be enclosed in brackets ([]) and separated by a period (.). The syntax is as follows:

```
[DimensionMember].[Ancestors...].[DuplicateMember]
```

For example:

```
[Market].[East].[State].[New York]
[Market].[East].[City].[New York]
```

See Oracle® Essbase Database Administrator's Guide.

Creating Check Entity Groups

A check entity group consists of one or more target-system entities. When a check report is generated, the entities of the entity groups that are assigned to the report location are consolidated and displayed in the report. You activate check entity groups by assigning them to locations. You define the check entities of a check entity group by entering values in the fields of the check entities form of the **Check Entities** screen.

The Check Entity Group screen consists of three grids:

- Check Entity Group Summary—Summary area that lists the names of the check entity group, and enables you to create a new check entity group.
- Check Entity Group Details—Detail area where you can name and describe the check entity group.
- Entity Details—Detail area where you can add information about the entity.
- To add a check entity group:
- 1 On the Setup tab, under Data Load Setup, select Check Entity Group.
- 2 On the Check Entity Group grid, click Add.

A blank row is added at the top of the grid.

- 3 From the Check Entity Group Detail area, enter the check entity group name in the Name field.
 - Optional: In the **Description** field, enter a description of the check entity group.
- 4 Click Save.
- To add an entity detail:
- 1 In Check Entity Group summary grid, select a check entity group.
- 2 In Entity details grid, click Add.

Blank options lines are displayed.

- 3 Complete the following fields:
 - Parent

- Entity
- Consolidate
- On Report
- Sequence

4 Click Save.

 Table 34
 Entity Detail Options and Descriptions

Option	Description
Parent	Specify the organization in which the entity is consolidated. For other target systems, you select the parent of the entity. If the Consolidate option is not selected, the selection is irrelevant.
Script Name	For Planning and Essbase calculation scripts, specify the calculation script name to execute.
	This field is only available when the Check Entity Calculation method is set to "calculation script" in the Essbase or Planning application options.
Calc Script Parameters	Click to browse and set the script for the calculation script on the Set Calculation Script Parameters screen. You can also add a calculation script on the Set Calculation Script Parameters screen.
	As part of the check rule process, FDMEE references any calculation script parameters included in custom calculation scripts. A calculation script is a series of calculation commands, equations, and formulas that enable you to define calculations other than those defined by the database outline. Calculation scripts are defined in Essbase and Planning target application options, see "Using Calculation Scripts" on page 69.
	This field is only available when the Check Entity Calculation method is set to "calculation script" in the Essbase or Planning application options.
	If the Check Entity Calculation method is set to "dynamic", this field is unavailable.
Entity	Specify the target entity to consolidate and display in the check report. If the Consolidate option is selected, the entity is consolidated before it is displayed in the check report.
Consolidate	Select to consolidate an entity prior to displaying it in the check report.
	Planning—Runs the default calculation or calculation script specified in the Calc Script Name depending on the "Check Entity Calculation Method" property of the target Application.
	Essbase—Runs the default calculation or calculation script specified in the Calc Script Name depending on the "Check Entity Calculation Method" property of the target Application.
	Financial Management—The consolidation of data occurs in the Financial Management database.
On Report	The option selected in the On Report column determines whether an entity is displayed in the check report. If On Report is not selected and Consolidate is selected, the entity is consolidated but not displayed.
Sequence	Specify the order in which entities are consolidated and displayed in the check report.
	It is good practice to increment the sequence number by 10, to provide a range for the insertion of entities.

Data Write-Back

6

In This Chapter

Creating Mapping Definitions	220
Defining Data Write-Back Rules	221
Managing Write-Back Rules	225
Loading Data into Supported Source Systems	235

Financial budgeting in formation often must be compared with and controlled with actuals and stored in the general ledger system. You define data write-back rules to determine how you want to extract the budget data from Planning and write it back to the ERP system. For example, you may want to write-back budget data.

You can define write back rules for Planning, Essbase aggregate storage (ASO), and Essbase block storage (BSO) 11.1.2.x applications only as the source and for these target source systems:

- PeopleSoft Enterprise Financial Management
- Oracle E-Business Suite
- Fusion General Ledger

Considerations:

- Data Write-back is supported for Planning, Essbase aggregate storage, and Essbase block storage 11.1.2.x applications only. Applications created in earlier releases are not supported.
- For BSO Essbase applications, FDMEE requires that the Period dimension be designated as "Dense". Other dimension can also be Dense but Period is required to be dense for write back to work.
- Data Write-back is not supported for EPMA deployed aggregate storage Essbase cubes,
- For E-Business Suite source systems, you can post to budgets with or without budget journals.
- Only monetary and statistical amounts are eligible to be written back to the general ledger.
- You cannot create data write-back rules for PeopleSoft Human Capital Management source systems.
- FDMEE loads data into the specific data interface table. You must then run the budget load routines provided by Oracle Fusion, E-Business Suite or PeopleSoft Enterprise Financial Management.

- Allocation from a source amount to multiple target amounts is not provided.
- You must have the GL Write-back security role to create data write-back rules.
- Write-back can be performed without first loading data from a GL source to an EPM target.

Creating Mapping Definitions

Before you create a data write-back rule, you should create the write-back mappings. Data write-back mappings occur at the member level and are used for data write-back rules. (Loading data from a Planning application to your ERP source system.) You create write-back mappings to replace outgoing dimension members with source segment members. More specifically, during budget write-back, the write-back mapping is referred to when replacing outgoing dimension members with segment values.

Note: The write-back feature is not available for Source Adapter based imports.

The following interface tables require "write" security privileges for the data write-back process:

E-Business Suite

- GL INTERFACE
- GL_INTERFACE_CONTROL
- GL_BUDGET_INTERFACE

Standard PeopleSoft—PS_HPYPB_ACCT_LN

PeopleSoft Commitment Control

- PS_HYP_KK_BD_HDR
- PS_HYP_KK_BD_LN

For additional information on database tables access by FDMEE, see Appendix B, "Source System Tables Used by FDMEE."

- To create write-back mappings:
- 1 From the Workflow tab, and then under Write Back, select Write-Back Mapping.
- 2 Enter the **Location Name** or click to navigate and select the location.
- 3 Select the Segment.
- 4 Choose the mapping type by selecting either the Explicit tab, Between tab, In tab, Multi-Dimension, or Like tab.
 - Explicit—The source value is matched exactly and replaced with the target value. For example, the source value, "ABC" is replaced with the target value, "123." "Explicit" write-back mappings are created the same for data load and data write-back rules. See "Creating Mappings Using the Explicit Method" on page 128.

- Between—The range of source values are replaced with one target value. For example, a range from "001" to "010" is replaced as one value: "999." "Between" write-back mappings are created the same for data load and data write-back rules. See "Creating Mappings Using the Between Method" on page 129.
- In—In mappings enable a list of nonsequential source accounts to be mapped to one target account. In this case, multiple accounts are mapped to one account within one rule, eliminating the need to create multiple rules (as is required for an Explicit map).
- Like—The string in the source value is matched and replaced with the target value. For example, the source value, "Department" is replaced with the target value, 'Cost Center A. See "Creating Mappings Using the Like Method" on page 130.
 - Write-back mappings provide a means to remove or strip characters that were added during the data load process. "Like" write back mappings are created similar to, but reverse from the data load.
- Multi-Dimension—Define member mapping based on multiple source column values.

Tip: You can click Refresh Values to refresh the list of segment or chartfield values that appear in the drop-down list from the source system. This is especially helpful when creating "Explicit," "Between," "Like", and "Multi-Dimension" mappings for data write-back data loads.

Defining Data Write-Back Rules

Subtopics

- Defining the Source Filter Options
- Defining Target Options
- Defining Write-Back for PeopleSoft Commitment Control
- Deleting Write Back Member Mappings

You create a write-back rule for extracting budget data from an application to a general ledger instance and ledger source.

When you create a data write-back rule, you can write back in these ways:

- Choose the Planning application.
- Choose the Essbase aggregate storage (ASO) or Essbase block storage application (BSO).
- For Public Sector Planning and Budgeting applications where you have consolidated personnel (HR) and non-personnel expenses in the aggregate storage cube, you pull information from the aggregate storage application. For non-personnel related expenses you see only the total number (combination) in the aggregate storage application.

Note: Public Sector Planning and Budgeting requires that you combine the regular planning results from an Essbase block storage cube, with the personnel position budget information in an Essbase aggregate storage cube to new aggregate storage cube.

When performing the data write-back for a Public Sector Planning and Budgeting application, you select the aggregate storage cube that you created in Public Sector Planning and Budgeting. You can also select a Planning application as the source for the write-back.

The process at a high level:

- 1. FDMEE imports data from Planning and writes it to FDMEE staging table.
- 2. FDMEE applies the necessary transformation in the staging table.
- 3. Data is exported from the FDMEE staging table to a general ledger interface table. The mapping is what you use to format the data in the journal import tables.
- 4. After a successful execution of the write-back rule, log in to the general ledger source system and run the budget import process in the general ledger.
- To define data write-back rules:
- 1 From the Workflow tab, and then under Write Back Rule, select Write-Back Rule.
- 2 Enter the **Location Name** or click to select the location name.
- 3 Click Add to add a data write-back rule.
- 4 Enter the Name.
- 5 For Planning applications: Select the Plan Type.
- 6 Optional: Enter a description.
- 7 Click Save.
- 8 Define the source filters. See "Defining the Source Filter Options" on page 222.

Defining the Source Filter Options

You can define source filter options to specify the subset of budget data to extract from your Planning application and load in the general ledger.

- To define the source filter options:
- 1 In the Source Filters area, click
- 2 Select the Dimension Name.
- 3 To assign a filter condition, perform an action:
 - Enter a member name or filter condition in the Filter Condition text box.
 - You can enter the member name or filter condition using Essbase syntax. Depending on the dimension, you can select one or more members as a filter condition used for

extracting the budget data. For example, for the Entity dimension, you may select the following members: E1, E5, E6. For information on Essbase syntax, see the *Oracle Essbase Database Administrator's Guide*.

• Click to select a member using the member selector. Then, click **Browse**.

The Selector dialog box is displayed. The member selector enables you to view and select members within a dimension. Expand and collapse members within a dimension using the [+] and [-].

The Selector dialog box has two panes—all members in the dimension on the left and selections on the right. The left pane, showing all members available in the dimension, displays the member name and a short description, when available. The right pane, showing selections, displays the member name and the selection type.

You can use the Menu button above each pane to change the columns in the member selector.

Note: Assign filters for dimensions. If you do not assign filters, numbers from the summary members are also retrieved.

To use the member selector:

- a. In the list of available dimensions and members on the left, select a member and click
- b. To deselect a member from the list of members, click <u>\$\limes\$</u>.
- c. To add special options for the member, click and select an option.

In the member options, "I" indicates inclusive. For example, "IChildren" adds all children for the member, including the selected member, and "IDescendants" adds all the descendants including the selected member. If you select "Children," the selected member is not included and only its children are included.

The member is moved to the right and displays the option you selected in the Selection Type column. For example, "Descendants" displays in the Selection Type column.

Tip: To clear all members from the list of selections, click

d. Click **OK** twice to continue defining the source filter details.

The member that you selected is displayed in an Essbase syntax in the Filter Condition field.

- 4 Click Save.
- 5 Define the target options. See "Defining Target Options" on page 224.

Defining Target Options

The target options define the options for importing a journal.

- ➤ To define target ERP source system options:
- 1 For Oracle Fusion and E-Business Suite source systems:
 - a. Select whether to create a budget journal.
 - b. Enter the **Budget** or click to select the budget.
 - c. Enter the **Budget Organization** or click to select the budget organization.
- 2 Click Save.
- 3 For PeopleSoft Enterprise Financial Management source systems, enter or click to make selections
 - a. Budget Scenario
 - b. Ledger Group
 - c. Ledger
- 4 Click Save.
- 5 Run the data write-back rule. See "Running Write-Back Rules" on page 226.

After you run the rule, you must perform required tasks in Fusion, E-Business Suite, and PeopleSoft Enterprise Financial Management. See "Loading Data into Supported Source Systems" on page 235.

Defining Write-Back for PeopleSoft Commitment Control

- ➤ To write back to PeopleSoft Commitment Control:
- Make sure that the PS_HYP_KK_BD_HDR and PS_HYP_KK_BD_LN interface tables have "write" privileges.
- When writing back to PeopleSoft Commitment Control, in Write-Back Rule, define a write-back rule by selecting parameters for the following:
 - Ledger Group
 - Ledger
 - Accounting Date
 - As of Date
 - Transaction Type—Select "check" to validate the rules or "post" to validate and post the rules.
 - Budget Transaction Type
 - Generate Parent Budget

- Parent Budget Transaction Type
- Default Entry Date
- 3 In Write-Back Rule, define write-back filters that identify the portions of budget data to load to PeopleSoft.
- 4 Optional: You can delete previous write-back rules by selecting Delete on the Execute Rules Options screen.
- 5 Run the write-back rule.

If you have selected a transaction type of "post," the rules are validated, the budget is written to Commitment Control interface tables and the posting process is initiated after writing the data. Any errors are forwarded to Public Sector Planning and Budgeting.

Deleting Write Back Member Mappings

You can delete the member mappings created in a session, and restore member mappings based on the point of view.

- To delete member mappings:
- 1 From the Write Back Mapping task bar, select Delete Mappings.
- When prompted with Are you sure you want to delete the selected data load mapping(s)?, click Yes.

Managing Write-Back Rules

Subtopics

- Editing Write-Back Rules
- Running Write-Back Rules
- Using the Write-Back Workbench
- Checking the Data Write-Back Rule Status
- Deleting Data Write-Back Rules

You can perform the following tasks to manage your data write-back rules:

- Edit data write-back rules. See "Editing Write-Back Rules" on page 226.
- Run data write-back rules. See "Running Write-Back Rules" on page 226.
- Check the status of data write-back rules. See "Checking the Data Load Rule Status" on page 167.
- Delete data write-back rules. See "Deleting Data Write-Back Rules" on page 235.
- View write-back rules before executing them—See "Using the Data Load Workbench" on page 168.
- Check the data write-back rule process details. See "Viewing Process Details" on page 180.

Editing Write-Back Rules

After you create data write-back rules, you can edit the rules as necessary.

Note: You cannot edit the Plan Type or Location after the rule is run.

- To edit data write-back rules:
- 1 From the Workflow tab, and then under Write Back, select Write-Back Mapping.
- 2 Enter the **Location Name** or click to select the location name.
- 3 Select the data write-back rule.
- 4 Modify the source filters or target options as necessary, and then click Save.

Running Write-Back Rules

After you create a data write-back rule, you must run the rule to write back the data from the EPM Planning application to the general ledger system.

- To run a data write-back rule:
- 1 Select the data write-back rules.
- 2 Click Execute.

After the extraction process is successful, you must log in to Oracle General Ledger or PeopleSoft General Ledger and import the budget data. See "Loading Data into Oracle E-Business Suite" on page 235.

3 To load data from the source EPM application, select Import from Source.

Select this option to review the information in a staging table, before exporting directly to the target general ledger system.

When you select "Import from Source," FDMEE imports the data from the EPM target application, performs the necessary transformations, and exports the data to the FDMEE staging table.

For information on staging tables, see "Staging Tables Used for Import from Source" on page 357.

4 To export data to the target general ledger system, select Export to Target.

FDMEE transfers the data into the general ledger system.

5 Click Run.

Note: After the rule successfully runs, you can view the information in the staging table. See "Staging Tables Used for Import from Source" on page 357.

After the rule runs, perform the required tasks in your general ledger system. See "Loading Data into Oracle E-Business Suite" on page 235 and "Loading Data into PeopleSoft Enterprise Financial Management" on page 236.

Using the Write-Back Workbench

The Write-Back Workbench provides the functionality from which you can review and verify the data to be written back to the ERP system.

Key features of the Write-Back Workbench include

- Interactive Load Process with options for Import and Export
- Provision to view Source (All) / Source (Mapped) / Target / Source and Target values
- PTD / YTD Value display for ready reference
- Display options for Valid, Invalid, Ignored and All Data
- Online and Offline Load process

Note: When you login with the Run Integration role, these links are visible in the Tasks pane: Data Load Workbench, Data Load, Member Mapping, HR Data Load, Metadata, and Process Detail.

The Workbench consists of three sections:

- POV Bar
- Status
- Data Grid

Using the POV Bar

For the Write Back Load Workbench, the POV bar shows the current:

- Location
- Period
- Category

Location KS7DIM_EBSTB Period Jan-08 Category Actual Rule KS7DIM_EBSTB Source File Target KS7DIM

By default only the data rule assigned to the Category POV is displayed.

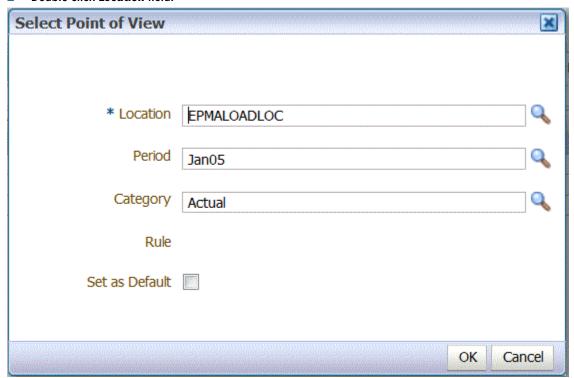
For the Write-Back Workbench, the POV only shows the current "Location".

The Source System and Target Application are displayed as context information.

Selecting the Location POV

When users log on, the location POV is set to the default location of the user. Users can view only locations that they have permission to access. Location permission is determined and granted by system administrators.

- ➤ To select another Location POV:
- 1 From the Workflow tab, and then under Data Load, select Write Back Workbench.
- 2 Double click Location field.



- 3 In Select Point of View, and then in Location, enter a full or partial string for the new location, and then, click OK.
- 4 **Optional**: To search on another location, from the **Location** drop-down, click **More**, navigate to the location on the **Search and Select: Location** screen, and then, click **OK**.
- 5 Optional: In Select Point of View, select Set as Default to use the new location as the default location.
- 6 Click OK.

Setting the Period POV

- To select another Period POV:
- 1 From the Workflow tab, and then under Data Load, select Write Back Workbench.
- 2 Double click Location field.
- 3 In Select Point of View, and then in Period, enter a full or partial string for the new period, and then, click OK.

- 4 **Optional**: To search on another period, from the **Period** drop-down, click **More**, navigate to the period on the **Search and Select**: **period** screen, and then, click **OK**.
- 5 Optional: In Select Point of View, select Set as Default to use the new period as the default period.
- 6 Click OK.

Setting the Category POV

Note: By default only the data rule assigned to the Category POV is displayed. If the Category POV is locked, only the data rule assigned to the category is displayed.

- To select another Period POV:
- 1 From the Workflow tab, and then under Data Load, select Write Back Workbench.
- 2 In Write Back Load Workbench, double click Location field.
- 3 In Select Point of View, and then in Category, select the new category, and then, click OK.
- 4 **Optional**: In **Rule**, select the rule assigned to the Category POV.
- 5 Select **Set as Default** to use the new category as the default category.
- 6 Click OK.

Using the Workbench Data Grid

The data grid includes two tabs in the main grid:

- Load Data/Write-back Load (POV)—Use this tab to import, view and verify and export data from source systems.
- Query Data—Use this tab to display historical data based on the Period and Categories POVs. The Period and Category are specified in the in the Table Action.

You perform tasks on the data grid by selecting options on the Table Action including:

- "Viewing Data" on page 174
- "Formatting Data" on page 175
- "Showing Data" on page 176
- "Opening Loaded Data in Microsoft Excel" on page 177
- "Querying by Example" on page 178
- "Freezing Data" on page 178
- "Detaching Data" on page 178
- "Wrapping Text" on page 179

Viewing Data

The following View drop-down options provides multiple ways to view data.

Table 35 View Options and Descriptions

View Option Description Customizes views. Options include: View - Table—Selects the source or target data to display in the grid including: Source (All)—Shows both mapped and unmapped source dimensions (ENTITY, ACCOUNT, UD1, UD2,... AMOUNT). Source (Mapped)—Shows only mapped source dimensions. Target—Shows only target dimensions (ENTITYX, ACCOUNTX, UD1X, UD2X,....AMOUNTX). Source and Target—Shows both source and target dimensions (ENTITY, ENTITYX, ACCOUNT, ACCOUNTX, UD1, UD1X, AMOUNT, AMOUNTX). • Columns—Selects the columns to display in the data grid including: Show All Entity Account Version Product Department STAT Amount PTD Amount YTD Amount • Freeze/Unfreeze—Locks a column in place and keeps it visible when you scroll the data grid. The column heading must be selected to use the freeze option. To unfreeze a column, select the column and from the shortcut menu, select Unfreeze. • Sort- Use to change the sort order of columns in ascending or descending order. A multiple level sort (up to three levels and in ascending and descending order) is available by selecting Sort, and then Advanced. From the Advanced Sort screen, select the primary "sort by" column, and then the secondary "then by" column, and then the third "then by" column. The search fields that display in the advanced search options differ depending on what artifact you are selecting. • Reorder Columns—Use to change the order of the columns. When you select this option, the Reorder Columns screen is displayed. You can select a column and then use the scroll buttons on the right to change the column order. • Query by Example—Use to toggle the filter row. You can use the filter row to enter text to filter the rows that display for a specific column. You can enter text to filter on, when available, for a specific column, and then press [Enter]. To clear a filter, remove the text to filter by in the text box, then press [Enter]. All text you enter is case sensitive.

Formatting Data

You can resize the width of a column either by the number pixel characters or a percentage. You can also wrap text for each cell automatically when text exceeds the column width.

- To resize the width of a column:
- Select the column to resize.

2 From the table action bar, select Format, and then Resize.

The Resize Column screen is displayed.

3 In the first Width field, enter the value to resize by.

You can select a column width from 1 to 1000.

- 4 In the second Width field, select either pixel or percentage as the measure to resize by.
- 5 Select OK.
- To wrap the text of a column:
- 1 Select the column with the text to wrap.
- 2 From the table action bar, select **Format**, and then **Wrap**.

Showing Data

You can select the type of data to display in the data grid including:

- Valid Data—Data that was mapped properly and is exported to the target application.
- Invalid Data—One or more dimensions that was not mapped correctly and as a result, the data is not exported to target.
- Ignored Data—User defined explicit IGNORE maps to ignore while exporting to target.
 IGNORE maps are defined in the member mapping by assigning a special target member of IGNORE.
- All Data—Shows all valid, invalid and ignored data.
- To show a type of data, select **Show**.
- 1 Select Show.
- 2 Choose either Valid Data: Invalid Data, Ignored Data, or All Data.

Selecting Write-Back Rules

- To select another write-back rule:
- 1 From the Write-Back Rule drop-down, select another write-back rule associated with the location POV.
- 2 Click OK.

Importing from a Planning or Essbase Cube Source

The Import from Source feature enables the Oracle Data Integrator to import the data from a Planning or Essbase Cube source system, performs the necessary transformation, such as import, and map and validate the data. The Import from Source features also allows you to import the source either online (immediate processing) or offline (runs in background).

Select this feature only when:

- You are running a data load rule for the first time.
- Your data in the source system changed. For example, when you reviewed the data in the staging table after the export, and it was necessary to modify data in the source system.

In many cases, source system data may not change after you import the data from the source the first time. Therefore, it is not necessary to keep importing the data when it has not changed.

- To submit the data load rule:
- 1 From the table action bar, select the **Data Rule** drop-down, and choose the data load rule.
- 2 Click

The Execution Mode Option screen is displayed.

3 From the Execution Mode drop-down, select the mode of importing the source.

Execution modes include:

- online—ODI processes the data in sync mode (immediate processing).
- offline—ODI processes the data in async mode (runs in background).

Click to navigate to the Process Detail page to monitor the ODI job progress.

- 4 Click OK.
- 5 After you run a data rule, check the status of the import on the POV bar.

Exporting to Target

Use the Export to Target feature to export data to a target application, which is the ERP application. Select this option after you have reviewed the data in the data grid and need to export it to the target application.

Export options:

- Current Dimension
- All Dimensions
- Export to Excel

When exporting data for Essbase, you can store, add, and subtract data.

When exporting data for Planning, the following options are available:

- Store Data—Inserts the data from the source or file into the target application. This replaces any value that currently exists.
- Add Data—This adds the value from the source or file to the value that exists in the target application. For example, when you have 100 in the source, and 200 in the target, then the result is 300.

- Subtract Data—This subtracts the value in the source or file from the value that exists in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
- Override All Data—Clears all data in the target, and then loads from the source or file. For
 example when you have a year of data in your planning application, but are only loading a
 single month, this option clears the entire year before performing the load.
- To submit the data load rule:
- 1 From the table action bar, select the **Data Rule** drop-down, and choose the data load rule.
- 2 Click .

The Execution Mode Option screen is displayed.

3 From the Execution Mode drop-down, select the mode of exporting to the target.

Execution modes include:

- online—ODI processes the data in sync mode (immediate processing).
- offline—ODI processes the data in async mode (runs in background).

Click to navigate to the Process Detail page to monitor the ODI job progress.

4 From the Export drop-down, select the export method: either Current Dimension or All Dimensions.

Export options:

- Current Dimension
- All Dimensions
- Export to Excel
- For **Current Dimension** and **All Dimensions** export methods, in **Select file location**, navigate to the file to export and click **OK**.

For the **Export to Excel** method, mappings are export to a Microsoft Excel spreadsheet.

- 6 Click OK.
- 7 After you run a data rule, you can check the status of the export on the POV bar.

Opening Write Back Rules

When reviewing data in the workbench, users can drill down from the amount to ERP source system. In the source system the data is displayed in the granularity with which it was loaded.

You can open the data load rules for a source system application in Microsoft Excel and review how the data is defined.

To open the data load rules, from the table action bar, click

You are prompted to open the rule in Microsoft Excel. You can specify another program in which to open the rule by selecting the "Open with" option and selecting the program.

Querying by Example

Use the Query by Example feature to filter rows that display for a specific column. You can enter text to filter on, when available, for a specific column, and then press [Enter]. To clear a filter, remove the text to filter by in the text box, then press [Enter]. All text you enter is case sensitive.

- To query by example:
- 1 From the table action bar, click lacksquare to enable the filter row.

The filter row must appear above the columns to use this feature.

2 Enter the text by which to filter the values in the column and press [Enter].

Note: When entering text to filter, the text or partial text you enter is case sensitive. The case must match exactly. For example, to find all target applications prefixed with "HR," you cannot enter "Hr" or "hr."

Freezing Data

Use the Freeze feature to lock a column in place and keeps it visible when you scroll the data grid.

- To freeze a column:
- 1 Select the column to freeze.
- 2 from the table action bar, click **111**.
- To unfreeze a column:
- 1 Select the frozen column.
- 2 On the shortcut menu, select **Unfreeze**.

Detaching Data

Use the Detach feature to detach column from the data grid, When you detach the grid, columns display in their own window. To return to the default view, select *View*, and then click *Attach* or click the Close button.

- ➤ To detach columns:
- 1 Select the column.
- 2 From the table action bar, click
- ➤ To reattach columns
- 1 Select the table.
- 2 From the table action bar, select View, and then, select Attach.

Wrapping Text

You can wrap text for each cell automatically when text exceeds the column width.

- To wrap text for a column:
- 1 Select the column.
- 2 Click 🔑.

Checking the Data Write-Back Rule Status

After you run a data write-back rule, you can check the status on the Process Details page. See "Viewing Process Details" on page 180.

Tip: You can also check the status of the rule in Oracle Data Integrator.

Deleting Data Write-Back Rules

- To delete data write-back rules:
- 1 From the Workflow tab, and then under Write Back, select, Write-Back Mapping.
- 2 Enter the **Location Name** or click to select the location name.
- 3 Select the data write-back rule.
- 4 Click Delete.

Loading Data into Supported Source Systems

Subtopics

- Loading Data into Oracle E-Business Suite
- Loading Data into PeopleSoft Enterprise Financial Management

To complete the data write-back process, you must perform required tasks in your supported source system.

Loading Data into Oracle E-Business Suite

If the target source system is Oracle E-Business Suite, you must import the budget data using Oracle General Ledger.

Refer to the Oracle Fusion and E-Business Suite user documentation for detailed information. At a high level, follow this process:

- 1. Launch Oracle General Ledger.
- 2. Run the Journal Import process.

If you select the Create Journal option:

- Data is extracted to the journal interface table (GL_INTERFACE).
- FDMEE stores the process ID in the GROUP_ID column. You can check the data that was created by filtering on the process ID.
- If no errors occur, a journal entry is created.

If you clear the Create Journal option:

- Data is extracted to the budget interface table (GL_BUDGET_INTERFACE).
- Carefully select Budget Name (BUDGET_NAME) and Budget Organization (BUDGET_ENTITY_NAME) when loading data into the GL_BUDGET_INTERFACE table.
- If no errors occur, the budget balances are posted.
- 3. Review the journal or budget balances.

Loading Data into PeopleSoft Enterprise Financial Management

If the target is PeopleSoft, you must import the budget data using PeopleSoft General Ledger.

Refer to the PeopleSoft Enterprise Financial Management user documentation for detailed information. At a high level, follow this process:

- 1. Launch PeopleSoft General Ledger.
- 2. Run the Generate Journals process.
- 3. When generating a journal request, define these required options:
 - Accounting Definition Name—Select HYPDEFN
 - Template—Select HYP_STDBUD

The data is staged in the PS_HPYPB_ACCT_LN interface table. FDMEE stores the process ID in the JRNL_LN_REF table. You can check the data that was created by filtering on the process ID.

4. Run the process.

If the process runs without errors, the journals are created.

5. Review the journal, create balancing entries, and then post the journal.

7

Loading Human Resources Data

In This Chapter

Understanding Human Resource Data Integration	237
Requirements	238
Public Sector Planning and Budgeting Dimensions	239
Smart Lists	240
Using Loaded Data in Public Sector Planning and Budgeting	240
Defining Human Resource Data Load Rules	240
Managing Human Resource Data Load Rules	245

Understanding Human Resource Data Integration

FDMEE supports loading human resource data from your PeopleSoft Human Capital Management source system into Public Sector Planning and Budgeting applications.

You load human resource data by creating mappings in FDMEE to map the data from your PeopleSoft Human Capital Management source system to Public Sector Planning and Budgeting accounts.

FDMEE supports loading human resource data into Public Sector Planning and Budgeting applications. At this time, no other application types are supported.

If you have made revisions to properties in your Public Sector Planning and Budgeting application, loading human resource data from your source system into the application automatically overrides any modified application properties.

FDMEE supports loading data into the following Public Sector Planning and Budgeting models:

- Employee
- Position
- Employee and Position (both)

For a high-level process overview of the human resource integration, see "Extracting Human Resource Data" on page 29.

Requirements

Before you begin your human resource integration, you must complete the following prerequisites:

- Run specific processes in PeopleSoft Human Capital Management. See "Running Processes in PeopleSoft Human Capital Management" on page 238.
- Specify parameters in Planning to enable data to be loaded into Essbase. See "Defining the Data Load Settings in Planning" on page 238.

Running Processes in PeopleSoft Human Capital Management

To load future-dated salary data from PeopleSoft Human Capital Management, perform these steps in PeopleSoft Human Capital Management.

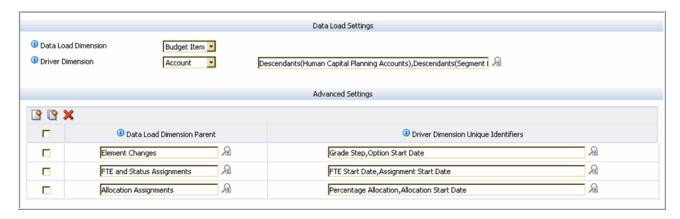
Refer to the PeopleSoft Human Capital Management user documentation for detailed information.

- Run the Copy Job Data process (BPJBCOPY) to put a copy of data in PS_JOB,
 PS_COMPENSATION and PS_JOB_EARNS_DIST in PS_BP_JOB,
 PS_BP_COMPENSATION and PS_BP_JOB_ERN_DIST. Copies of these tables are used
 instead of human resource tables to ensure that the step increment additions to the data
 only occur in Public Sector Planning and Budgeting tables and do not affect current human
 resource data when you run the step increment process.
- 2. Run the optional Load Step Increment (BPCMP107) process for budgets to generate future-dated job data in PS BP JOB and PS BP JOB ERN DIST.

Defining the Data Load Settings in Planning

In Planning, you must define the data load settings before you can use FDMEE to extract metadata and data from PeopleSoft Human Capital Management.

Data load settings enable you to choose the primary keys based of driver dimension members selected. You can specify other members that are not included in the selected driver dimension members. The data values of these members selected from the driver dimension as primary key are used to uniquely identify a row while finding the next available member from children of the selected member on the left hand side. The following figure shows the Data Load Settings window in Planning.



See Oracle Hyperion Planning Administrator's Guide for detailed information.

Public Sector Planning and Budgeting Dimensions

The following Public Sector Planning and Budgeting dimensions are populated by FDMEE:

- Position
- Employee
- Element
- Job Code
- Entity

These dimensions are associated with the following tables:

Dimension	Database Table
Position	AIF_PS_POSITION_DATA_STG
Employee	AIF_PS_JOB_STG
Job Code	AIF_PS_JOBCODE_TBL_STG
Element	AIF_PS_SAL_PLAN_TBL_STG
	AIF_PS_SAL_GRADE_TBL_STG
	AIF_PS_SAL_STEP_TBL_STG
	AIF_PS_EARNINGS_TBL_STG
	AIF_PS_BENEF_PLAN_TBL_STG
Entity	AIF_PS_DEPT_TBL_STG

For a complete list of Public Sector Planning and Budgeting tables, see Appendix D, "Staging Tables."

Smart Lists

Human Resource data such as salary information, union codes, and status are Smart Lists in Public Sector Planning and Budgeting applications. FDMEE automatically recognizes Smart Lists and populates the data accordingly.

FDMEE allows you to assign a prefix to Smart Lists in the Compensation Allocation point of view (POV). For general ledger integrations, you create metadata mappings and can optionally define member prefixes. For human resource integrations, you can optionally assign a Smart List prefix in the rule line mapping definition. You should ensure that member prefixes (used in a general ledger metadata mapping) are identical to Smart List prefixes (used in a human resource data rule mapping). For information on human resource rule mappings, see "Creating Mapping Definitions" on page 243. For information on general ledger metadata mappings, see "Defining Metadata Rules" on page 118.

For information about using Smart Lists in Planning and Public Sector Planning and Budgeting, see the *Oracle Hyperion Planning Administrator's Guide* and the *Oracle Hyperion Public Sector Planning and Budgeting User's Guide*.

Using Loaded Data in Public Sector Planning and Budgeting

After loading human resource data, perform these tasks:

- For Classic Planning applications, in Planning, refresh the application database as required. See the *Oracle Hyperion Planning Administrator's Guide*.
- For Planning applications administered in Performance Management Architect, navigate to the Application Library and redeploy the application. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.
- To view the data and use it as part of Public Sector Planning and Budgeting, see the *Oracle Hyperion Planning Administrator's Guide* and the *Oracle Hyperion Public Sector Planning and Budgeting User's Guide*.

Defining Human Resource Data Load Rules

Subtopics

- Defining the Rule Details
- Creating Rule Line Definitions by Copying
- Creating Rule Line Definitions by a Template Mapping
- Creating Rule Line Definitions
- Creating Mapping Definitions

At a high level, creating human resource data load rules that enable you to populate data and metadata in Public Sector Planning and Budgeting applications involves these steps:

- Create the human resource data rule and define the rule details. See "Defining the Rule Details" on page 241.
- Create the rule line definitions and mappings manually or by running a script. See "Creating Rule Line Definitions by a Template Mapping" on page 242 and "Creating Rule Line Definitions" on page 243.
- "Creating Mapping Definitions" on page 243

Before you begin, ensure that you have performed all required tasks described in Chapter 7, "Loading Human Resources Data."

- "Running Processes in PeopleSoft Human Capital Management" on page 238
- "Defining the Data Load Settings in Planning" on page 238

Before you create a human resource data load rule:

- 1. Register your human resource source system.
- 2. Register your target Public Sector Planning and Budgeting application.
- 3. Review the selected Source Accounting Entities.
- 4. Define import formats and locations.

For a high-level process overview, see "Extracting Human Resource Data" on page 29.

Defining the Rule Details

- To define rule details:
- 1 From the Workflow tab, and then under HR Data Load, select HR Data Load.
- 2 Enter or click to select the **Location**.
- 3 In the Rule Name area, click Add.
- 4 In the Rule Definition area, enter the Rule Name.
- 5 Enter or click to select the value of the **Scenario** dimension you identified when you registered the application. This enables you to classify and collect time-specific data.
- 6 Optional: Enter a Description.
- 7 Select the Salary Assignment Option to specify the salary assignment details to load by the database table.

Salary Assignment Options:

- Standard—Load data from PS_JOB table that is live on a given "As of Date."
- Auto Increment—Load data from PS_BP_JOB table that is either live on a given "As of Date" or is later than it.

Note: To use this option, perform these steps in "Running Processes in PeopleSoft Human Capital Management" on page 238.

- Standard Include Future Dated—Load data from PS_JOB table that is either live on a given "As of Date" or is later than it.
- 8 Enter or click to select the member of the **Version** dimension.
- 9 Click Save.
- 10 Next, create the rule line definition.

You have three options to create the rule line definition:

- Copy a rule—See "Creating Rule Line Definitions by Copying" on page 242.
- Use a template to create the rule line definitions—See "Creating Rule Line Definitions by a Template Mapping" on page 242.
- Create the rule line definitions manually—See "Creating Rule Line Definitions" on page 243.

Creating Rule Line Definitions by Copying

You can create a rule line definition by copying a human resource data rule. Copying a rule copies all the POV rules and their mapping details.

Note: You cannot copy rules across applications.

- To create rule line definitions by copying:
- 1 Perform the steps in "Defining the Rule Details" on page 241.
- In the Rule Definition area of the HR Data Load page, select Create Mapping, and then Copy from another rule.

The Copy Rule dialog box is displayed.

- 3 Select the source rule.
- 4 Click Copy.
- 5 Click OK.

The POV rules and their mapping details are displayed in the Rule Line area.

6 Edit the mappings as necessary. See "Creating Mapping Definitions" on page 243.

Creating Rule Line Definitions by a Template Mapping

You can use a template mapping to automatically create rule line definitions for Public Sector Planning and Budgeting dimensions.

- To create rule line definitions using a template:
- 1 Perform the steps in "Defining the Rule Details" on page 241.
- 2 Select Create Mapping, and then Create Template Mapping.
- 3 Select the Language.
- 4 Click OK.
- 5 Click OK.

The Rule Line Definition area is populated with POVs from the template.

- 6 Edit the rule line definitions or add new rule line definitions as necessary. See "Editing Human Resource Data Load Rules" on page 245.
- 7 Run the data load rule. See "Running Human Resource Data Load Rules" on page 245.

Creating Rule Line Definitions

You create rule line definitions to map dimension members to a Planning account.

- To create rule line definitions:
- 1 In the Rule Line area, click 🛨
- 2 In **POV Name**, enter a descriptive name for the point of view.
- 3 In POV Type, select the set of dimensions:
 - Position Property—Position dimension member properties
 - Employee Property—Employee dimension member properties
 - **Position and/or Employee Assignment**—Property and salary data associated with positions and employees.
 - Compensation Allocation—Allocation of salary costs to general ledger accounts for positions and employees.
 - **Jobcode Property**—Available when you created an Employee Only application in Planning, this dimension provides position and employee member properties.
 - Salary Grade—Rates for Element dimension members that correspond to salary grades.
 - Salary Step—Rates for Element dimension members that correspond to salary steps.
- 4 Define the mappings for the POV. See "Creating Mapping Definitions" on page 243.
- 5 Repeat steps 1 step 3 through for each POV to define.

Creating Mapping Definitions

You define mapping definitions to specify which Public Sector Planning and Budgeting accounts to load with the extracted human resource data. For each dimension in the point of view, you must select a member, and then map the column in the source system to a Planning account.

- ➤ To create mapping definitions:
- 1 In the Rule Line area, click to add a rule line mapping.

The Rule Line Mapping page is displayed.

2 In the Rule Line Point of View area, enter or click to search for a member and map the members for dimensions in the point of view (POV).

You must enter a member for:

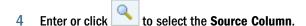
Budget Item—For Budget Item dimensions, you may want FDMEE to automatically
create the budget item values. This option is available only for the Unspecified Budget
Member. You must select a parent member when you plan to select the optional Auto
create flag setting.

Note: The parent member that you select must match what you selected in the Oracle Hyperion Planning Data Load Settings window. See "Defining the Data Load Settings in Planning" on page 238.

- Year
- Period

Selecting a member for all other dimensions is optional. If you do not select a member for a dimension, FDMEE loads the values as-is from the source system.

3 In the Rule Line Mapping area, define the source column to account mappings. To add a mapping, click



You select the source column in the PeopleSoft Human Capital Management table that contains the human resource data to load. For example, when you are creating a rule line mapping for Salary Grade, select the source column "Grade" in the PeopleSoft human resource table. The source columns that display are specific to the POV Type that you selected when you created the rule line definition.

5 Enter or click to select the **Account**.

You select the Public Sector Planning and Budgeting account into which to load the extracted human resource data. For example, to load employee names and IDs, select Employee Name and Employee Number. For a list of all tables and columns, see Appendix D, "Staging Tables."

The Data Type and Smart List Name fields are automatically populated based on the Account you selected.

Note: Several Smart Lists in Public Sector Planning and Budgeting applications are paired (for example, Account Segment and Account Segment Description). When you map from the source system to a paired Smart List, map one of the paired Smart List members (for example, Account Segment, but not Account Segment Description).

- 6 Click Save.
- 7 Click Back to return to the HR Data Load page.
- 8 Next, run the data load rule. See "Editing Human Resource Data Load Rules" on page 245.

Managing Human Resource Data Load Rules

Perform the following tasks to manage your data rules:

- Edit data load rules—Typically, when you used a script to create the rule line definitions, you may want to add or edit the mappings. See "Editing Human Resource Data Load Rules" on page 245.
- Run data load rules. See "Editing Human Resource Data Load Rules" on page 245.
- Delete data load rules. See "Deleting Human Resource Data Load Rules" on page 247.
- Check the data rule process details. See "Checking the Human Resource Data Load Rule Status" on page 247.

Editing Human Resource Data Load Rules

- To edit human resource data load rules:
- 1 From the Workflow tab, and then under HR Data Load, select HR Data Load.
- 2 Enter or click to select the **Location**.
- 3 Select the rule to modify and change the rule details, rule line definition, and mapping settings. See "Editing Human Resource Data Load Rules" on page 245.
- 4 Click Save.

Running Human Resource Data Load Rules

To extract and load data from PeopleSoft Human Capital Management, run the human resource data load rule that you defined.

When you run a human resource data load rule, you have several options.

• Import from Source—FDMEE imports the data from the source system, performs the necessary transformations, and loads the data to the FDMEE staging table.

Select this option only in these situations:

O You are running a data load rule for the first time.

O Your data in the source system changed. For example, when you reviewed the data in the staging table after the export and it was necessary to modify data in the source system.

In many cases, your source system data may not change after you import the data from the source the first time. Therefore, you need not import the data.

• Export to Target—Exports the data to the target application.

Select this option after you have reviewed the data in the staging table and you need to export it to the target application.

Note: Select both options only in cases where you want to load the data directly into the target application.

- To run human resource data load rules:
- 1 On the HR Data Load page, select the rule.
- 2 Click Execute.
- 3 Enter or select the As of Date from which to load data from the human resource source system.
- 4 To extract data from the source system, select Import from Source.
- To seed loaded data into your target Public Sector Planning and Budgeting application, select Export to Target.

Selecting export to target exports the data to the target application.

- 6 If you selected **Export to Target**, perform these actions:
 - a. To load data and/or metadata into a Classic Public Sector Planning and Budgeting application: Select the **Load Type**:
 - Data—Loads only the data.
 - Metadata—Loads only the Public Sector Planning and Budgeting metadata.
 - Both—Loads the data and metadata.

Note: Choose either Metadata or Both as the load type on a new application; otherwise the data load fails.

- b. To load data and/or metadata into a Performance Management Architect Public Sector Planning and Budgeting application:
 - i. Select Metadata and run the data load rule.
 - ii. After the rule runs successfully, in Oracle Hyperion EPM Architect, deploy the application.
 - iii. Return to FDMEE. On the HR Data Load page, locate the rule.
 - iv. Click Execute to run the data load rule, and then select Data.
- c. Select the **Department Load Option** to indicate whether to load all or specific departments and department data in the business unit:

- All—Load data for all departments to the target application.
- Selected—Load data for departments that you select to the target application. You can press the [Ctrl] key and select multiple departments.

Note: FDMEE merges data and does not "replace" balance data in a target application.

7 Click Run.

Data is loaded into your Oracle Hyperion Public Sector Planning and Budgeting accounts.

Deleting Human Resource Data Load Rules

When you delete a human resource data rule, only the rule is removed. The extracted data or dimensions are not removed. You cannot delete a data rule when it is running.

- To delete human resource data rules:
- 1 From the Workflow tab, and then under HR Data Load, select HR Data Load.
- 2 Enter or click to select the **Location**.
- 3 Select the rule to remove and then click **Delete**.

The Delete Confirmation dialog box is displayed.

4 Click Yes.

The rule is deleted.

Checking the Human Resource Data Load Rule Status

- To check data load rule status:
- 1 Select the Task pane.
- 2 Select Process Details.

ee "Viewing Process Details" on page 180.

Note: See the aif_c: \Oracle \Middleware\user_projects\epmsystem1\tmp\aif_2548.log) to view details of any rows rejected during the export step.

Batch Processing

8

In This Chapter

Working with Batch Definitions	250
Adding a Batch Group	254
Executing Batches	255
Using Open Batches	255
Scheduling Jobs	258
Working with Batch Scripts	260

Using the FDMEE batch processing feature, you can:

- Combine one or more load rules in a batch and execute it at one time.
- Run jobs in a batch in serial or parallel mode.
- Define the parameters of the batch.
- Derive the period parameters based on POV settings.
- Create a "master" batch that includes multiple batches with different parameters.
 - For example, you can have one batch for metadata rules run in serial mode, and a second batch for the data rule run in parallel mode.
- Associate a batch with a batch group for ease of use and security.
- Instruct the batch to submit included jobs in parallel mode and return control.
- Instruct the batch to submit included jobs in parallel mode and return control only when *all* jobs are complete.

Using the FDMEE batch processing feature, you can:

Batch processing options are available on the FDMEE task pane, or by executing batch scripts.

If you process batches from the FDMEE task pane, use the Batch Definition option to create a batch, and specify its parameters and tasks included in the batch. See "Working with Batch Definitions" on page 250. Use the Batch Execution option to execute batches. See "Executing Batches" on page 255.

Note: FDMEE batch load features are not available to the Account Reconciliation Manager. For Accounts Reconciliation Manager load and scheduling features, see the *Oracle Hyperion Financial Close Management User's Guide*.

Working with Batch Definitions

A batch definition is used to define the batch jobs and parameters, and the type of rules included in the batch. A batch can contain one type of rule only. Valid types of rules are:

- metadata
- write-back
- data
- batch
- open batch

Note: Only an administrator can create batch definitions.

The Batch Definition features consist of three regions:

- Batch Definition detail—Enables you to add and delete a batch definition. If adding or modifying a definition, specify the definition name, target application, process method, return control method, and wait parameters.
- Batch Definition parameters—Enables you to derive period parameters based on the Import to Source, Export to Target, POV period, and to indicate data extract parameters. The parameter definition is not available for the batch types "batch" and "metadata (rules)".
- Batch Definition jobs—Enables you to add and delete jobs to a batch. Based on the type of batch, a specific types of rules are allowed.
- To add a batch definition:
- 1 From the **Setup** tab, and then under **Batch**, select **Batch Definition**.
- 2 In the Batch Definition summary section, click Add.

Use the blank Name and Target Application fields in the Batch Definition summary grid to enter a batch name or target application on which to search.

- 3 In Batch Definition detail section, select the Definition tab.
- 4 In Name, specify the name of the batch definition.

The name must contain only alpha, numeric or underscore characters. Do not enter spaces or any other character.

- 5 From the Target Application drop-down, select the name of the target application.
- 6 In the Type drop-down, select the type of rule for the definition.

Available types are:

- metadata
- data
- batch
- open batch

The Open Batch type is used only for file-based data sources and does not contain any batch jobs. When you run this type of batch, the process reads the files automatically from the openbatch directory and imports them into appropriate POV based on the file name. When the open batch is run, the master folder is emptied.

- 7 In the **Execution Mode** drop-down, select the batch process method.
 - Serial—Processes files sequentially, requiring that one file complete its process before the next file starts its process.
 - Parallel—Processes files simultaneously.
- 8 For batch processing run in parallel mode, complete the following fields
 - **Wait for Completion**—Select **Wait** to return control only when the batch has finished processed.
 - Select *No Wait* to run the batch in the background. In this case, control is returned immediately.
 - **Timeout**—Specify the maximum time the job can run. FDMEE waits for the job to complete before returning control.
 - The Timeout can be in seconds or minutes. Enter a number and followed by a **S** for seconds or **M** for minutes.
- 9 In Open Batch Directory for an open batch type, specify the folder under inbox\batches openbatch where the files to be imported are copied. If this field is blank or null, all files underinbox\batches\openbatch are processed.
- 10 In File Name Separator for an open batch, select the character to use when separating the five segments of an open batch file name.

Options:

- ~
- @
- :
- •
- 11 Select Auto Create Data Rule to create the data rule automatically for file-based data loads.

Note: The Auto Create Data Rule option is available when the rule type is "open batch".

When FDMEE assigns the data rule name, it checks when a data rule with name the "Location_Category" exists. If this name does not exist, FDMEE creates the data rule using the file name convention:

- Rule Name—Location_Category
- Description—"Auto created data rule"
- Category—Category
- File Name—Null

- Mode—Replace
- 12 Optional: In the Description field, enter a description of the batch definition.
- 13 Click Save.
- 14 Optional: From the Batch Group drop down, select the batch group to associate with the batch.
- 15 Optional: In Before Batch Script, enter the name of the custom script to run before the batch.

You can browse and select, or upload a custom script on the Select screen by clicking The custom script can be written in either Jython or Visual Basic script.

16 Optional: In After Batch Script, enter the name of the custom script to run after the batch.

You can browse and select, or upload a custom script on the Select screen by clicking ...

The custom script can be written in either Jython or Visual Basic script.

Note: Before and After batch scripts are not executed for a batch or batches. You must add the script to the individual batches with the master batch script.

- 17 Click Save.
- To add batch definition parameters:
- 1 From the **Setup** tab, and then under **Batch**, select **Batch Definition**.

Note: Batch parameters are not available for a "batch" type.

- In Batch Definition, and then under Batch Definition Detail, select the Parameter tab.
- In **Parameters**, select **Import From Source** to import the data from the source system, perform the necessary transformations, and export the data to the FDMEE staging table.
- 4 Select **Export To Target** to export the data to the target application.
- 5 Select POV Period to derive the period parameters based on POV settings.

If the POV period is selected, the period key start and end period fields are disabled.

This field is only available for a data load batch.

When setting up a batch, you can either choose the POV to drive the period, or enter the periods explicitly. If you select the POV, it is pulled from the Default POV Period setting in System/Application, or User settings.

Specify dates in the Start Period and End Period to derive period parameters through which the data is processed.

Use the date format based on the locale settings for your locale. For example, enter the date using the format MM/DD/YY.

If the Start Period and End Period fields are selected, the POV Period field is disabled.

This field is only available for a data load batch.

7 In the Import Mode drop-down, select the mode to extract data all at once for an entire period or incrementally during the period.

Option are:

• **Snapshot**—Extracts everything for the selected source set for an entire period.

Note the following behavior of Snapshot mode:

- When the source data for the selected period *has never been run*, data is extracted from the source.
- When the source data for the selected period has been run, data is extracted from the FDMEE staging table, and not from the source. This means that if a user has two locations that extract from the same ERP source, data is extracted from the source only once.

For example, if you load data to Financial Management from the EBS for a selected period, but you run the integration to ARM for the same source and period, use what is in the interface tables since data is only extracted the first time from the source. This result in a significant performance gain for any subsequent data loads. The first extraction take the longest, but any other subsequent extractions are fast.

- **Incremental**—Extracts those records that were added after the prior data extract.
- **Full Refresh**—Performs a clean extraction from the source system, thereby clearing any existing data rows in the appropriate FDMEE staging tables for a given source Business Unit and source period.
- **Append**—(File-based source systems) Existing rows for the POV remain the same, but new rows are appended to the POV (that is, appends the new rows in TDATASEG)
- **Replace**—(File-based source systems) Replaces the rows in the POV with the rows in the load file (that is, replaces the rows in TDATASEG).

Note: The import mode options (Snapshot, Incremental and Full Refresh) are only applicable to Data Rules in a Location using a Standard Import Format. Data Rules in a Location with a Source Adapter Import format always perform a full data extraction (similar to Full Refresh) directly into the TDATASEG_T table.

This field is only available for a data load batch.

8 Select Extract Exchange Rate to extract the exchange rate.

This field is not applicable for a file-based source systems

For more information on extracting exchange rates, see "How Currencies are Processed" on page 116.

For Essbase and Planning applications, in the Export Mode drop-down, select the mode of exporting data.

For Planning applications, in the **Export Mode** drop-down, select the mode of exporting data. Options for Essbase and Planning applications are:

- STORE_DATA
- ADD_DATA
- SUBTRACT_DATA
- OVERRIDE_ALL_DATA

Options for the Financial Management applications:

- Accumulate
- Replace
- Merge

This field is only available for a data load batch.

- 10 Click Save.
- To add a batch job:
- 1 From the **Setup** tab, and then under **Batch**, select **Batch Definition**.
- 2 In Batch Definition and then under Batch Jobs, click Add.
- 3 In Rule Name, specify the rule name associated with the batch job.

You can also select the to navigate to and select the rule name.

- 4 In Job Sequence, specify the order in which to sequence jobs.
- 5 Click Save.

Adding a Batch Group

Batch groups enable you to determine security eligibility when executing batches.

For more information about batch security, see "Setting Security Options" on page 44.

- To add a batch group:
- 1 From the **Setup** tab, and then under **Batch**, select **Batch Definition**.
- 2 Select the Batch Group tab.
- 3 Click Add.
- 4 In Name, specify the batch group name.
- 5 Click Save.
- 6 **Optional**: Associate a batch with a batch group:
 - a. Select the **Batch** tab.
 - b. From the **Batch** summary section, select the batch to associate with a batch group.
 - c. From the **Batch** detail section, select the **Definition** tab.

- d. From the **Batch Group** drop down, select the batch group to associate with the batch.
- e. Click Save.

Executing Batches

Use the Batch Execution feature to show all batches that you have accessed based on the batch group assigned. You can also use the Batch Execution feature to select a batch and execute a rule after parameters passed with the rule have been validated.

Note: The Batch Execution option is only accessible to a user with a Run Integration role.

- To execute a rule:
- 1 From the Workflow tab, and then under Other, select Batch Execution.
- 2 On the Batch Execution summary screen, select a batch name and then, click Execute.
- Optional: You can also schedule a job (see "Scheduling Jobs" on page 258, and check the status of the batch (see "Viewing Process Details" on page 180).
- To check the status of an executed batch:
- 1 From the Workflow tab, and then under Other, select Batch Execution.
- 2 On the Batch Execution summary screen, select a batch name and then, click Check Status.

The Process Details screen is displayed. For more information, see "Viewing Data" on page 174.

Using Open Batches

The open batch functionality is used to read file-based data sources and import them into the appropriate POV based on the file name. This feature enables you to automate the process of loading large number of files. Open batches cannot contain jobs. Additionally, open batches can be scheduled to run periodically.

The high-level process overview of the Open Batches function consists of:

- 1. In Batch Definition, create an openbatch folder in the application inbox\batches subdirectory where the files to be imported are copied.
- 2. Select the File Name Separator character.
 - This character is used to separate the five segments of an open batch file name.
- 3. Select the Auto Create Data Rule option.
- 4. Stage the open batch files by copying files to the inbox\batches\openbatch folder using the name format for batch files.
- 5. In Batch Execution, process the batch.

After a batch is processed, a directory is created, all files within the openbatch directory are moved into the new directory, and the new directory is assigned a unique batch ID

Creating Open Batches

- To create and process open batch:
- 1 From the **Setup** tab, and then under **Batch**, select **Batch Definition**.
- 2 In the Batch Definition summary section, click Add.

Use the blank Name and Target Application fields in the Batch Definition summary grid to enter a batch name or target application on which to search.

- 3 In Batch Definition Detail section, select the Definition tab.
- 4 In Name, specify the name of the batch definition.
- 5 From the **Target Application** drop-down, select the name of the target application.
- 6 In the Type drop-down, select Open Batch.
- 7 In the Execution Mode drop-down, select Serial.

The serial execution mode processes files sequentially, requiring that one file complete its process before the next file starts its process.

- In Open Batch Directory, specify the folder under inbox\batches\openbatch subdirectory where the files to be imported are copied. If this field is blank or null, all files under inbox\batches \openbatch are processed.
- In **File Name Separator**, for an open batch type, select the character to use when separating the five segments of a batch file name.

Options:

- ~
- @
- :
- ;
- 10 Select Auto Create Data Rule to create the data rule automatically for file-based data loads.

When FDMEE assigns the data rule name, it checks whether a data rule with the "Location_Category" name exists. If the name does not exist, FDMEE creates the data rule using the following file name convention:

- Rule Name—Location_Category
- Description—"Auto created data rule"
- Category—Category
- File Name—Null
- Mode—Replace

To use predefined data rules that load data based on specific categories, leave this field blank

- 11 Optional: In the Description field, enter a description of the batch definition.
- 12 Click Save.
- 13 Stage the file-based data source files by copying them to inbox\batches\openbatch using one of the following methods:
 - Predefined Data Load Rule— To use a predefined data rule that loads data based on specific categories, leave the Auto Create Data Rule field blank on the Batch Definition screen and create the data load rule (see "Defining Data Load Rules to Extract Data" on page 143).

Next, create the open batch file name using the following format: FileID_RuleName_Period_LoadMethod. The file id is a free-form field that can be used to control the load order. Batch files load in alphabetic order by file name.

The load method is defined using two-character code identifying the load method where the first code represents the append or replace method for the source load, and second character represents the accumulate or replace method for the target load.

For the source load method, available values are:

- o A—Append
- o R—Replace

For the target load method, available values are:

- A—Accumulate
- o R—Replace
- o M—Merge
- o S—Replace by Security

 $\label{lem:continuous} Examples of an open batch file name are: \verb|a_Texas_Actual04_Jan-2004_RR.txt| and \verb|b_Texas_Actual04_Jan-2004_RR.txt|$

- Auto-Created Data Load Rule—To load data to any location category and have FDMEE create the data load rule automatically, create the open batch file name using the following format: "FileID_Location_Category_Period_LoadMethod". In this case FDMEE looks for the data rule with the name "Location_Category". If it does not exist, FDMEE creates the data rule automatically with the name "Location_Category"
- 14 Optional: Apply any scheduling conditions to the open batch file.

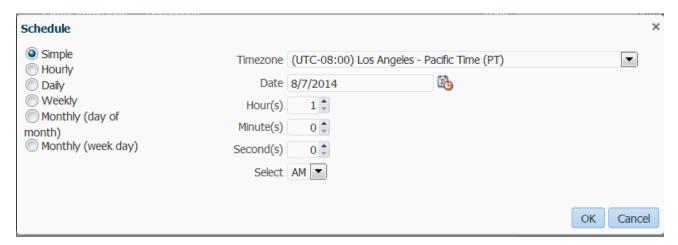
See "Scheduling Jobs" on page 258 for more information.

- 15 From the Workflow tab, and then under Other, select Batch Execution.
- 16 On the Batch Execution summary screen, select the open batch file and then click Execute.

After an open batch is processed, a directory is created, all files within the openbatch directory are moved into the new directory, and the new directory is assigned a unique batch ID.

Scheduling Jobs

The scheduling jobs feature provides a method to orchestrate the execution times of metadata load rules and data load rules.



- To schedule a job:
- 1 From the Batch Execution screen, Metadata screen, or Data Load Rule screen, select the batch name (from the Batch Execution screen) or rule (from the Metadata screen or Data Load Rule/Write Back Rule screens) to schedule and click Schedule.
- 2 In Schedule, select any rule feature specific options.

For example, when you select the **Schedule** option from the **Data Load Rule** screen, specify the Import from Source, Recalculate, Export to Target, and so on options.

For Data Load Rule scheduling specific options, see "Running Data Load Rules" on page 163.

For Write Back Rule scheduling specific options, see "Running Write-Back Rules" on page 226.

- 3 Specify the type of scheduling and select the associated date and time parameters.
- 4 Click OK.

 Table 36
 Schedule Types and Parameters

Schedule Type	Data and Time Parameters
Simple	Submits the job for execution at a specific day and time, but does not repeat:
	Options are:
	Timezone
	• Date
	• Hour(s)
	Minute(s)
	• Second(s)
	Select (AM/PM)
Hourly	Executes at the specified minutes and seconds after the top of the hour every hour until cancelled. Options are:
	Timezone
	Minute(s)
	Second(s)
Daily	Executes at the same time each day. Options are:
	Timezone
	• Hour(s)
	Minute(s)
	Second(s)
	Select (AM/PM)
Weekly	Executes at the specific time for each day selected. Options are:
	Timezone
	Monday-Sunday
	• Hour(s)
	Minute(s)
	Second(s)
	Select (AM/PM)
Monthly (day of month)	Execute at the specific time for the day of the month selected. Also enables you to select the "Last Day of Month" or "D Before End of Month". Options are:
	Timezone
	Monthly Date
	• Hour(s)
	Minute(s)
	• Second(s)
	Select (AM/PM)

Schedule Type	Data and Time Parameters				
Monthly (week day)	You can select the first, second, third, fourth, fifth, last and then the specific day or the week on which to execute the job. Options are:				
	Day of Month				
	• Day				
	Hour(s)				
	Minute(s)				
	Second(s)				
	Select (AM/PM)				
	Note: The Timezone drop down is not available for the Monthly (week day) schedule type.				

Canceling a Scheduled Job

When you cancel a scheduled job from FDMEE, all instances of the scheduled job for the object selected are cancelled. If you want to cancel a specific instance of a schedule, use the ODI Studio or ODI Console.

- To cancel scheduled jobs,
- From the Batch Execution screen, Metadata screen, Data Load Rule screen or Write Back Rule screen, select the batch name (from the Batch Execution screen) or rule (from the Metadata screen or Data Load Rule/Write Back Rule screens).
- 2 Click Cancel Schedule.

Working with Batch Scripts

FDMEE provides a set of Windows batch / UNIX shell scripts that allow users to execute the data load rules from a command line or schedule loads from any scheduler without writing Java code. Batch scripts can be invoked from the command line. In turn, scripts call the data load and metadata load API in the FDMEE application server that execute the rules using the normal process used in data rule and workbench. Batch scripts are located under a<EPM_ORACLE_INSTANCE>/FinancialDataQuality directory (<EPM_ORACLE_INSTANCE> is typically is located at: C:\Oracle\Middleware \user_projects\epmsystem1)

Using a batch script to run data load rules includes:

- Executing the batch script for data load rules. See "Executing the Batch Script for Data Load Rules" on page 261.
- Executing the batch script for metadata rules.

Note: Period names cannot include spaces when used in a batch script.

Using Password Encryption

FDMEE provides a Win/UNIX script to encrypt a password and store in a file. Script encryptpassword.bat/sh is located in EPM ORACLE INSTANCE/FinancialDataQuality directory.

- ➤ To encrypt a password:
- 1 Navigate to the directory that has the batch files.
 - Typically, the batch file directory is EPM ORACLE INSTANCE/FinancialDataQuality directory
- 2 From the EPM ORACLE INSTANCE/ Financial Data Quality directory, at a command prompt, run the script encryptpassword.bat passwordfile.
- 3 When prompted, enter the password and click Enter.

Note the following:

- The password is masked when you type it.
- When running the batch script, you can provide the password by file name as a parameter in the following format: -f:passwordfile. The file name used as a parameter is placed in the location defined in "Encrypted Password Folder" option of System Settings.
- Do not include a disk or directory specification as part of the file name, just enter a file name with or without an extension.
- Replace the [file] with the actual file name, and do not include the brackets.
- The script encrypts the password and writes it to the file provided as parameter.
- For information on running a data load rule batch script with a password from a file, see "Executing the Batch Script for Data Load Rules" on page 261.
- For information on running a metadata load rule batch script with a password from a file, see "Executing the Batch Script for Metadata Rules" on page 263.

Executing the Batch Script for Data Load Rules

- To run the data load rule batch script with a plain text password:
- 1 At a Windows command window or UNIX shell, paste the command: loaddata USER PASSWORD RULE_NAME IMPORT_FROM_SOURCE EXPORT_TO_TARGET EXPORT_MODE IMPORT_MODE LOAD_FX_RATE START_PERIOD_NAME END_PERIOD_NAME SYNC_MODE
- 2 Run the command.
- To run the data load rule batch script with a password from a file:
- 1 At a Windows command window or UNIX shell, paste the command: loaddata USER f:PASSWORD_FILE RULE_NAME IMPORT_FROM_SOURCE EXPORT_TO_TARGET

2 Run the command.

Setting the Parameters for Data Load Rules

The parameters used to execute a batch script for data load rules are:

 Table 37
 Parameters for Data Load Rules

Parameter	Value			
User	Username			
Password	Password or -f:Password file name			
IMPORT_FROM_SOURCE	Y or N			
EXPORT_TO_TARGET	Y or N			
EXPORT_MODE	Essbase and Planning applications export modes are: STORE_DATA ADD_DATA SUBTRACT_DATA OVERRIDE_ALL_DATA Note: The EXPORT_MODE is not supported for custom target applications. The Hyperion Financial Management application export mode is: Accumulate Replace Replace Replace_By_Security			
IMPORT_MODE	 Snapshot Incremental Full Refresh The file-based source system export modes are Append Replace 			
START_PERIOD_NAME	Period Name or POV when the POV specified period value is retrieved from the period profile.			
END_PERIOD_NAME	Period Name or POV when the POV specified period value is retrieved from the period profile.			
SYNC_MODE	SYNC / ASYNC SYNC—Process runs immediately and control returns when process completes. ASYNC—When the ODI job is submitted, control returns. The load process continues to execute in ODI.			

Executing the Batch Script for Metadata Rules

- To run the metadata load rule batch script with a plain text password:
- 1 At a Windows command window or UNIX shell, paste the command: loadmetadata USER PASSWORD LOCATION_NAME SYNC_MODE
- 2 Run the command.
- To run the metadata load rule batch script with a password from a file:
- 1 At a Windows command window or UNIX shell, paste the command: loadmetadata USER f:PASSWORD_FILE LOCATION_NAME SYNC_MODE
- 2 Run the command.

Setting the Parameters for Metadata Rules

The parameters used to execute a batch script for metadata rules are:

Table 38 Parameters for Metadata Rules

Parameter	Value		
User	Username		
Password	Password or –f:Password file name		
Location	Location Name		
SYNC_MODE	SYNC/ASYNC SYNC—Process run immediately and control returns when the process completes. ASYNC—When ODI job is submitted, control returns. The load process continues to execute executed in ODI.		

Executing the Batch Script for HR Data Rules

- To run the HR data rule batch script with a plain text password:
- 1 At a Windows command window or UNIX shell, paste the command: loadhrdata USER PASSWORD LOCATION_NAME SYNC_MODE
- 2 Run the command.
- To run the HR data d rule batch script with a password from a file:
- 1 At a Windows command window or UNIX shell, paste the command: loadhrdata USER f:PASSWORD_FILE LOCATION_NAME SYNC_MODE
- 2 Run the command.

Setting the Parameters for Human Resource Data Rules

The parameters used to execute a batch script for Human Resource data rules are:

Table 39 Parameters for Human Resource Data Rules

Parameter	Value
User	Username
Password	Password or –f:Password file name
Rule Name	HR data rule name
IMPORT_FROM_SOURCE	Y or N
EXPORT_TO_TARGET	Y or N
As of Date	Date used to determine applicable effective date. Date must be in the YYYY-MM-DD format.
Load Type	Specify the load type:
	Data—Loads only the data
	Metadata—Loads only the Public Sector Planning and Budgeting metadata
	Both—Loads the data and metadata.

Executing the Batch Script for Write-Back Rules

- To run the write-back rule batch script with a plain text password:
- 1 At a Windows command window or UNIX shell, paste the command: writebackdata USER PASSWORD LOCATION_NAME SYNC_MODE
- 2 Run the command.
- To run the write-back rule batch script with a password from a file:
- 1 At a Windows command window or UNIX shell, paste the command: writebackdata USER f:PASSWORD_FILE LOCATION_NAME SYNC_MODE
- 2 Run the command.

Setting the Parameters for Write-Back Rules

The parameters used to execute a batch script for write-back rules are:

Table 40 Parameters for Write-Back Rules

Parameter	Value	
User	Username	
Password	Password or -f:Password file name	

Parameter	Value
Rule Name	Write-back data rule name
IMPORT_FROM_SOURCE	Y or N
EXPORT_TO_TARGET	Y or N
SYNC_MODE	SYNC/ASYNC SYNC—Process run immediately and control returns when the process completes. ASYNC—When ODI job is submitted, control returns. The load process continues to execute executed in ODI.

Running a Batch

- To run the batch with a plain text password:
- 1 At a Windows command window or UNIX shell, paste the command: runbatch USER PASSWORD LOCATION_NAME SYNC_MODE
- 2 Run the command.
- To run the batch with a password from a file:
- 1 At a Windows command window or UNIX shell, paste the command: runbatch USER f:PASSWORD_FILE LOCATION_NAME SYNC_MODE
- 2 Run the command.

Setting the Parameters for Running the Batch

The parameters used to run the batch are:

Table 41 Parameters for Running a Batch

Parameter	Value
User	Username
Password	Password or -f:Password file name
Rule Name	Batch Name

9

Creating and Using Scripts

In This Chapter

Overview	267
Key Jython Concepts	267
Using the Script Editor	268
Using Import Scripts	269
Using Mapping Scripts	280
Using Event Scripts	283
Using Custom Scripts	287
Working with Drill Through Scripts	292
Using the JAVA API	292
Using Visual Basic API and Samples.	298

Overview

FDMEE offers a powerful extensibility framework by providing Jython and Visual Basic based scripting. Using the rich set of the API library, users can extend and customize the product to best fit their needs. FDMEE supports four types of scripts:

- Import scripts—Executed as source files are imported. Uses Jython script only.
- Mapping scripts—Used to build mapping logic to derive target members based on multiple source column and IF THEN logic. Uses Jython and SQL script.
- Event script—Executed in response to FDMEE such as before or after an import. Uses Jython and Visual Basic script.
- Custom script—Enables manual execution of scripts. Uses Jython and Visual Basic script.
- Drill script

FDMEE provides a set of Java APIs that enable you to look up frequently used data load properties or to perform a common action. FDMEE also supports Visual Basic based event scripts.

Key Jython Concepts

Before using scripts, you should be familiar with the following important Jython concepts:

Code Indentation

- No Data Type Declaration
- Array Index starts at 0
- Substring is str[Start Position: Start Position + Length].
- ability to access any Java library

For information on Jython concepts: see:

- www.jython.org
- www.python.org

Using the Script Editor

Overview

The Script Editor is used to define Jython scripts that run in response to FDMEE events or file import processes. You can also create custom scripts, and JavaScript supported drill through scripts that allows you to drill through to a graphical user interface instead of an URL. Scripts are saved in the data\scripts directory of the FDMEE application (with a .py extension for Jython scripts or a .vbs extension for Visual Basic scripts). Import scripts are saved to the data\scripts\import subdirectory, event scripts are saved to the data\scripts\custom subdirectory. Scripts can be copied to other FDMEE environments and edited using a text editor.

Launching the Script Editor

- ➤ To access the Script Editor:
- 1 Select the Setup tab.
- 2 Select Scripts, and then select Script Editor.

Script Editor Layout

The left pane of the Script Editor lists the directories that contain the two types of FDMEE scripts: Import and Event. The right pane enables you to write and copy the code for script.

Using Import Scripts

Overview

Note: FDMEE import integration is implemented using the Jython scripting engine. The Visual Basic scripting engine cannot be used with import scripts.

When working with import scripts, note the following considerations:

- Import scripts are executed as source files are imported.
- Import scripts are only valid for file-based data loads.
- Import scripts must be created using Jython functions only. The format is "def <name>(parm1, parm2)" and a return value.
- Import script pass two parameters by FDMEE:
 - o a dimension value defined by the import format
 - o an entire record from the source

For example, when the account column is defined as position 2 to position 4 characters, then this value gets passed as the value for the field, and the whole input line is passed as the second parameter. In the sample script (see "Import Script Sample" on page 279) these parameters are specified by the StrField and StrRec values, but can be any names as long as you remember that the first parameter is the field, and the second one is the entire record.

- The file name and the function name must be the same.
- The return value from the function populates the source column in the TDATASEG table for the specified dimension. You can only return a single value.
- It is recommended that you use a text editor to edit your script, and then copy and paste it into FDMEE.
- Import scripts are not handled in Lifecycle Management.

Creating Import Scripts

- To create import scripts:
- 1 From the **Setup** tab, and then **Scripts**, select **Script Editor**.
- 2 From Script Editor, click New.
- 3 From Script Type, select Import.

Note: The Technology drop down is Jython because import scripts can only be created based on Jython scripting.

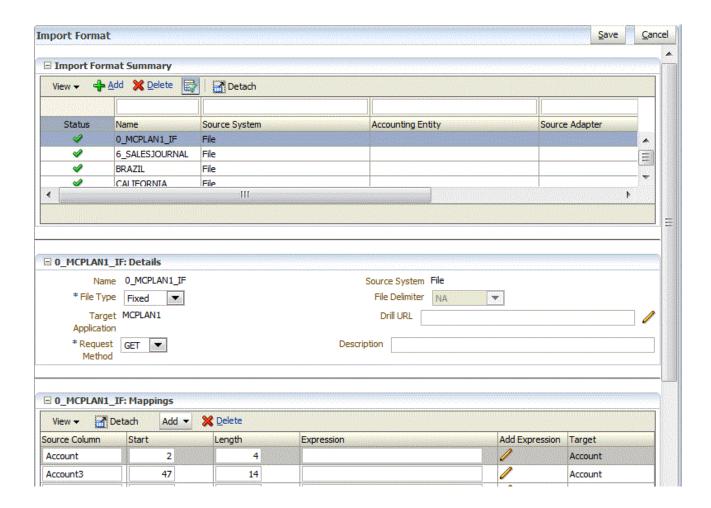
- 4 In File Name, enter a name for the script.
- 5 Click OK.
- 6 From the Target Application drop down, select the target application for this drill through script.
- 7 Write the code for the custom script in the Script Editor.
- 8 Click Save.

Import Script Parameters

The import format script is defined as a Jython function. The Jython function name should be same as the script file name. This function accepts two parameters: the current Field and current row being processed.

- strField—The values of the source-file column that is defined in the import format (For example, for import scripts assigned to the Amount column, strField contains amount values, and, for import scripts assigned to the Account column, strField contains account values.
- **strRecord**—The source-file line (record) that is being scanned.

In following example, the script concatenates the account code with the account description from the source file. For example, the first data line of the file is account code 1110 with a description of "Cash". This script concatenates 1110 with "Cash" and creates a source account in the TDATASEG table named "1110-Cash". (See "TDATASEG Table Reference" on page 387). In the script, you select the field that is defined by the import format, and then the field defined by places 14 through 34 in the current record. (Jython starts numbering columns with 0 rather than 1.)

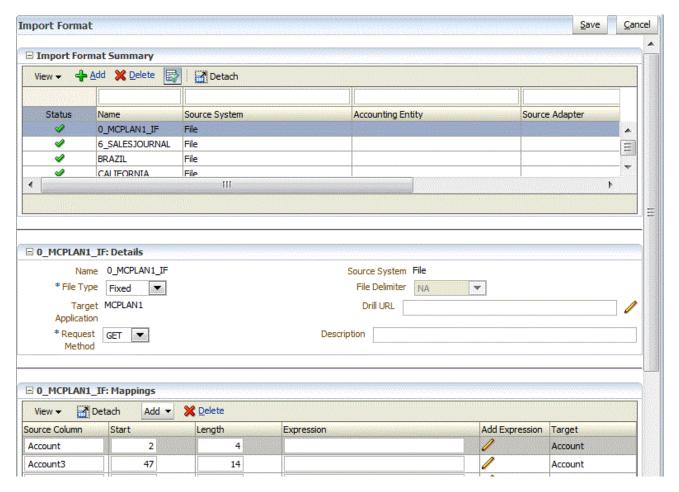


Assigning Import Scripts to Import Formats

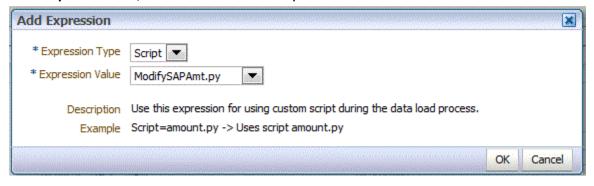
After creating and saving a script in the Script Editor, you assign the script to an import field—any import field within any import format.

- To assign import scripts to import fields:
- 1 From Integration Setup, select Import Format.
- 2 From the Import Format summary grid, select the import format name.
- 3 From the Import Format detail grid, select the Add Expression icon.

You can also type the value directly in the field rather than use the Expression editor.

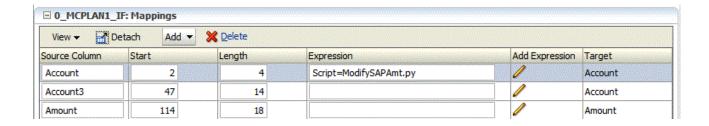


- 4 From Add Expression, and then from Expression Type, select Script.
- 5 In Expression Value, browse to and select the script.



6 Click OK.

The script is assigned to the import field. The name of the import script is displayed in the Expression column.



Using Import Script Functions

Within import scripts, you use functions to manipulate the data that FDMEE imports.

This section uses a problem/solution format to explain how to use the various import script functions.

Extracting Beginning Characters

Problem: The account numbers of the Indiana_GL screen, which are in the first field of each line, include varying numbers of digits. However, the mapping associated with the file uses only four digits.

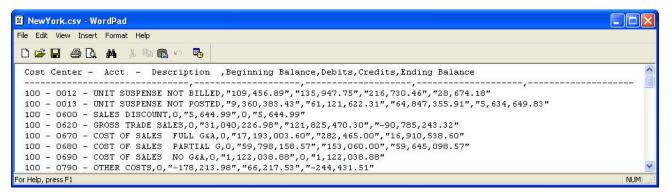
```
📕 Indiana_GL.csv - WordPad
        View Insert
                  Format
           a
                  44
                       お聞聞い
  1100, Cash In Bank, "48, 044.54 "
  1100-101-000-00, Dallas National Bank, "2,000.00 "
  1100-102, Houston Bank One, "6,656.00 "
  1100-103, Midland Bank & Trust, "110,000.00 "
  1100-104, First National Bank, "(10,000.00)"
  1190, Petty Cash, 500.00
  1190-101, Sales, 200.00
  1190-102, Accounting, 500.00
  1210, Trade Receivables, "6,272,205.42 "
  1221, Other Non-Trade Rec, "339,000.00 "
  1221-102, San Antonio, "735, 312.52 "
  1221-104, Other, "117, 169.65 "
  1290-101, North Am. HQ, "45, 759.00 "
  1300-101, Weisbaden, "2, 276, 083.81 "
  1300-102, Berlin, "12, 632, 246.77 "
For Help, press F1
                                                                   NUM
```

Solution: In the Import Scripts screen, assign the following script to the Expression field of the Account row. The script assigns the first four digits of each account number (the first four digits on the left) to Parse_Account:

Result: The import file displays only the first four digits of each account number.

Extracting Middle Characters

Problem: The NewYork file presents cost center, account, and account description as a continuous string, rather than as three strings. You want to separate the account strings from the cost center and account description strings.



Solution: In the Import Formats screen, assign the following script to the Expression field of the Account row. : In the Import Formats screen, assign the following script to the Expression field of the Account row. The script extracts and returns the characters of the account strings (the characters from positions 7 to 10 of the string):

```
def Parse_Account (strfield, strrecord):
    return strField[6:10]
```

Result: Account strings are separated from cost center and account description strings.

Result: In the Account column of the import file, only account numbers are displayed.

Extracting End of String

Problem: The NewJersey screen presents account numbers as the last four characters of account fields. You want to extract only account numbers.

```
NewJersey.csv - WordPad
                                                                                                                _ 🗆 ×
File Edit View Insert Format Help
 Description - Cost Center - Acct ,Beginning Balance,Debits,Credits,Ending Balance
   UNIT SUSPENSE NOT BILLED - 100 - 0012 ,"109,456.89","135,947.75","216,730.46","28,674.18"
   UNIT SUSPENSE NOT POSTED - 100 - 0013 ,"9,360,383.43","61,121,622.31","64,847,355.91","5,634,649.83"
   SALES DISCOUNT - 100 - 0600 ,0,"5,644.99",0,"5,644.99"
   GROSS TRADE SALES - 100 - 0620 ,0,"31,040,226.98","121,825,470.30","-90,785,243.32"
   COST OF SALES FULL G&A - 100 - 0670 ,0,"17,193,003.60","282,465.00","16,910,538.60"

COST OF SALES PARTIAL G - 100 - 0680 ,0,"59,798,158.57","153,060.00","59,645,098.57"
   COST OF SALES NO G&A - 100 - 0690 ,0,"1,122,038.88",0,"1,122,038.88"
   OTHER COSTS - 100 - 0790 ,0,"-178,213.98","66,217.53","-244,431.51"
   BUSINESS UNIT GENERATED - 100 - 0800 ,0,-0.33,0,-0.33
   IR&D OVER/(UNDER) ABSORB - 100 - 0810 ,0,"45,251,768.58","45,251,768.58",0
   BUSINESS UNIT GENERATED - 100 - 0850 ,0,800.92,801,-0.08
   SELLING EXPENSE OVER/(UN - 100 - 0890 ,0,"10,961,886.39","10,961,886.39",0
   BUSINESS UNIT GENERATED - 100 - 0900 ,0,"3,560.99","3,560.99",0
   ENVIRONMENTAL ALLOCATED - 100 - 0905 ,0,520.63,520.63,0
   GENERAL & ADMINISTRATIVE - 100 - 0910 ,0,"59,196.10","59,196.10",0
   GOH HOLDING - 100 - 0920 ,0,"34,247,451.70","34,247,451.70",0
   INCOME ON TAXES AND TAX - 100 - 1020 ,0,0.24,0.35,-0.11
   INTEREST INCOME CUSTOMER - 100 - 1350 ,0,0,224.61,-224.61
  DRAFTS PAYABLE 630142571 - 100 - 1660 ,"-2,532,239.00","11,642,577.00","9,110,338.00",0
TRADE ACCOUNTS RECEIVABL - 100 - 1920 "17 799 164 41" "124 690 256 55" "137 993 009 44" "4 496 411
                                                                                                                 52"
For Help, press F1
                                                                                                                 NUM
```

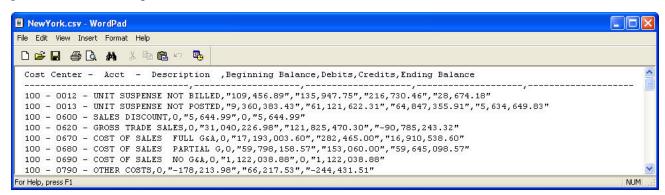
Solution: In the Import Formats screen, assign the following script to the Expression field of the Account row. The script extracts and returns the account numbers (the last four characters of the account fields):

```
def Parse_Account (strfield, strrecord):
return strField[-4:]
```

Result: In the Account column of the import file, only account numbers are displayed.

Using Split Functions

Problem: The NewYork screen presents cost center, account, and account description in one field, rather than in three fields. The three strings are separated by dashes (-). You want to present the three strings as three fields.



Solution: In the Import Formats screen, assign the following scripts, each of which uses the split function, to the Expression fields of the Entity, Account, and Account Description rows (first script to Entity, second script to Account, and third script to Account Description). The first script returns the set of characters before the first hyphen (a cost center value), the second script

returns the set of characters after the first hyphen (an account value), and the third script returns the set of characters after the second hyphen (an account description value).

```
def NY_ParseCenter (strfield, strrecord):
    seglist = strfield.split("-")
    return seglist[0].strip()

def NY_ParseAccount (strfield, strrecord):
    seglist = strfield.split("-")
    return seglist[1].strip()

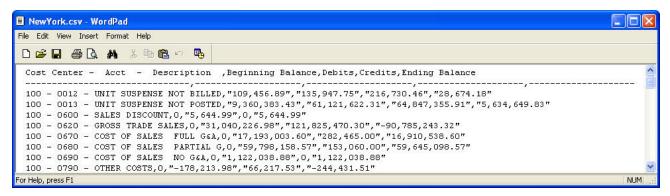
def NY_ParseDesc (strfield, strrecord):
    seglist = strfield.split("-")
    return seglist[2].strip()
```

Result: In the import file, cost center, account and account description strings are presented in three separate fields.

Using the Skip Function (Conditional Skip)

Problem: You want FDMEE to skip all lines of the NewYork screen that contain an entity value that begins with 06.

You must specify fdmSkip as the return value from your function in order to skip a line in the file that is being processed in your import script. You specify fdmSkip as the return argument from an import script by entering return fdmSkip. fdmSkip is only available for import scripts.



Solution: In the Import Scripts screen, you assign the following script to the Expression field of the Entity row. The script parses the entity column and uses a local variable that is discarded after the script executes:

```
def NY_Skip06Center(strField, strrecord):
    if strfield.count("-") > 0:
        seglist = split(strField, "-")
        strEntity = seglist[0]
    if strEntity[:2] == "06":
        return fdmSkip
    else:
        return strEntity
```

Result: No line that contains entity values that begin with 06 is imported.

Storing and Retrieving Temporary Variables

When FDMEE imports a source file, it skips lines that do not contain valid amounts but executes all scripts assigned to the Amount column, regardless of whether amounts are valid. Therefore, you can use scripts that run for lines that FDMEE would otherwise skip to store global variables that can be retrieved by other scripts.

Storing Temporary Variables

Within source files, not all lines contain all fields. For example, in the Georgia screen shown below, entity values, such as 0565 0565 Test 3, are presented in the header of each report section after the Bus Area / Dept label, but are not presented in every line. Therefore, entity values must be stored in global variables and assigned to the lines of the source file.

For the Georgia file, to store entity values in global variables, in the Import Formats screen, you assign the following script to the Expression field of the Amount row. The script uses an if.. statement and the string function to determine whether lines contain the Bus Area / Dept: label. If a line contains the label, the script stores the entity value (located in position 33 and including 4 characters) in a global variable. If a line does not include the label, strfield returned.

To use a global variable, define a string variable outside the import function. When assigning a value to the variable inside the import function, designate it as global. By defining a variable outside the function, it is available to any function used in that session.

The global variables remain available throughout the current FDMEE data load process (unlike local variables, which lose their values when the current script is closed). Therefore, values stored from within one script can be retrieved by other scripts.

```
GeorgiaCenter = ""
def GeorgiaGetCenter (strfield, strrecord):
   if strrecord[15:31] == "Bus Area / Dept:":
        global GeorgiaCenter
        GeorgiaCenter = strrecord[32:36]
   return strfield
```

	gia.glo - Notepad					
File Edit	Format View Help					
Upstream Software			Summary1 Trial Balance Period: NOVO3-04		Report Date: 16-DEC-2003 13:08 Page: 44 of 63	
	Currency: Balance Type: Bus Area / Dept Range:	Year to Date				
	Bus Area / Dept:	0563 0563 - Test1				
Acct	Description		Beginning Balance	Debits	Credits	Ending Balance
0010 0012	0010 – Cash 0012 – AP		(971,295.74) 0.00	951.00 2,002.00		(970,344.74) 0.00
		-	(971,295.74)	2,953.00	2,002.00	(970,344.74)
Upstre	am Software		Summary1 Trial Bal Period: NOVO3	lance 3-04	Report Date: Page:	16-DEC-2003 13:08 45 of 63
	Currency: Balance Type: Bus Area / Dept Range:	Year to Date			Sole Poste	
	Bus Area / Dept:	0564 0564 - Test2				_
Acct	Description		Beginning Balance	Debits	Credits	Ending Balance
0010 0012	0010 - Cash 0012 - AP		5,666,682.18 0.00	1,282,699.97 403.00	6,949,282.15 403.00	100.00
			5,666,682.18	1,283,102.97	6,949,785.15	0.00
Upstream Software		Summary1 Trial Bal Period: NOVO3	lance 3-04	Report Date: Page:	16-DEC-2003 13:08 46 of 63	
	Currency: Balance Type: Bus Area / Dept Range:	Year to Date				
	Bus Area / Dept:	0565 0565 - Test3				
Acct	Description		Beginning Balance	Debits	Credits	Ending Balance
0010	0010 – Cash				269,201,268.90	

Retrieving Temporary Variables

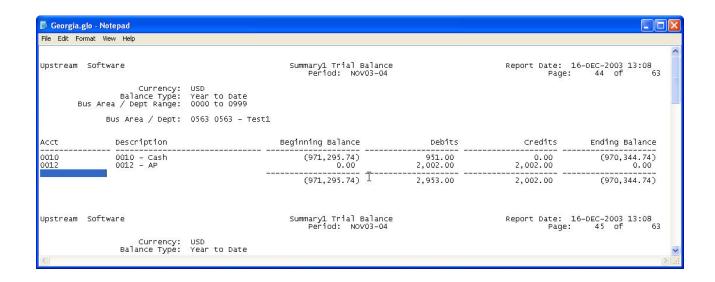
You use scripts to assign global, stored variables to the fields of a column. For example, when you are working with the Georgia screen, you begin by using the following script to assign the global variable GeorgiaCenter to the **GeorgiaPutCenter** function.

```
def GeorgiaPutCenter (strfield, strrecord)
return GeorgiaCenter
```

Then, in the Import Formats screen, you assign the script to the Expression field of the Entity row and, thus, assign the values of the global variables to the Entity fields. In this case, entity values are not read from the source file.

Because the Georgia file includes subtotal lines that must not be imported, the Skip function is required.

To direct FDMEE to skip lines without account numbers, you configure a Skip Row in the Import Format to recognize blank Account fields (15 blank spaces) by defining appropriate start and length fields for the expression.



Import Script Sample

This sample import script returns the location as a value for the column.

```
# Sample shows how to use the value from the fdmContext map, In
# this sample return the Location as value for the column
def getOrgfromLoc(strfield, strrec):
   org = fdmContext['LOCNAME']
   return org
# Sample to show the Jython string function. The script below is
# used to parse a account column 01-205-4110-0000-000 and return the
# third string
#-------
def getSegfromAcct(strfield, strrec):
 if strfield.count("-") > 0:
   seglist = strfield.split('-')
   result = seglist[2]
   return result
# Sample to process header record in Trial Balance report and
# store value in global variable to be used during detail row
# This should be called from the Amount column in import format
globalorg = ""
globalcur = ""
def copyglobal(strfield, strrec):
 if strrec[18:27] == "Currency:" :
   global globalcur
   globalcur = strrec[29:32]
```

Using Mapping Scripts

Overview

Mapping Scripts are used to build mapping logic to derive target members based on multiple source column and IF THEN logic. Mapping scripts are added in the Data Load Mapping screen, and are available for: Between, IN, Like types. They are stored in the TDATAMAP table.

When you use Lifecycle Management to export mapping rules, any related mapping scripts are included.

The import of mapping rules using an Excel template does provides a place to specify a mapping script.

Creating Mapping Scripts

- To create a mapping script:
- 1 From the Workflow tab, select Data Load Mapping.
- 2 Optional: Select the desired location.
- 3 Select either the Like, Between, or In tab.

Mapping scripts are not available for "Explicit" and "Multi-dimension" mapping types.

- 4 Select the **Source Value**.
- 5 In Target Value, select one of the following:
 - For a Jython based mapping script, enter **#SCRIPT**.
 - For a SQL based mapping script, enter **#SQL**.

Note: For SQL based mappings, FDMEE specifies the special value #SQL to the "Target Value." For this reason, this field cannot be used to specify a value of "IGNORE", which is otherwise used to set the VALID_FLAG to Yes and No. For SQL based mappings, you must include the VALID_FLAG in the SQL script and specify the value of "I" based on the user selected criteria.

- 6 In Rule Name, enter the data rule name for the mapping.
- 7 Click Save.

The Script Editor icon () is enabled.

- 8 Click the Script Editor icon.
- 9 From Edit Script, enter the mapping script, and click OK.

For example the following Jython based mapping script checks when the first two characters of the account equals 11 and the entity equals 01. When a match occurs, then the target is set to Cash2 and the target amount is set to 0. In addition, the script also updates the attribute1 column in the TDATASEG table. (See "TDATASEG Table Reference" on page 387). This script uses the fdmResultMap object (see "Using Jython Objects with Mapping Scripts" on page 282).

In the next example, a SQL script can be used where logic is implemented in a SQL CASE statement. The CASE statement is used in the SET command of the internal UPDATE statement. he mapping statement below is converted to the UPDATE statement listed

```
Script

CASE
WHEN ACCOUNT LIKE 'L%' AND ICP = '000' THEN 'A4140'
WHEN ACCOUNT IN ('110','120','130') THEN 'A100''
LSE 'A' || ACCOUNT

ELSE 'A' || ACCOUNT

DESC1 = COALESCE (DESC1, ACCOUNT || '.' || ICP)
```

UPDATE TDATASEG

```
CASE

WHEN ACCOUNT LIKE 'L%' AND ICP = '000' THEN 'A4140'
WHEN ACCOUNT IN ('110','120','130') THEN 'A100''
ELSE 'A' || ACCOUNT

END
, DESC1 = COALESCE(DESC1, ACCOUNT || '.' || ICP)

WHERE ACCOUNT ....
```

Using Jython Objects with Mapping Scripts

The following predefined FDMEE Jython objects can be used within mapping scripts.

Table 42 Using Jython objects with a Mapping Script

Jython Object	Description
fdmRow	fdmRow id used to access any column in the current row being processed by the mapping process. You can access any column in the TDATASEG table. To access a value in a column you specify the following: fdmRow.getString[" <column name="">"]. For example, to get the value of the ENTITY column you specify fdmRow.getString["ENTITY"].</column>
fdmResult	fdmResult is used to return the value from the mapping function. You can set the value as follows fdmResult = "Cash". The fdmResult is used to update the value for the target dimension, which is also referred to as the "X" column.
	For every dimension in the TDATASEG table there are two columns, one for source and another for target. For example, ENTITY provides the source value, and ENTITYX provides target value that is the result of the mapping process.
fdmResultMap	fdmResultMap is used to update any column in the TDATASEG table. The column name in the TDATASEG table is used as the key in the map. For example, to update the ATTR1 column use fdmResultMap["ATTR1"] = "Asset Acct". To set the target amount to 0 use fdmResultMap["AMOUNTX"] = 0.

Mapping Script Sample

This sample mapping script evaluates the account and entity columns and assigns a value for the target column. It also shows how to update the other columns of the current row using the fdmResult map:

Using Event Scripts

Overview

Event scripts are executed in response to FDMEE events. You can invoke any custom logic. For example, custom logic could be invoked to send an email notification after a successful load, or send a email when a validation error occurs. Or an email could be sent when you download data from Financial Management and load data to Essbase for reporting. Event scripts are based either on Jython or Visual Basic. Jython scripts have a .py extension, and Visual Basic scripts have a .vbs extension. Events associated with a script are highlighted in the list of events. Any event that includes a script is executed for that event for the selected location.

Note: Event scripts are not handled in Lifecycle Management.

FDMEE Supported Event Scripts

FDMEE supports the following events for execution, during the data load process:

Table 43 FDMEE Events

Event	Description		
Befimport	This event is the initial state of the system before any processing for the selected location has begun. If the user defines import scripts, they are run between the BefImport and AftImport events. This step in the processing loads data into the TDATASEG_T table.		
AftImport	Data is present in the TDATASEG_T table after this event is processed.		
BefCalculate	Called for a validation run only, and called before the validation process.		
AftCalculate	Called for a validation run only, and called after the validation process.		

Event	Description
BefProcLogicGrp	Called before Logic Account is processed.
AftProcLogicGrp	Called after the Logic Account is processed.
BefProcMap	Called before the mapping process is started in the TDATASEG_T table. Mapping scripts are executed between the BefProcMap and AftProcMap events. Data is moved from the TDATASEG_T table to the TDATASEG table between these events after all data has been mapped. Updates to audit tables are also included as part of this process.
AftProcMap	Called after all the mapping has been completed. When this event runs, the data has already been moved from the TDATASEG_T table to the TDATASEG table.
BefValidate	Checks when data mapped in the TDATASEG table.
AftValidate	Called after the BefValidate event.
BefExportToDat	Called before you write to the file for export. It is also executed for Essbase in the export to file mode feature. Note: This event is not available for the Accounts Reconciliation Manager.
AftExportToDat	Called after the file is created. Note: This event is not available for the Accounts Reconciliation Manager.
BefLoad	Called before you load to a target application.
	Note: This event is not available for the Accounts Reconciliation Manager.
AftLoad	Called after data is sent to the target application, but does not wait for the target application to complete the load process.
	Note: This event is not available for the Accounts Reconciliation Manager.
BefConsolidate	Oracle Hyperion Financial Management and Oracle Essbase only: This event executed when a check rule is included in the location that is being processed.
	Note: This event is not available for the Accounts Reconciliation Manager.
AftConsolidate	Called after the BefConsolidate event.
	Note: This event is not available for the Accounts Reconciliation Manager.
BefCheck	Called before the Check Rule.
	Note: This event is not available for the Accounts Reconciliation Manager.
AftCheck	Called after the Check Rule.
	Note: This event is not available for the Accounts Reconciliation Manager.

Creating Event Scripts

- ➤ To create event scripts:
- 1 From the **Setup** tab, and then under **Scripts**, select **Script Editor**.
- 2 Single or double click an event.
- 3 Write the script in the Script Editor.

Dynamically Changing Import Formats

You can use the BefFileImport event to modify the import format for a location dynamically. The following script changes the import group; the change is based on the file name.

To implement the script, you create two import formats and assign one of them to the FDMEE location. The script evaluates the file name and, when necessary, changes the import format.

```
if fdmContext["LOCNAME"] == "ITALY":
    filename = fdmContext["FILENAME"]
    if filename[:12] == "ProductSales":
        fdmAPI.updateImportFormat("SALESJOURNAL", fdmContext["LOADID"])
```

Using the File System Object in Event Scripts

You can use the Jython file system object to process files and folders. The following example uses the file system object to create a file and to copy the contents of an existing file to the new file.

```
Read the following Input File
Entity, Currency, ICP, Product, Store, Channel, Custom4, Custom5, Custom6, Custom7, UnitsSold, Sale
EastSales, USD, [ICP None], H740, Comma_Phone_Stores, Retail_Direct, [None], [None],
[None], [None], 127, 9954.103768
EastSales, USD, [ICP None], H740, Freds, National_Accts, [None], [None], [None], 112,
6610.371552
EastSales, USD, [ICP None], H740, Good_Buy, National_Accts, [None], [None], [None], [None],
112,6610.371552
Write the following Output File
EastSales, USD, [ICP None], H740, Comma_Phone_Stores, Retail_Direct, [None], [None],
[None], [None], UnitsSold, 127
EastSales, USD, [ICP None], H740, Comma_Phone_Stores, Retail_Direct, [None], [None],
[None], [None], Sales, 9954.103768
EastSales, USD, [ICP None], H740, Freds, National_Accts, [None], [None], [None],
[None], UnitsSold112
EastSales, USD, [ICP None], H740, Freds, National_Accts, [None], [None], [None],
[None], Sales6610.371552
EastSales, USD, [ICP None], H740, Good_Buy, National_Accts, [None], [None], [None],
[None], UnitsSold, 112
EastSales, USD, [ICP None], H740, Good_Buy, National_Accts, [None], [None], [None],
[None], Sales, 6610.371552
infilename = fdmContext["INBOXDIR"]+"/InputFile.txt"
outfilename = fdmContext["INBOXDIR"]+"/DataFile.txt"
infile = open(infilename, "r")
outfile = open(outfilename, "w")
for line in infile:
 column = line.rsplit(',',2)
 if column[2].strip() != "Sales" :
    outfile.write(column[0] + ",UnitsSold," + column[1] + "\n")
    outfile.write(column[0] + ",Sales," + column[2])
outfile.close()
```

Events Script Sample

This sample Event script updates the table_xyz table during the data load execution:

```
#-----
# Sample to update table_xyz table during data load rule execution
#-----
query = "UPDATE table_xyz SET accountx = 'SCRIPT_' | | account WHERE loadid = ? and
accountx is NULL"
params = [ fdmContext["LOADID"] ]
print fdmAPI.executeDML(query, params, False)
fdmAPI.commitTransaction()
#-----
# Sample to import data from a custom SQL source and upload into FDMEE
# open interface table. This script should be called in BefImport Event.
# This is alternate to the FDMEE integration import script.
#______
import java.sql as sql
batchName = "Batch_" + str(fdmContext["LOCATION"])
insertStmt = """
INSERT INTO AIF_OPEN_INTERFACE (
  BATCH_NAME
 ,COL01
 ,COL02
 , AMOUNT
 ,DESC1
) VALUES (
,?
,?
,?
,?
)
....
sourceConn = sql.DriverManager.getConnection("jdbcUrl", "user", "password");
# Limiting number of rows to 5 during the test runs.
selectStmt = "SELECT * FROM orders WHERE rownum < 6"</pre>
stmt = sourceConn.prepareStatement(selectStmt)
stmtRS = stmt.executeQuery()
while(stmtRS.next()):
 params = [ batchName, stmtRS.getBigDecimal("Customer_Id"),
           stmtRS.getString("Ship_Country"),
           stmtRS.getBigDecimal("Freight"), stmtRS.getString("Ship_Name") ]
 fdmAPI.executeDML(insertStmt, params, False)
fdmAPI.commitTransaction()
stmtRS.close()
stmt.close()
sourceConn.close()
# Sample to send email messages using Jython SMTP library
```

Using Custom Scripts

Overview

FDMEE enables you to perform FDMEE tasks on a "as needed" basis such as executing data load rules using custom scripts.

FDMEE supports custom scripts in Jython and Visual Basic. To execute a custom ODI process, then use a Jython script. FDMEE stores custom scripts in the <app display="font-size: custom folder">APP DIR>/data/scripts/custom folder.

Note: Custom scripts are not handled in Oracle Hyperion Enterprise Performance Management System Lifecycle Management.

Creating a Custom Script

- To create a custom script:
- 1 From the **Setup** tab, and then under **Scripts**, select **Script Editor**.
- 2 From Script Editor, click New.
- 3 From Script Type, select Custom.
- 4 From Technology, select Jython or Visual Basic.

Scripts created in Jython are saved with a .py extension. Scripts created in Visual Basic are saved with a .vbs extension.

Custom scripts cannot use JavaScript technology.

- 5 In **File Name**, enter a name for the script.
- 6 Click OK.
- 7 From the Target Application drop down, select the target application for this drill through script.

- 8 Write the code for the custom script in the Script Editor.
- 9 Click Save.

Working with Custom Scripts

You must create custom script groups before registering custom scripts.

You register custom scripts to select the parameters to pass when the script is executed.

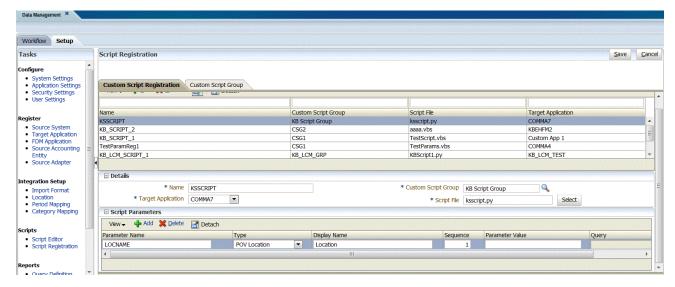
Adding a Custom Script Group

Custom scripts groups enable you to assign similar types of custom scripts under a group for ease of use. They are also used to determine security eligibility when executing custom scripts.

- ➤ To add a custom group:
- 1 From the Workflow tab, and then under Scripts, select Script Registration.
- 2 Select the Custom Script Group tab.
- 3 Click Add.
- 4 In Name, specify the custom script group name.
- In Sequence, enter the display order used when custom script groups are shown during registration and execution.
- 6 Click Save.

Registering Scripts

Registered scripts are associated with a script file (which consists of either Jython or Visual Basic script) and a target application. When the custom script is executed, the user is prompted with a list of values from which to complete the parameters of the script.



- To register a script:
- 1 Select Setup, and then under Scripts, select Script Registration.

The Custom Script Registration screen consists of three regions:

- Summary—Lists all custom scripts.
- Details— Shows the script name, associated target application, and script file.
- Parameters—Shows the parameter name and type, display name, sequence, parameter value, and any query definition used to supply a list of values for a given custom script.
- 2 Above the Summary grid, click Add.
- 3 In the Details grid, in Name, enter the name of the custom script.
- 4 In Target Application drop down, select the target application associated with the custom script.
- To associate the custom script with a custom script group, from the Custom Script Group drop down, select the group.
- 6 In Script File, select the script file to associate with the custom script.

To search on a script file, click **Select** and choose a script file from the **Select Script File** screen.

To upload a script file, click **Select**. On the **Select Script File**. click **Upload**. Then click **Browse** to navigate to the script file to upload and click **OK**.

- 7 Click Save.
- To define the parameters of the custom script:
- 1 From the Summary grid, select the name of the custom script to which to add parameters.
- 2 In the Script Parameters grid, click Add.

A blank line is displayed to which to add parameters.

3 In **Parameter Name**, enter the name of the parameter that you will reference in the script.

For example, enter POVLOC, for the POV location, POVPeriod for the POV period, POVCAT for POV category, or POVTARGET for the target category.

The parameter name is user-defined. You can select a prefix or suffix to distinguish them in the custom script.

4 In **Type**, select the type of parameter.

The types are the parameters that you will reference in your custom script Available types:

- POV—Prompts for the POV location, period, category or rule. The POV values default from the users' POV when the custom script is executed.
- Query—The Query type allows you to create a query that populates a list of values that a user can select from when the custom script is executed.

• Static—A Static type indicates that the parameter is a constant or a predefined value, and the parameter value is supplied by the user. When the custom script is executed, you can supply a different value to the parameter.

You can use any name for the parameter and also use a prefix or suffix with the name to distinguish them in the custom script.

The parameter name must exactly match what is provided in the script registration

- 5 In **Display Name**, enter the name to display for the parameter for the Execute Script screen.
- 6 In Parameter Value, enter the value for the "Static" parameter type.

The parameter value must be retrieved in the custom script using the following API:

```
fdmAPI.getCustomScriptParameterValue("PARAMETER NAME"))
```

- 7 In **Sequence**, enter the display order of the parameter on the Execute Script screen.
- 8 In **Query**, specify the query definition that provides a list of values for the parameter.

For example, to show a list of data rules for the user to select on the Generate Report screen, specify the query definition to define these queries.

9 Click Save.

Executing a Custom Script

- To execute a custom script:
- 1 From the Workflow tab, and then under Other, select Script Execution.
- 2 In Script Execution, from the Custom Script Group drop down, select a custom script group.
- 3 From the **Scripts** grid, select the custom script.
- 4 Click Execute.
- 5 When prompted, enter parameter values on the Execute Script screen.
 - a. If applicable, modify the Period, Category, and Location values.
 - b. From Execution Mode, select the online method of running the report.
 - The online method processes the report immediately.
 - c. Click OK.

Custom Script Sample using Jython

This sample custom script provides information about the contents of the custom script:

```
#
This sample Jython script provides code related to custom scripts. All the messages being logged (meaning printed) are visible in the process log file as per Log Level profile. #
```

```
# Log the beginning of the script, at the INFO level
fdmAPI.logInfo("==============")
fdmAPI.logInfo("Custom Script: Begin")
fdmAPI.logInfo("=========================")
# Log the target application name from the context, at the DEBUG level
fdmAPI.logDebug("Target App Name = " + fdmContext["TARGETAPPNAME"])
# Log the script name from the context at the DEBUG level
fdmAPI.logDebug("Custom Script name = " + fdmContext["SCRIPTFILE"])
# Get all script parameters and log their names and values at the DEBUG level
fdmAPI.logDebug("Custom script parameter values by name: begin")
fdmAPI.logDebug("The value of parameter CUSTOM_LOCATION is = " +
fdmAPI.getCustomScriptParameterValue("CUSTOM_LOCATION"))
fdmAPI.logDebug("The value of parameter CUSTOM_YEAR is = " +
fdmAPI.getCustomScriptParameterValue("CUSTOM_YEAR"))
fdmAPI.logDebug("Custom script parameter values by name: end")
# Example submitting a data rule
import os
import subprocess
os.chdir("C:/Oracle/Middleware/user_projects/epmsystem1/FinancialDataQuality")
myScriptName = "C:/Oracle/Middleware/user_projects/epmsystem1/FinancialDataQuality/
"admin", "password", "SRESBA1_DR1", "Y", "N", "STORE_DATA", "SNAPSHOT", "N", "Jan-2003",
"Jan-2003", "ASYNC")
fdmAPI.logDebug("Submitting a data rule via a script using the following command: " +
retcode = subprocess.Popen(command)
fdmAPI.logDebug("Data rule submitted fine.")
# Close the connection and log the end of the script, at INFO level
fdmAPI.closeConnection()
fdmAPI.logInfo("===================")
fdmAPI.logInfo("Custom Script: end")
fdmAPI.logInfo("==========")
The output for the executed custom script is:
2013-09-25 08:12:26,080 INFO [AIF]:
______
2013-09-25 08:12:26,081 INFO [AIF]: Custom Script: Begin
2013-09-25 08:12:26,082 INFO [AIF]:
______
2013-09-25 08:12:26,083 DEBUG [AIF]: Target App Name = SRESBA1
2013-09-25 08:12:26,084 DEBUG [AIF]: Custom Script name = SRCustomScript1.py
2013-09-25 08:12:26,087 DEBUG [AIF]: CUSTOM_LOCATION = 1
2013-09-25 08:12:26,087 DEBUG [AIF]: CUSTOM_YEAR = 2013
2013-09-25 08:12:26,088 DEBUG [AIF]: Custom script parameter values by name: begin
2013-09-25 08:12:26,091 DEBUG [AIF]: The value of parameter CUSTOM_LOCATION is = 1
2013-09-25 08:12:26,093 DEBUG [AIF]: The value of parameter CUSTOM_YEAR is = 2013
2013-09-25 08:12:26,094 DEBUG [AIF]: Custom script parameter values by name: end
2013-09-25 08:12:26,535 DEBUG [AIF]: Submitting a data rule via a script using the
following command: C:/Oracle/Middleware/user_projects/epmsystem1/FinancialDataQuality/
```

Working with Drill Through Scripts

See "Drill Through using JavaScript" on page 60.

Using the JAVA API

Overview

FDMEE automatically initializes the data load workflow context information prior to invoking the Import, Mapping and Event scripts. The fdmContext object is initialized with the list of properties listed below. The properties can be accessed by referencing as fdmContext["<PROPERTY NAME>"]. For example to access Location Name, use fdmContext["LOCNAME"]. To execute a script for a specific data load, write something like the following:

```
if fdmContext["LOCNAME"] == "ITALY":
     Write script logic here
```

FDM object properties include:

- APPID
- BATCHSCRIPTDIR
- CATKEY
- CATNAME
- CHKSTATUS
- EPMORACLEHOME
- EPMORACLEINSTANCEHOME
- EXPORTFLAG
- EXPORTMODE
- EXPSTATUS
- FILEDIR
- FILENAME
- IMPORTFLAG

- IMPORTFORMAT
- IMPORTMODE
- IMPST
- IMPSTATUS
- INBOXDIR
- LOADID
- LOCKEY
- LOCNAME
- MULTIPERIODLOAD
- OUTBOXDIR
- PERIODNAME
- PERIODKEY
- PROCESSSTATUS
- RULEID
- RULENAME
- SCRIPTSDIR
- SOURCENAME
- SOURCETYPE
- TARGETAPPDB
- TARGETAPPNAME
- VALSTATUS

JAVA API List

Table 44 JAVA API List

API	Description
BigDecimal getPOVLocation(BigDecimal pLoadId)	Returns the Partition Key based on the LOADID.
BigDecimal getPOVCategory(BigDecimal pLoadId)	Returns the Category Key based on the LOADID.
Date getPOVStartPeriod(BigDecimal pLoadId)	Returns the Period Key of the start period based on the LOADID.
Date getPOVEndPeriod(BigDecimal pLoadId)	Returns the Period Key of the end period based on the LOADID. when loading single period the start period and end period are the same.
int executeDML(String query,Object[] parameters)	Execute any DML Statement. Provide the query and parameter. The parameter is provided as a list.

API	Description
logDB(String pEntityType, String pEntityName, int pLogSequence, String pLogMessage)	Log messages to a database table AIF_PROCESS_LOGS.
logFatal(String pLogMessage)	Log a message when a fatal error occurs. This message is displayed at all log levels.
logError(String pLogMessage)	Log a message when a error occurs. This message is displayed at log level 2 or higher.
logWarn(String pLogMessage)	Log a message when a warning condition error occurs. This message is displayed at log level 3 or higher.
logInfo(String pLogMessage)	Log a informational message. This message is displayed at log level 4 or higher.
logDebug(String pLogMessage)	Log a debug message. This message is displayed at log level 5.
Map getProcessStates(BigDecimal pLoadId)	Returns status of Workflow process.
	Available statuses:
	IMPSTATUS—Returns the status of the import process.
	A <i>O</i> status indicates that the process has not yet started, or the process failed. A <i>1</i> status indicates the process is successful.
	VALSTATUS—Returns the status of validation process.
	A <i>O</i> status indicates that the process has not yet started, or the process failed. A <i>1</i> status indicates the process is successful.
	EXPSTATUS—Returns the status of export process.
	A 0 status indicates that the process has not yet started, or the process failed. A 1 status indicates the process is successful.
	CHKSTATUS—Returns the status of check process.
	A <i>O</i> status indicates that the process has not yet started, or the process failed. A <i>1</i> status indicates the process is successful.
	PROCESSSTATUS—Returns the exact error code. The detail for the error code can be found in the tlogprocessstates table.
Map getPeriodDetail(Date pPeriodKey,String pApplicationName) // returns PERIODTARGET and YEARTARGET	Returns the Target period mapping for a given Period key.
Object getCustomScriptParameterValue(BigDecimal pLoadId,String pParameterName)	Returns the value for given custom script parameter name and loadID.
Object getCustomScriptParameterValue(String pParameterName)	Returns the value for given custom script parameter name and context initialized loaded,
ResultSet getBatchDetails()	Returns batch definition information from AIF_BATCHES table.
ResultSet getBatchJobDetails(BigDecimal pLoadId)	Retrieve error messages logged to the database table AIF_ PROCESS_LOGS for the given loadid.

API	Description
ResultSet getCategoryList()	Returns list of Categories in a result set.
ResultSet getCheckEntityGroupList(BigDecimal pApplicationId)	Returns list of Check Group in a result set.
ResultSet getCheckEntityForGroup(String pValGroupKey)	Returns list of Entity in a Check Group in a result set.
ResultSe tgetCheckEntityGroupList(BigDecimal pApplicationId)	Returns list of Check Rule Group in a result set.
ResultSet getCheckEntityForGroup	Returns list of Check Rule Group rules in a result set.
ResultSet getCustomDBLog()	Retrieve error messages logged to the database table AIF_ PROCESS_LOGS for the current process.
ResultSet getCustomDBLog(BigDecimal pLoadId)	Returns the log statements from DB for a given loadID
ResultSet getCustomScriptParameters()	Returns the list of custom script parameters in a resultset for the context initialized loadID.
ResultSet getCustomScriptParameters(BigDecimal pLoadId)	Returns the list of custom script parameters in a resultset for the given loadID
ResultSet getPeriodList()	Returns list of Periods in a result set.
ResultSet executeQuery(String query, Object[] parameters)	Execute any SQL Query Statement. The query results are returned in a result set. Provide the query and parameter. The parameter is provided as a list.
ResultSet getImportFormatDetails(String pImpGroupKey)	Returns the Import Format details in a result set based on the Import Format key.
ResultSet getImportFormatMapDetails(String pImpGroupKey)	Returns the Import Format Mapping details in a result set for a given Import Format key. This currently supports only file based import formats.
ResultSet getLocationDetails(BigDecimal pPartitionKey)	Returns the Location details in a record set for a given Location key.
ResultSet getRuleDetails(BigDecimal pRuleId)	Returns the Data Rule details in a record set for a given Data Rule ID.
showCustomDBLog()	Show a list of custom messages in the user interface after completion of a process. Message can be displayed at the end of a data load workflow step like import, validate, export, check or at the end of a custom script execution. Note messages are displayed only when the process are run in online mode.
showCustomFile(String filePath)	Show a custom file (log file, report file) in the user interface after completion of a process. Message can be displayed at the end of a data load workflow step like import, validate, export, check or at the end of a custom script execution. Note messages are displayed only when the process are run in online mode.
showCustomMessage(String message)	Show a custom message in the user interface after completion of a process. Message can be displayed at the end of a data load workflow step like import, validate, export, check or at the end of a custom script execution. Note messages are displayed only when the process are run in online mode.

API	Description
String getCategoryMap(BigDecimal pCatKey,String pApplicationName)	Returns the Scenario for a given Category and Application Name.
String getCustomMessage()	Retrieve the last custom message raised for the current process.
String getCustomMessage(BigDecimal pLoadId)	Retrieve the last custom message raised for the given loadid.
String getCustomFile()	Retrieve the custom file created for the current process.
String getCustomFile(BigDecimal pLoadId)	Retrieve the custom file created for the given loadid.
String getPOVDataValue(BigDecimal pPartitionKey)	Returns the data value of the Location.
String getDirTopLevel(BigDecimal pApplicationId)	Returns the Top Level directory based on the Application.
String getDirInbox(BigDecimal pApplicationId)	Returns the Inbox directory based on the Application.
String getDirOutbox(BigDecimal pApplicationId)	Returns the Outbox directory based on the Application.
String getDirScripts(BigDecimal pApplicationId)	Returns the Scripts directory based on the Application.
String getProcessLogFile(BigDecimal pLoadId)	Returns the log file name for the given loadID.
String getProfileOptionValue(String pProfileOptionName, BigDecimal pApplicationId, String pUserName)	Returns the value set for an option. Options can be set at the System Setting, Application Setting, and User Setting. The order of precedence is: User, Application, and System. The API determines the appropriate applicable value and returns the value.
void writeToProcessLogsDB(BigDecimal pLoadId, String pEntityType, String pEntityName, int pLogSequence,String pLogMessage)	Writes the log information to the AIF_PROCESS_LOGS table. Use Entity Type and Entity Name to group the logged messages. Logs can be created only from a data load workflow process.
void writeToProcessLogsFile(BigDecimal pLoadId, String pLogMessage)	Writes the log information to the Data Load Process log file. The logs is written based on the process log level. Logs can be created only from a data load workflow process.
	Note: It is recommended that you use the logging API, for example logDebug or, logInfo API, instead of using the writeToProcessLogsFile API.
void closeConnection()	Use to close the database connection.
void closeResultSet(ResultSet resultSet)	Use to close result object.
void updateImportFormat(String pImpgroupKey,BigDecimal pLoadId)	Update the import format for the current run. This is applicable only for File-based import formats.

Working with UNICODE Characters in Jython Scripts

When writing script code in Jython technology, specify any non-English strings in UNICODE by prefixing the letter "u" before the string in double quotes. This means instead of defining a string as "MyValue" define it as u"MyValue". See the following example used in data load mapping script for the Account dimension

```
entity = fdmRow.getString("ENTITY")
account = fdmRow.getString("ACCOUNT")
if (entity == u"グローバル" and account == "1110"):
fdmResult = u"Global テスト"
elif (entity == u"ローカル" and account == "1110"):
fdmResult = u"Local テスト"
else:
fdmResult = account
```

The scripts above uses the "u"prefix for the user defined strings. You can optionally specify the u prefix for English/ASCII strings (that is, you can use "1110" or u"1110"). The following shows the result of the mapping applied on the workbench.

View 🗸	Format Show	Download Template	9 ▼ (38	×		F
			1110*	a educháha tanana a				
*	Source-Company	Entity	Source-A	ccount		Accour	nt	
	ローカル	SRE_ローカル	1110			Local 5	テスト	
	グローバル	SRE_グローバル	1110			Global	テスト	
	ローカル	SRE_ローカル	1110			Local 5	가	

Using JAVA IDE to Develop Scripts

You can use popular Java IDE tools like Oracle jDeveloper, or Eclipse to develop and test scripts. Before using Eclipse to develop scripts you must install and configure the PyDev Interpreter. Refer to http://pydev.org for more details. After you have configured the IDE environment, copy the following JAR files from the EPM Server where FDMEE is installed (File location EPM_ORACLE_HOME/products/FinancialDataQuality/lib)

- 1. aif-apis.jar
- 2. aif-custom.jar

In addition, download appropriate (Oracle or SQL Server) JDBC driver JAR. After you have copied these files to the Project working directory, include them in the Project you create. Below is sample of the initialization steps required when running from your selected IDE:

```
#Start Initialize Code
#Required for Dev Mode. Not required in production script
import java.math.BigDecimal as BigDecimal
import java.sql as sql
import com.hyperion.aif.scripting.API as API

fdmAPI = API()
conn = None
conn = sql.DriverManager.getConnection("jdbc:oracle:thin:@server:1521:orcl", "user",
"password");
conn.setAutoCommit(False)
```

```
fdmAPI.initializeDevMode(conn);
print "SUCCESS CONNECTING TO DB"
fdmContext = fdmAPI.initContext(BigDecimal(1720))
#End Initialize Code Required for Dev Mode. Not required in production script
#Code to be added to production script
print fdmContext["LOCNAME"]
print fdmContext["LOCKEY"]
print fdmContext["APPID"]
```

Using Visual Basic API and Samples

Overview

This section includes:

- "Visual Basic API List" on page 298
- "Visual Basic Sample Scripts" on page 315

Visual Basic API List

The API list below shows the objects you can use to programmatically extend FDMEE using Visual Basic.

API Class (clsAppServerDM)

fdmAPI.API Call

Table 45 API Class (clsAppServerDM)

API Call	Method/ Property/ Object Reference	Description
fInitializeMainAPI	Method	Initialize the VB API. This function must be called first in order to use the VB API.
		Input: ByVal LoadID As String, ByVal Token As String, ByVal OracleInstancePath As String, ByVal JavaHomePath As String, ByVal DatabaseProvider As String
		Output: True or False
URLDecodeString	Method	Decode a URL Encoded string.
		Input: ByVal value as String
		Ouput: String
Dispose	Method	Initiates the .Net garbage collection process to destroy the API object

Application Class (clsApp)

fdmAPI.API.DataWindow.Connection.PobjApp.API Call

 Table 46
 Application Class (clsApp)

API Call	Method/Property/Object Reference	Description		
PstrConnInfoCacheFile	Property	Returns the path to the ccs file stored in the user projects directory under the FDM folder. This file contains the connection information returned by COMJNI. This file is encrypted.		
PstrConnInfoTempFilePath	Property	Path of ccs file only		
PstrDatabaseName	Property	Database name		
PstrDatabasePassword	Property	Database password		
PstrDatabaseServer	Property	Database server		
PstrDatabaseUserName	Property	Database user name		
PstrDataDir	Property	Data directory for the LoadID		
PstrDBConnectionString	Property	DB connection string returned from COMJNI		
PstrDBPort	Property	DB port		
PstrDBProvider	Property	DB provider		
PstrInboxDir	Property	Inbox directory for Load ID		
PstrJavaHomePath	Property	Java Home Path Load ID		
PstrLoadID	Property	Load ID		
PstrOracleHomePath	Property	Oracle Home		
PstrOracleInstancePath	Property	Oracle Instance Path		
PstrOutboxDir	Property	Outbox directory for Load ID		
PstrScriptsDir	Property	Scripts directory for Load ID		
PstrToken	Property	Token		
PstrTopLevelDir	Property	Top level directory for Load ID		

Connection Class (clsConnection)

fdmAPI.API.DataWindow.Connection.API Call

 Table 47
 API.DataWindow.Connection

API Call	Method/Property/Object Reference	Description		
GingErrNo	Property	Contains the error number in the event of an error.		
GstrError	Property	Contains the error number in the event of an error.		
PblnEncodingUnicode	Property	Returns "True" when set to UTF-8, but "false" for all others.		
PbInUseTrans	Property	This property defaults to "False." If it is set to "True," the query executed in ExecuteDML is executed as a transaction.		
PobjApp	Object Reference	Reference to the clsAPP class		
PstrConnection	Property	Returns the connection string to FDMEE.		
PstrDatabase	Property	Connection string database catalog name		
PstrDatabaseUserID	Property	Contains the database user ID.		
PstrDirData	Property	Returns the path to the Data directory.		
PstrDirInbox	Property	Returns the path to the Inbox directory.		
PstrDirOutbox	Property	Returns Outbox directory based on LoadID.		
PstrDirReport	Property	Returns blank. Needs to point to install the "templates" directory.		
PstrDirScripts	Property	Returns Scripts directory for LoadID.		
PstrLanguageCode	Property	Returns FDMEE API language code. Returns "en" (for English) only. This property cannot be changed in the user interface.		
PstrProvider	Property	Returns the provider used by FDMEE API.		
PstrQueryDateMask	Property	Get/Set Query Date mask which is used through the API for queries involving date fields. The default is "yyyMMdd".		
PstrResourceFileDirectory	Property	Points to the FDM API localization files directory path.		
PstrServer	Property	Prints the database server name.		
fGenerateRnd	Method	Returns the random number (short type).		
fGetOSUserID	Method	Returns the OS user of the current process.		

DataManipulation Class

fdmAPI.API.DataWindow.DataManipulation.API Call

 Table 48
 DataManipulation Class

API Call	Method/Property/Object Reference	Description
fExecuteDML	Method	Execute DML Query. Inputs: ByVal strSQL As String Output: Boolean
PingRecordsAffected	Property	Returns the number of records affected by the query executed in fExecuteDML

Utilities Class (clsUtilities.cls)

fdmAPI.API.DataWindow.Utilities.API Call

Table 49 Utilities Class (clsUtilities.cls)

API Call	Method/ Property/ Object Reference	Description
fParseString	Method	Extract a substring from a delimited string based on its field position within the total number of fields. Inputs: Inputs: ByVal strRecord As String, ByVal intFldCount As Integer, ByVal intFldToReturn As Integer, ByVal strDelimiter As String Output: String
fGetRandomNumber	Method	Get a random number based on a two integer range. Inputs: ByRef intUpper As Short, ByRef intLower As Short Output: Short

Data Access Class (clsDataAccess)

fdmAPI.API.DataWindow.DataAccess.API Call

Table 50 Data Access Class (clsDataAccess)

API Call	Method/ Property/ Object Reference	Description
farsFireHose	Method	Open an ADO forward only/read only recordset for fast data streaming.
		Inputs: ByVal strSQL As String, Optional ByVal blnServerCursor As Boolean, Optional ByVal IngMarshalType As Long
		strSQL—SQL statement to execute
		blnServerCursor—Optional, Cursor location switch (True=Database Server,False=Client)
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB Recordset
farsKeySet	Method	Open an ADO keyset type recordset based on the specified lock type.
		Inputs: Inputs: ByVal strSQL As String, Optional ByVal IngLockType As Long, Optional ByVal blnServerCursor As Boolean, Optional ByVal IngMarshalType As Long
		strSQL—SQL statement to execute
		 IngLockType—ADO LockTypeEnum (adLockReadOnly=1;adLockPesimistic=2;adLockOptimistic=3;adLockBatch Optimistic=4)
		blnServerCursor—Optional, cursor location switch (True=Database Server,False=Client)
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB.Recordset
farsPage	Method	Open an ADO BatchOptimistic recordset limited to the records found on the specified page. The recordset returned is programmatically created and is disconnected.
		Inputs: ByVal strSQL As String, ByVal IngPageSize As Long, ByVal IngPage As Long, IngOPRecordCount As Long, Optional ByVal IngMarshalType As Long
		strSQL—SQL statement to execute
		IngPageSize—Number of records per page
		IngPage—Desired page number
		IngOPRecordCount—Output parameter containing recordset record count
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB.Recordset

API Call	Method/ Property/ Object Reference	Description
farsPageFind	Method	Open an ADO BatchOptimistic recordset limited to the records found on the specified page and for the specified criteria. The recordset returned is programmatically created and is disconnected.
		Inputs: ByVal strSQL As String, ByVal strCriteria As String, ByVal IngStartPosition As Long, ByVal IngPageSize As Long, IngOPRecordCount As Long, IngOPFindPostion As Long, Optional ByVal IngMarshalType As Long
		strSQL—SQL statement to execute
		strCriteria—ADO recordset FIND criteria statement (See ADO FIND method)
		IngStartPosition—Record index to begin the search
		IngPageSize—Number of records per page
		IngPage—Desired page number
		IngOPRecordCount—Output parameter containing recordset record count
		 IngOPFindPostion—Output parameter containing position where record was found
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB.Recordset
farsPageFindFromKey	Method	Open an ADO BatchOptimistic recordset limited to the records found on the specified page and for the specified criteria. The recordset returned is programmatically created and is disconnected.
		Inputs: ByVal strSelect As String, ByVal strFrom As String, ByVal strWhere As String, ByVal strGroupBy As String, ByVal strHaving As String, ByVal strOrderBy As String, ByVal strKeyFieldName As String, ByVal strFindFieldName As String, ByVal strCriteria As String, ByVal IngStartPosition As Long, ByVal IngPageSize As Long, IngOPRecordCount As Long, IngOPFindPostion As Long, Optional ByVal
		IngMarshalType—As Long
		strSelect –Select clause of SQL statement to execute
		strFrom—From clause of SQL statement to execute
		strWhere—Where clause of SQL statement to execute
		strGroupBy GroupBy—Clause of SQL statement to execute
		strHaving—Having clause of SQL statement to execute
		strOrderBy OrderBy—Clause of SQL statement to execute
		strKeyFieldName—Name of key field used for fast page location
		strFindFieldName—Name of the field used in the find criteria
		strCriteria—Find criteria statement
		IngStartPosition—Record index to begin the search
		IngPageSize—Number of records per page
		IngPage—Desired page number
		IngOPRecordCount—Output parameter containing recordset record count
		IngOPFindPostion—Output parameter containing position where record was found
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB.Recordset

API Call	Method/ Property/ Object Reference	Description
farsPageFromKey	Method	Open an ADO BatchOptimistic recordset limited to the records found on the specified page. This method is optimized for high volume queries. The recordset returned is programmatically created and is disconnected.
		Inputs: ByVal strSelect As String, ByVal strFrom As String, ByVal strWhere As String, ByVal strGroupBy As String, ByVal strHaving As String, ByVal strOrderBy As String, ByVal strKeyFieldName As String, ByVal IngPageSize As Long, ByVal IngPage As Long, IngOPRecordCount As Long, Optional ByVal IngMarshalType As Long
		strSelect—Select clause of SQL statement to execute
		strFrom —From clause of SQL statement to execute
		strWhere—Where clause of SQL statement to execute
		strGroupBy—GroupBy clause of SQL statement to execute
		strHaving—Having clause of SQL statement to execute
		strOrderBy—OrderBy clause of SQL statement to execute
		strKeyFieldName—Name of key field used for fast page location
		IngPageSize—Number of records per page
		IngPage—Desired page number
		IngOPRecordCount—Output parameter containing recordset record count
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Output: ADODB.Recordset
farsSnap	Method	Open an ADO static type read only recordset.
		Inputs: ByVal strSQL As String, Optional ByVal blnServerCursor As Boolean, Optional ByVal IngMarshalType As Long
		strSQL—SQL statement to execute
		blnServerCursor—Optional, cursor location switch (True=Database Server,False=Client)
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Outputs:ADODB.Recordset
farsTableAppend	Method	Open an ADO keyset type recordset based on the specified lock type using the ADO "Table Direct" command directive and with the "Append-Only Rowset" property set to true.
		Inputs: ByVal strTableName As String, Optional ByVal IngMarshalType As Long
		strTableName—Name of table to open
		IngMarshalType—Optional, Value used to control how the recordset is returned
		Outputs: ADODB.Recordset

SQL Manager Class (clsSQLMgr)

fdmAPI.API.SQLMgr.API Call

Table 51 SQL Manager Class (clsSQLMgr)

API Call	Method/ Property/ Object Reference	Description
flmpFieldsForGroup	Method	Returns a recordset containing all of the records ("Mappings") for the Import Format name (string) passed in at runtime.
		Inputs: ByVal strGroupKey As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		strGroupKey—Name of Import group key to return field parsing instructions for
		• blnReadOnly—Optional, Flag to create a read only recordset (Type=farsSnap)
		blnSQLOnly—Optional, Switch for generating SQL only
		IngMarshalType—Optional, Value used to control how the recordset is returned
flmpGroup	Method	Returns a recordset containing all records in TBHVIMPGROUP table for the Import Format name (string) passed in.
		Inputs: ByVal strGroupKey As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		strGroupKey—Name of Import group key to return field parsing instructions for
		blnReadOnly—Optional, Flag to create a read only recordset (Type=farsSnap)
		blnSQLOnly—Optional, Switch for generating SQL only
		IngMarshalType—Optional, Value used to control how the recordset is returned
flmpGroupList	Method	Returns a recordset containing all import format groups based on group type.
		Inputs: ByVal IngGroupType As Integer, Optional ByVal bInIncludeNone As Boolean = False, Optional ByVal bInReadOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fIntDimensionList	Method	Returns a recordset of dimensions used in the Target Application.
		ByVal strApplicationID As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fLoadDetails	Method	Return load details for specified Load ID.
		Inputs: ByVal strLoadID as String, Optional ByVal bInReadOnly as Boolean = False, Optional ByVal bInSQLOnly as Boolean = False, Optional ByVal IngMarshalType as Integer = 0
		Output: ADODB.Recordset

API Call	Method/ Property/ Object Reference	Description
fPartition	Method	Returns a recordset containing all of the properties of the location passed into the function.
		ByVal strLoc As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fPartitionList	Method	Returns a recordset containing all properties of all locations that exist.
		Inputs: Optional ByVal strLocNoRelatives As String = "", Optional ByVal strLocChildren As String = "", Optional ByVal blnNoParent As Boolean = False, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal lngMarshalType As Integer = 0
		Output: ADODB.Recordset
fPeriodKeyList	Method	Returns a recordset containing the list of period keys for the loadid passed to the function.
		Inputs: ByVal strLoadID As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.RecordsetOutput: ADODB.Recordset
fPOVDataValue	Method	Return the data value for the specified partition key.
		Inputs: ByVal strPartitionKey As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fProcessBallmpFile	Method	Return a recordset containing the name of the imported file for a given load id. For all non-file based imports, this function will return an empty string.
		Inputs: ByVal strLoadID As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fProcessLogForPOV	Method	Return a single process status record for the specified POV value/intersection.
		Inputs: ByVal strLoc As String, ByVal strCat As String, ByVal strPer As String, ByVal strRuleID As String, Optional ByVal bInReadOnly As Boolean = False, Optional ByVal bInSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fTrialBalanceConverted	Method	Returns a recordset containing all records pulled from: tDataSeg based on LoadID, Loc, Scenario, and Period.
		Inputs: ByVal strLoadID As String, ByVal strLoc As String, ByVal strCat As String, ByVal strPer As String, Optional ByVal blnExcludeIgnore As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngPageSize As Integer = 0, Optional ByVal IngPage As Integer = 0, Optional ByVal IngOPRecordCount As Object = Nothing, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset

API Call	Method/ Property/ Object Reference	Description
fTrialBalanceLite	Method	Return a recordset listing the DataKey field from the data segment table assigned to the specified location. The data is returned based on the specified POV values and calculation types. The purpose of this recordset is to create a fast/light weight recordset that can be used for executing a looping cursor delete.
		nput: ByVal strLoadID As String, ByVal strLoc As String, ByVal strCat As String, ByVal strPer As String, ByVal strCalcTypes As String, Optional ByVal IngStartCalcType As Integer = 0, Optional ByVal IngEndCalcType As Integer = 0, Optional ByVal strJournalID As String = "", Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fValEntGroupList	Method	Returns a recordset containing all validation entity group records.
		Input: Optional ByVal bInIncludeNone As Boolean = False, Optional ByVal bInReadOnly As Boolean = False, Optional ByVal bInSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fValEntitiesForGroup	Method	Return a recordset containing validation entity specifications for the specified validation entity group.
		Inputs: ByVal strGroupKey As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False,Optional ByVal lngMarshalType As Integer = 0
		Output: ADODB.Recordset
fValGroup	Method	Return a recordset containing a single validation rule group.
		Inputs: ByVal strGroupKey As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset
fValGroupList	Method	Return a recordset containing all validation rule group records.
		Inputs: Optional ByVal bInIncludeNone As Boolean = False, Optional ByVal bInReadOnly As Boolean = False, Optional ByVal bInSQLOnly As Boolean = False, Optional ByVal IngMarshalType As Integer = 0
		Output: ADODB.Recordset

SQL Class (clsSQL)

fdmAPI.API.DataWindow.SQL.API Call

Table 52 SQL Class (clsSQL)

API Cali	Methods/ Properties/ Object Reference	Description
fGetAppID	Method	Return the application ID for the Load ID passed in. Inputs: ByVal strLoadID As String Output: String
fGetLanguageCode	Method	Returns the language code. Inputs: ByVal AppID As String Output: String
fGetTopLevelDirectory	Method	Return top level directory. Inputs: ByVal AppID As String Output: String
fOptionLlstForKey	Method	Returns a recordset containing one record with values for the specified option name. Inputs: ByVal strOptionName As String, Optional ByVal blnReadOnly As Boolean = False, Optional ByVal blnSQLOnly As Boolean = False, Optional ByVal lngMarshalType As Integer = 0, Optional ByVal blnTranslateCaptions As Boolean = False Output: ADODB.Recordset

DB Tools Class (clsDBTools)

fdmAPI.API.DataWindow.DBTools.API Call

 Table 53
 DB Tools Class (clsDBTools)

API Call	Methods/ Properties/ Object Reference	Description
fGlobalOptionExists	Method	Determine when the global system option exists. Inputs: ByVal strOptionName As String Output: Boolean
fGlobalOptionGet	Method	Retrieve a global system option as a string containing the current value of the option. Inputs: ByVal strOptionName As String Output: String

API Call	Methods/ Properties/ Object Reference	Description
mLog	Method	Write an entry to the database activity log (AIF_PROCESS_LOGS table).
		Inputs: ByVal intProcessID As Integer, ByVal strEntityType As String, ByVal strEntityName As String, ByVal intLogSequence As Integer, ByVal strLogMessage As String
		Output: None
mLogError	Method	Write an error to a user runtime error log. The log is located in the outbox directory of the application under the file name "LoadID_FDM_API.log".
		Inputs: ByVal IngErrorCode As Integer, ByVal strErrorDesc As String, ByVal strErrorSource As String
		Output: None

Maintenance Manager Class (clsMaintenanceMgr)

fdmAPI.API.MaintenanceMgr.API Call

 Table 54
 Maintenance Manager Class (clsMaintenanceMgr)

API Call	Methods/ Properties/ Object Reference	Description
fProcessStatus	Method	Retrieve the current workflow process status for the specified POV values. The process status is returned as a COM object of the type "objProcessStatus". The best way to determine the current status is to check the IngStatus property of the return object, which contains a value that can be checked against the enumeration "enmProcessState".
		Inputs: ByVal strLoc As String, ByVal strCat As String, ByVal strPer As String, ByVal strRuleID As String
		Output: objProcessStatus object
		objProcessStatus Properties:
		• strLoc
		• strCat
		• strPer
		• blnlmp
		• strImp
		blnTBVal
		strTBVal
		● blnExp
		● strExp
		blnEntLoad
		strEntLoad
		blnEntVal
		strEntVal
		IngStatus
		dteLastActive
		• blnCert
		• strCert
		• blnAsses
		• strAsses
		• blnChildDone
		strChildDone
		• blnUD1
		• strUD1
		• blnUD2
		• strUD2
		• blnUD3
		• strUD3

API Call	Methods/ Properties/ Object Reference	Description
		• blnUD4
		• strUD4
		• blnValDirty
		blnWCDirty
		blnLogicDirty
		intLockState

POV Manager Class (clsPOVMgr)

fdmAPI.API.POVMgr.API Call

Table 55 POV Manager Class (clsPOVMgr)

API Call	Methods/ Properties/ Object References	Description
fCategoryKey	Method	Return the numerical category key for the specified category name. Inputs: ByVal strCat As String Output: Integer
fCategoryName	Method	Return the text name of a category for the specified category key. Inputs: ByVal IngCatKey As Integer Output: String
fCheckLocParent	Method	Check when the requested location is assigned as the MAPPING parent to any other location. Inputs: ByVal strLoc As String Output: Boolean

API Call	Methods/ Properties/ Object References	Description
fCheckLocProf	Method	Return all behavior attributes for the specified location as a COM object of type "objLocProfile". This object also contains the key information of the location and the data segment assignment.
		Inputs: ByVal strLoc As String
		Output: objLocProfile
		objLocProfile Properties:
		blnAdapterImport
		• blnChild
		● blnExists
		blnScriptImport
		● blnSeq
		IngControlsType
		● IngLocKey
		IngParentLocKey
		IngRedFlagLevel
		● IngSegKey
		IngType
		strAdaptor
		strCLogicGrp
		strControlsApprover
		strControlsApproverProxy
		• strControlsGrp1
		• strControlsGrp2
		strControlsParent
		• strCurrency
		• strDataValue
		• strImportGrp
		• strIntgConfig1
		• strIntgConfig2
		• strIntgConfig3
		strintgConfig4 otal agisCra
		strLogicGrpstrNotes
		• strivities • strParent
		strValEntityGrp
		strValRuleGrp
fGetLocationName	Method	Return the text name of a location based on its numeric key. Inputs: ByVal IngPartKey As Integer
		Output: String

API Call	Methods/ Properties/ Object References	Description
fPeriodKey	Method	Return attributes associated with the specified period value as a COM object of type "objPeriod". This procedure can also be used to return the prior period of a location by setting the optional parameter "blnPriorPer" to true. By default the procedure returns the target period associated with the 'M' or 'Monthly' frequency, but the optional parameter "strFreq" can be used to retrieve a target value associated with a different frequency.
		Inputs: ByVal strPer As String, Optional ByVal bInPriorPer As Boolean = False, Optional ByVal strFreq As String = ""
		Output: objPeriod
		objPeriod Properties:
		dblSerialKey
		dteDateKey
		• strDateKey
		strSerialKey
		strTarget
		strTargetYear

State Class (clsState)

API State

Table 56 State Class (clsState)

API Call	Methods/Properties/Oject Reference	Description
AppID	Property	Returns the Application ID for the LoadID
bstMaintenanceMgr	Object Reference	Reference to the clsMaintenanceMgr class
bstSQLMgr	Object Reference	Reference to the clsSQLMgr class
CatKey	Property	Returns the Category Key for the LoadID.
CatName	Property	Returns the Category Name for the LoadID.
DataValue	Property	Returns the DataValue assigned to the Location for the LoadID.
dstConnection	Object Reference	Reference to the clsConnection class
dstDataAccess	Object Reference	Reference to the clsDataAccess class
dstDataManipulation	Object Reference	Reference to the clsDataManipulation class
dstDBTools	Object Reference	Reference to the clsDBTools class
dstSQL	Object Reference	Reference to the clsSQL class

API Call	Methods/Properties/Oject Reference	ect Description		
dstUtilities	Object Reference	Reference to the clsUtilities class		
EndPeriodKey	Property	Returns the End Period for the LoadID.		
ExportFlag	Property	Returns Y or N. If the Data Rule is executed with Export checked, this property returns Y.		
ExportMode	Property	Returns Y or N. f ExportFlag is Y, this property returns the selected mode for Export (Replace or Append). Otherwise, it returns an empty string.		
FileDir	Property	Returns the FILE_PATH field from the AIF_BAL_RULE_LOADS table.		
		Note: The FileDir defaults to the property you specify in the Directory field on the data load rule. If the file is in root folder, you do not need to specify anything for this field. In this case, FILEDIR returns null.		
FileName	Property	Returns a string containing the name of the file that was imported.		
GstrDateMask	Property	Returns the date mask.		
ImportFlag	Property	Returns the IMPORT_FROM_SOURCE_FLAG from the AIF_BAL_RULE_LOADS table. If LoadID initiates the Import Process, this flag is set to Y. Otherwise it is set to F.		
ImportFormat	Property	Returns the Import Format used for the Load ID being processed.		
ImportMode	Property	Returns REPLACE or APPEND based on the LoadID. This property returns null when the LoadID did not Import data (that is, Validate only),		
LoadID	Property	Returns the Load ID.		
LocKey	Property	Returns the Location Key for the Location being processed for the LoadID.		
LocName	Property	Returns the Location Name for the location being processed for the LoadID.		
MultiPeriodLoad	Property	Returns True or False. This property is True when multiple periods are being processed. Otherwise, returns False.		
PeriodKey	Property	Returns the Period Key for the LoadID.		
RuleID	Property	Returns the Rule ID for the Data Rule being processed.		
RuleName	Property	Returns the Rule Name for the Data Rule being processed.		
SourceName	Property	Returns the SOURCE_SYSTEM_NAME for the data rule.		
SourceType	Property	Returns the SOURCE_SYSTEM_TYPE for the data rule.		
TargetAppDB	Property	Returns "NA" for HFM. If the target application type is HPL then this property will return the PLAN_TYPE. If the target application type is ESSBASE this property will return the ESSBASE_DB_NAME.		
TargetAppName	Property	Returns the APPLICATION_NAME for the target system.		

Visual Basic Sample Scripts

Sample Visual Basic scripts in this section include:

- "Visual Basic Script to Send Email" on page 315
- "Visual Basic Script to Copy a File to the Inbox" on page 317
- "Visual Basic Script to Execute SQL" on page 318
- "Visual Basic Script to Access Various Properties" on page 319

Visual Basic Script to Send Email

The following script shows how to send an email:

```
'REVIEW INPUT PARAMETERS
If WScript.Arguments.Count <> 5 Then
       Wscript.Echo "You must supply 5 runtime variables." & vbNewLine & "Usage:
HFM_LOAD.vbs LoadID CSS_Token OracleInstancePath JavaHomePath OLE_DB_PROVIDER"
       WSCript.Quit -1
Else
       LoadID = Wscript.Arguments.Item(0)
       CSSToken = WScript.Arguments.Item(1)
       OracleInstancePath = WScript.Arguments.Item(2)
       JavaHomePath = WScript.Arguments.Item(3)
       Provider = WScript.Arguments.Item(4)
'Initialize FDMEE VB API
Set objFDMAPI = CreateObject("fdmapi.clsAppServerDM")
WScript.Echo "FDMEE VB API created."
blnInit = objFDMAPI.fInitializeMainAPI(CStr(LoadID), CStr(CSSToken),
CStr(OracleInstancePath), CStr(JavaHomePath), CStr(Provider))
If blnInit Then
       WScript.Echo "FDMEE VB API initialized."
Else
       WScript.Quit -1
       WScript.Echo "fInitializeMainAPI failed."
End If
WScript.Echo CBool(True)
'Sample Code
If objFDMAPI.API.POVMgr.PPOVLocation = "LOCATION_NAME" Then
       'Check Status of LoadID
       Set objProcessStatus =
objFDMAPI.API.MaintenanceMgr.fProcessStatus(objFDMAPI.API.State.GstrPOVLoc,
objFDMAPI.API.State.GstrPOVCat, objFDMAPI.API.State.GstrPOVPer)
       If objProcessStatus.blnImp Then
               'Import was successful
```

```
'Send an email
                    SendEmail("FromUser@domain.com", "ToUser@domain.com", "Subject",
"Import was successful"
          End If
'Destroy objects
objFDMAPI.Dispose
Set objFDMAPI = Nothing
Sub SendEmail(strFromAddress, strToAddress, strSubject,strEmailMsqLine)
    'Declare local variables
    Dim strFromAddress
    Dim strToAddress
    Dim strSubject
    Dim strEmailMsgLine
    Dim objMsq
    Dim objConfig
    Dim intSendUsing
    Dim strSMTPServer
    Dim intSendPort
    'Initialize Mail Server variables
    'SMTP server name
    strSMTPServer = "YourMailServer"
    'CdoSendUsing enumeration-1=use smtp on local machine, 2=use smtp over network
    intSendUsing = 2
    'SMTP port on server
    intSMTPPort = 25
    'Create CDO configuration object
    Set objConfig = CreateObject("CDO.Configuration")
    'Set Config object settings
    With objConfig.Fields
        .Item("http://schemas.microsoft.com/cdo/configuration/sendusing") =
intSendUsing
        .Item("http://schemas.microsoft.com/cdo/configuration/smtpserver") =
strSMTPServer
        .Item("http://schemas.microsoft.com/cdo/configuration/smtpserverport")=
intSMTPPort
    .Item("http://schemas.microsoft.com/cdo/configuration/smtpconnectiontimeout") = 10
        .Update
    End With
    'Create CDO message object
    Set objMsg = CreateObject("CDO.Message")
    'Assign config object to configuration property of message object
    Set objMsg.Configuration = objConfig
    'Set Message object settings and send mail
    With objMsq
         .To = strToAddress
         .From = strFromAddress
         .Subject = strSubject
         .TextBody = strEmailMsgLine
```

```
.Send
End With

'Destroy message objects
Set objMsg = Nothing
Set objConfig = Nothing
End Sub
```

Visual Basic Script to Copy a File to the Inbox

The following script shows how to copy a file to the Inbox:

```
'REVIEW INPUT PARAMETERS
If WScript.Arguments.Count <> 5 Then
    Wscript. Echo "You must supply 5 runtime variables." & vbNewLine & "Usage:
HFM LOAD.vbs LoadID CSS Token OracleInstancePath JavaHomePath OLE DB PROVIDER"
WSCript.Quit -1
Else
    LoadID = Wscript.Arguments.Item(0)
    CSSToken = WScript.Arguments.Item(1)
    OracleInstancePath = WScript.Arguments.Item(2)
    JavaHomePath = WScript.Arguments.Item(3)
  Provider = WScript.Arguments.Item(4)
'Initialize FDMEE VB API
Set objFDMAPI = CreateObject("fdmapi.clsAppServerDM")
WScript.Echo "FDMEE VB API created."
blnInit = objFDMAPI.fInitializeMainAPI(CStr(LoadID), CStr(CSSToken),
CStr(OracleInstancePath), CStr(JavaHomePath), CStr(Provider))
If blnInit Then
    WScript.Echo "FDMEE VB API initialized."
Else
    WScript.Quit -1
    WScript.Echo "fInitializeMainAPI failed."
End If
'Sample Code
If objFDMAPI.API.State.LocName = "MT COMMA7" Then
    'Copy file to Inbox
    *************************
* * *
    strFileDir = "C:\"
    strFileName = "CopyThisFile.txt"
    'Create FileSystemObject
    Set objFSO = CreateObject("Scripting.FileSystemObject")
```

Visual Basic Script to Execute SQL

The following script shows how to execute SQL:

```
'REVIEW INPUT PARAMETERS
If WScript.Arguments.Count <> 5 Then
     Wscript. Echo "You must supply 5 runtime variables." & vbNewLine & "Usage:
HFM_LOAD.vbs LoadID CSS_Token OracleInstancePath JavaHomePath OLE_DB_PROVIDER"
     WSCript.Quit -1
Else
        LoadID = Wscript.Arguments.Item(0)
     CSSToken = WScript.Arguments.Item(1)
        OracleInstancePath = WScript.Arguments.Item(2)
     JavaHomePath = WScript.Arguments.Item(3)
     Provider = WScript.Arguments.Item(4)
End If
'Initialize FDMEE VB API
Set objFDMAPI = CreateObject("fdmapi.clsAppServerDM")
WScript.Echo "FDMEE VB API created."
blnInit = objFDMAPI.fInitializeMainAPI(CStr(LoadID), CStr(CSSToken),
CStr(OracleInstancePath), CStr(JavaHomePath), CStr(Provider))
If blnInit Then
     WScript.Echo "FDMEE VB API initialized."
Else
     WScript.Quit -1
     WScript.Echo "fInitializeMainAPI failed."
End If
'Sample Code
If objFDMAPI.API.State.LocName = "MT_COMMA7" Then
     'Delete one record from tdataseg
     strDeleteSQL = "DELETE FROM TDATASEG_T WHERE LOADID = " & CStr(LoadID) & " AND
ENTITY = 'EastSales'"
     objFDMAPI.API.DataWindow.DataManipulation.fExecuteDML CStr(strDeleteSQL)
End If
```

```
'Destroy objects
objFDMAPI.Dispose
Set objFDMAPI = Nothing
WScript.Echo "FDMEE VB API destroyed."
Visual Basic Script to Access Various Properties
The following script shows how to access various properties:
'REVIEW INPUT PARAMETERS
If WScript.Arguments.Count <> 5 Then
    Wscript.Echo "You must supply 5 runtime variables." & vbNewLine & "Usage:
HFM_LOAD.vbs LoadID CSS_Token OracleInstancePath JavaHomePath OLE_DB_PROVIDER"
    WSCript.Quit -1
Else
    LoadID = Wscript.Arguments.Item(0)
    CSSToken = WScript.Arguments.Item(1)
    OracleInstancePath = WScript.Arguments.Item(2)
    JavaHomePath = WScript.Arguments.Item(3)
    Provider = WScript.Arguments.Item(4)
'Initialize FDMEE VB API
Set objFDMAPI = CreateObject("fdmapi.clsAppServerDM")
WScript.Echo "FDMEE VB API created."
blnInit = objFDMAPI.fInitializeMainAPI(CStr(LoadID), CStr(CSSToken),
CStr(OracleInstancePath), CStr(JavaHomePath), CStr(Provider))
If blnInit Then
    WScript.Echo "FDMEE VB API initialized."
Else
    WScript.Quit -1
    WScript.Echo "fInitializeMainAPI failed."
'Sample Code
WScript.Echo "AppID = " & objFDMAPI.API.State.AppID
WScript.Echo "CatKey = " & objFDMAPI.API.State.CatKey
WScript.Echo "CatName = " & objFDMAPI.API.State.CatName
WScript.Echo "DataValue = " & objFDMAPI.API.State.DataValue
WScript.Echo "EndPeriodKey = " & objFDMAPI.API.State.EndPeriodKey
```

'Need to test this during Export process

```
WScript.Echo "ExportFlag = " & objFDMAPI.API.State.ExportFlag
WScript.Echo "ExportMode = " & objFDMAPI.API.State.ExportMode
WScript.Echo "FileDir = " & objFDMAPI.API.State.FileDir
WScript.Echo "FileName = " & objFDMAPI.API.State.FileName
'WScript.Echo "GblnPOVLocal = " & objFDMAPI.API.State.GblnPOVLocal
WScript.Echo "GdtePOVPerKey = " & objFDMAPI.API.State.GdtePOVPerKey
WScript.Echo "GlngPOVCatKey = " & objFDMAPI.API.State.GlngPOVCatKey
WScript.Echo "GlngPOVLocKey = " & objFDMAPI.API.State.GlngPOVLocKey
WScript.Echo "GlngPOVLocType = " & objFDMAPI.API.State.GlngPOVLocType
WScript.Echo "GstrDateMask = " & objFDMAPI.API.State.GstrDateMask
WScript.Echo "GstrMethodID = " & objFDMAPI.API.State.GstrMethodID
WScript.Echo "GstrMsg = " & objFDMAPI.API.State.GstrMsg
WScript.Echo "GstrPOVCat = " & objFDMAPI.API.State.GstrPOVCat
WScript.Echo "GstrPOVLoc = " & objFDMAPI.API.State.GstrPOVLoc
WScript.Echo "GstrPOVPer = " & objFDMAPI.API.State.GstrPOVPer
WScript.Echo "ImportFlag = " & objFDMAPI.API.State.ImportFlag
WScript.Echo "ImportFormat = " & objFDMAPI.API.State.ImportFormat
WScript.Echo "ImportMode = " & objFDMAPI.API.State.ImportMode
WScript.Echo "LoadID = " & objFDMAPI.API.State.LoadID
WScript.Echo "LocKey = " & objFDMAPI.API.State.LocKey
WScript.Echo "LocName = " & objFDMAPI.API.State.LocName
WScript.Echo "MultiPeriodLoad = " & objFDMAPI.API.State.MultiPeriodLoad
WScript.Echo "PeriodKey = " & objFDMAPI.API.State.PeriodKey
WScript.Echo "RuleID = " & objFDMAPI.API.State.RuleID
WScript.Echo "RuleName = " & objFDMAPI.API.State.RuleName
WScript.Echo "Scenario = " & objFDMAPI.API.State.Scenario
WScript.Echo "SourceName = " & objFDMAPI.API.State.SourceName
WScript.Echo "SourceType = " & objFDMAPI.API.State.SourceType
WScript.Echo "TargetAppDB = " & objFDMAPI.API.State.TargetAppDB
WScript.Echo "TargetAppName = " & objFDMAPI.API.State.TargetAppName
If objFDMAPI.API.POVMgr.PPOVLocation = "LOCATION_NAME" Then
      'Enter Sample Code HERE
Else
      'Enter Sample Code HERE
End IF
'Destroy objects
objFDMAPI.Dispose
Set objFDMAPI = Nothing
```

10

FDMEE Reports

In This Chapter

FDMEE Reports	321
Working with Query Definitions	
Working with Report Definitions	
Running Reports	329
FDMFF Detail Reports	331

FDMEE provides prebuilt reporting tools that capture business-critical operations and revenuegenerating activities within your organization. These reports provide key information on how metadata and data are integrated from source to target

The FDMEE reporting framework represents a unified solution that incorporates source and target data, templates, and pre-defined SQL queries. Templates, created in Oracle Business Intelligence Publisher, consume data in XML format and generate reports dynamically. You can add SQL queries to extract data from tables, or couple them with the report parameters to extend the definition of a standard report.

FDMEE reports can be generated as PDF, Excel, Word, or HTML output.

The FDMEE reporting framework has the following components:

- Query Definition
- Report Group
- Report Definition
- Report Execution

FDMEE Reports

The standard FDMEE report groups are described below. For information on the subcategories of each report, see "FDMEE Detail Reports" on page 331.

Location: KS7DIM_FILE Category: Actual Period: 2005-01-01 Currency: [NONE]

GL Account	GL Center	GL Account Description	Amount	Acct Rule	Entity Rule
Travel	EastAdmin		2,991.37	ACCT_DEFAULT	ACCT_DEFAULT
Travel	WestAdmin		1,937.34	ACCT_DEFAULT	ACCT_DEFAULT
Travel	EastSales		381.33	ACCT_DEFAULT	ACCT_DEFAULT
Travel	WestSales		1,368.48	ACCT_DEFAULT	ACCT_DEFAULT
		Total	6,678.51		
Software	EastAdmin		1,197.38	ACCT_DEFAULT	ACCT_DEFAULT
Software	EastSales		152.64	ACCT_DEFAULT	ACCT_DEFAULT
Software	WestSales		547.77	ACCT_DEFAULT	ACCT_DEFAULT
Software	WestAdmin		775.47	ACCT_DEFAULT	ACCT_DEFAULT
		Total	2,673.26		
Meals	EastAdmin	Total	1,000.96	ACCT_DEFAULT	ACCT DEFAULT
Meals	WestSales		457.92	ACCT_DEFAULT	ACCT_DEFAULT
Meals	WestAdmin		648.26	ACCT_DEFAULT	ACCT_DEFAULT
Meals	EastSales		127.60	ACCT_DEFAULT	ACCT_DEFAULT
		Total	2,234.74		_
BuildingDepr	EastAdmin	10141	1,509.86	ACCT_DEFAULT	ACCT_DEFAULT
BuildingDepr	WestSales		1,600.00	ACCT DEFAULT	ACCT DEFAULT
BuildingDepr	EastSales		3,838.17	ACCT_DEFAULT	ACCT_DEFAULT
BuildingDepr	WestAdmin		183.32	ACCT_DEFAULT	ACCT_DEFAULT
- '		Total	7,131.35	_	_
Advertising	EastAdmin	1000	1,261.62	ACCT_DEFAULT	ACCT_DEFAULT
Advertising	WestSales		577.16	ACCT_DEFAULT	ACCT_DEFAULT
Advertising	EastSales		160.82	ACCT_DEFAULT	ACCT_DEFAULT
Advertising	WestAdmin		817.08	ACCT_DEFAULT	ACCT_DEFAULT
		Total	2,816.68		
TransportationDepr	EastSales		1,584.40	ACCT_DEFAULT	ACCT_DEFAULT
TransportationDepr	WestAdmin		75.67	ACCT_DEFAULT	ACCT_DEFAULT
TransportationDepr	EastAdmin		623.27	ACCT_DEFAULT	ACCT_DEFAULT
TransportationDepr	WestSales		575.00	ACCT DEFAULT	ACCT DEFAULT

2/8/2013 12:44:20 PM Page 1 of 37

Table 57 Report Groups and Descriptions

Report Group	Description		
Audit Reports	An audit report displays all transactions for all locations that comprise the balance of a target account. The data returned in these reports depends on the location security assigned to the user.		
	The subcategories of the Audit Reports:		
	Account Chase		
	Account Chase with Wildcards		
	Intersection Drill Down		
Base Trial Balance Reports	The base Trial Balance reports represent account balance source data in a General Ledger system. You use a base Trial Balance report to validate and compare balances as data is loaded from the source General Ledger system to the target applications.		
	The subcategories of base Trial Balance reports:		
	Trial Balance Location, With Targets (Cat, Per)		
	Trial Balance Current Location, With Rules (Cat, Per)		
	 Trial Balance Current Location, All Dimensions-Target Entity-Acct (Cat, Per) 		
	Trial Balance Converted Current Location, By Target Entity-Acct (Cat, Per)		
	Trial Balance Current Location, with Target Entity-Acct (Cat, Per)		
	 Trial Balance Current Location, All Dimension-Targets (Cat, Per) 		
	Trial Balance Current Location, by Target Acct (Cat, Per)		

Report Group	Description				
Check Reports	Check reports provide information on the issues encountered when data load rules are run. Check reports return target system values that include aggregation or calculations from the target system.				
	The subcategories of Check reports:				
	Check Report—Displays the results of the validation rules for the current location (pass or fail status).				
	• Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods.				
	 Check Report by Validation Entity Sequence —Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group. 				
	Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown when warning criteria are met. This report does not show rules that passed the validation.				
Listing	Listing reports summarize metadata and settings (such as the import format, or check rule) by the current location.				
Reports	The subcategories of Listing reports:				
	Import Format by Location				
	Location Listing				
Location	Location Analysis reports provide dimension mapping and system log information by the current location.				
Analysis Reports	The subcategories of Location Analysis reports:				
Керопа	Dimension Map for POV (Dimension, Cat, Per)				
	Dimension Map (Dimension)				
Process Monitor Reports	A Process Monitor report displays, for the current report, a list of locations and their positions within the data conversion process. You can use the process monitor report to monitor the status of the closing process. The report is time-stamped. Therefore, it can be used to determine to which locations and at which time data was loaded.				
	The subcategories of Process Monitor reports:				
	Process Monitor (Cat, Per)				
	Process Monitor All Categories (Cat, Per)				
	Process Monitor with Lock (Cat, Per)				
	Process Status Period Range (Cat, Start Period, End Period)				
Variance Reports	A Variance report displays source/trial balance accounts for one target account, showing over two periods or categories.				
	The subcategories of Variance reports:				
	Account Chase Variance (Target Account, Cat1, Per1, Cat2, Per2)				
	Trial Balance Variance (Cat1, Per1, Cat2, Per2)				

Working with Query Definitions

Two types of SQL queries can be used in a report definition: a base query and a parameter query.

A base query enables users to pull data from various tables and display the data as report output. The base extends the definition of a standard report, and can be reused with multiple reports. For example, using one query definition, you can show different columns or groupings. In one report, you can list amounts by account and group by entity, and in another list amount by entity and group by account.

The parameter SQL query enables you to run a query against the parameters in the report definition. For example, the query can be used to select the Location, Period, Category, or Account.

For information on the tables and joins you can use to build your SQL queries, see "TDATASEG Table Reference" on page 387 and "TLOGPROCESS Table Reference" on page 391 in Appendix G. The TDATASEG table is used to store the data loaded by the user, and the transformation between the source dimension members and results of the mapping process. The TLOGPROCESS table is used to store the workflow process status for a location, category, and period.

You can save a query definition as an XML file, which, in turn, you can use to create custom templates using BI Publisher or the BI Publisher desktop addin for Microsoft Word.

- To add a query definition:
- 1 From the **Setup** tab, and then under **Reports**, select **Query Definition**
- 2 In Query Definition, click Add.
- 3 In Name, enter the name of the query definition.

Oracle recommends that you assign a name that corresponds to the report definition in which the SQL is embedded.

- 4 In **Select Clause**, specify the SQL Select clause used to query the database and return the data that corresponds to your selected criteria.
- In **Where Clause**, specify the SQL Where clause used to restrict the data that is returned to only the specific criteria that you specify.
- 6 In Group by/Order by Clause, specify the Group by or Order by clause.

The ORDER BY clause sorts the records in the result set. The ORDER BY clause can be used only in SQL SELECT statements.

The GROUP BY clause fetches data across multiple records and returns the results grouped by one or more columns.

7 Click Validate Query.

If the query definition is validated, FDMEE returns the message: "Query validation successful."

If the query definition is not validated, FDMEE indicates that an error was found in the SQL. You must fix the error before validating the query again.

- 8 Click Save.
- 9 **Optional**: To save the query definition to an XML file, click **Generate XML**.

Working with Report Definitions

Report definitions are the attributes that determine the content and structure of a report. Using the Report Definition option, you can:

- Associate a report with a report group.
- Associate a query definition with the report.
- Associate a Rich Text Format template (RTF) with the report.
- Define report parameters.

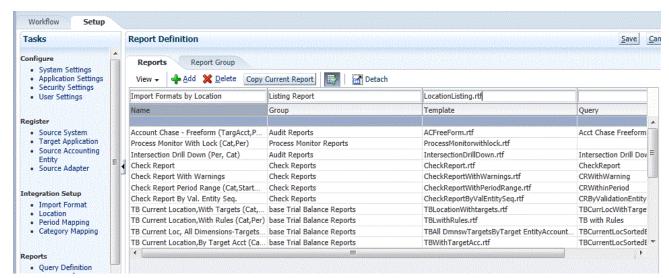
Adding Report Groups

Report groups enable you to assemble similar types of reports into one category for ease of use.

- To add a report group:
- 1 From the Setup tab, and then under Reports, select Report Definition.
- 2 In Report Definition, select the Report Group tab.
- 3 In Report Group, click Add.
- 4 In the blank **Name** field, enter the title of the report group.
 - For example, enter "Base Trial Balance Reports."
- In **Sequence**, enter a numeric value that identifies the display order of the report group on the Report Definition screen.
 - For example, to display the Check Report group before the Location Listing group, enter 1 in the Sequence field for the Check Report group, and 2 for the Location Listing group.
- 6 Click Save.

Working with Report Definitions

A report definition can include a query definition created in SQL and any RTF template set up for use with a selected report. Additionally, you can associate a report with the query definition that prompts the user with a list of values when selecting the parameters of the report.



- To add a report definition:
- 1 Select **Setup**, and then under **Reports**, select **Report Definition**.
- 2 In Report Definition, select the Reports tab.

The Report tab consists of three regions:

- Summary—Lists all report definitions.
- Details— Shows the report name, associated base query definition, report group, and associated template.
- Parameters—Shows the parameter name and type, display name, sequence, parameter value, and any query definition used to supply a list of values for a given report parameter.
- 3 In the summary grid, click Add.
- 4 In the detail grid, in Name, enter the name of the report definition.
- 5 In **Group**, select the report group name associated with the definition.

To search on a report group, click and choose a report group from the Search and Select: Group screen.

Report groups are created on the Report Group tab. See "Adding Report Groups" on page 325.

6 In Query, select the name of the query definition to associate with the report definition.

To search on a query definition, click and choose a query definition from the Search and Select: Query screen.

Query definitions are defined in the Query Definition option. See "Working with Query Definitions" on page 323.

7 In **Template**, select the RTF template to associate with the report definition.

To search on a template, click and choose a template from the Search and Select: Template screen.

FDMEE report templates are saved with a Rich Text Format (RTF) file type. The RTF file type is a specification used by common word processing applications, such as Microsoft Word. The templates available to the FDMEE report are created in BI Publisher. See the *Report Designer's Guide for Oracle Business Intelligence Publisher* for more information. To download the Oracle BI Publisher Desktop for Microsoft Office, see http://www.oracle.com/technetwork/middleware/bi-publisher/downloads/index.html.

To upload a template, click . In **Search and Select: Template**. click **Upload**. Then click **Browse** to navigate to the template to upload and click **OK**.

8 Click Save.

- To define the parameters of the report definition:
- 1 From the summary grid, select the name of the report definition to which to add parameters.
- 2 In the detail grid, click Add.

A blank line is displayed to which to add report parameters.

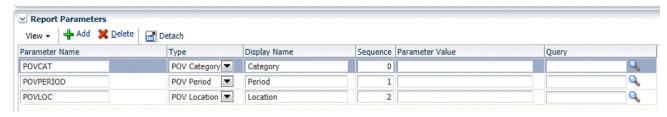
3 In Parameter Name, enter the POV category to which the parameter belongs.

For example, enter POVLOC, for the POV location, POVPeriod for the POV period, POVCAT for POV category, and POVTARGET for the target category.

4 In **Type**, select the type of parameter for the report.

Available types:

- POV—Prompts for the POV location, period, category or rule. The POV values default from the users POV when they run the report parameter.
- Query—The Query type allows you to create a query that populates a list of values that a user can select from when running the report. The parameter name entered by the user must match a parameter used in the assigned query. For example, when the parameter name is "California", a corresponding parameter in the query must be named California. The name in the query must have a "~" character as a prefix so that the system can parse the query correctly.
- Static—A Static type Indicates that the parameter is a constant or a predefined value, and the parameter value is supplied by the user. When the parameter is supplied and the report is executed, the query uses this parameter as part of the Select statement.
- Title (indicates the header part of column title)



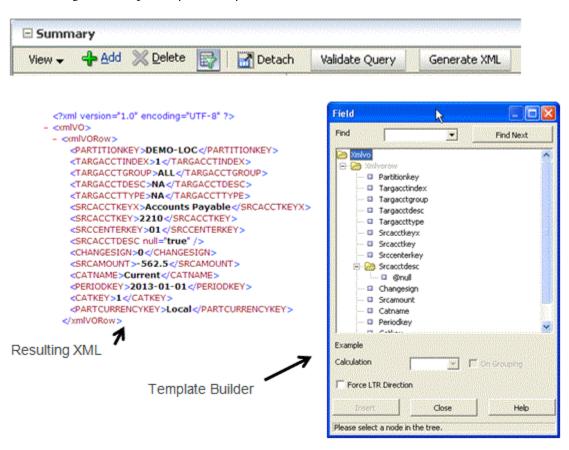
- 5 In Display Name, enter the name to display for the parameter on the Generate Report screen.
- 6 In **Parameter Value**, enter the value for the "Static" parameter type. For example, when you have an aging report for 30 days, enter 30 in this field.
- 7 In **Sequence**, enter the display order of the parameter on the Generate Report screen.
- 8 In Query, specify the query definition that provides a list of values for the parameter. For example, to show a list of data rules for the user to select in the Generate Report screen, specify the query definition to define these queries.
- 9 Click Save.
- To copy a report:
- 1 Select **Setup**, and then under **Reports**, select **Report Definition**.
- 2 In Report Definition, in the Report summary grid, select the report.

3 In the Report summary grid, click Copy Current Report.

The copied report is added to the list of reports. The name of the report takes the original report name appended with "_copy".

Creating a Report Template

Report templates, created in Oracle Business Intelligence Publisher, consume data in XML format and generate reports dynamically.



- To create a report template:
- 1 Create a query definition for the report.
 - a. From the **Setup** tab, and then under **Reports**, select **Query Definition**
 - b. In Query Definition, click Add.
 - c. In **Name**, enter the name of the query definition.
 - Oracle recommends that you assign a name that corresponds to the report definition in which the SQL is embedded.
 - d. In **Select Clause**, specify the SQL Select clause used to query the database and return the data that corresponds to your selected criteria.

- e. In **Where Clause**, specify the SQL Where clause used to restrict the data that is returned to only the specific criteria that you specify.
- f. In **Group by/Order by Clause**, specify the Group by or Order by clause.

The ORDER BY clause sorts the records in the result set. The ORDER BY clause can be used only in SQL SELECT statements.

The GROUP BY clause fetches data across multiple records and returns the results grouped by one or more columns.

g. Click Validate Query.

If the query definition is validated, FDMEE returns the message: "Query validation successful."

If the query definition is not validated, FDMEE indicates that an error was found in the SQL. You must fix the error before validating the query again.

- h. Click Save.
- 2 Click Generate XML.
- 3 Create the report template.
 - a. Install the BI Publisher Desktop.

To download the To download the Oracle BI Publisher Desktop for Microsoft Office, see http://www.oracle.com/technetwork/middleware/bi-publisher/downloads/index.html.

b. Load the XML created in step 2.

Reference template builder tutorial as needed.

- c. Save the template and then upload it to the server.
- 4 Create report definition with group, query and template.

Running Reports

- To run reports:
- 1 From the Workflow tab, and then under Other, select Report Execution.
- 2 In Report Execution, in Report Groups, select a report group.
- 3 In Reports, select a report.

To filter the display listing by a report name within a report group, enter the name of the report in the blank entry line above the **Name** field and press Enter. For example, to view only reports beginning with **Account**, enter **Account** and press Enter.

To filter the display listing by a base query name within a report group, enter the query name in the blank entry line above **Query**.

- 4 Click Execute.
- 5 When prompted, enter parameter values on the Generate Report screen.

- a. If applicable, modify the Period, Category, and Location values.
- b. From the Report Output Format, select the output format.

Available output formats are:

- PDF
- HTML
- EXCEL (.XLS)
- c. From Execution Mode, select the online method of running the report.

The online method processes the report immediately.

- d. **Optional:** To create a report batch script that generates a report when the scripts are executed, click **Create Report Script**.
- e. Click OK.
- To create a report script:
- 1 From the Workflow tab, and then under Other, select Report Execution.
- 2 In **Report Execution**, in **Report Groups**, select a report group.
- 3 In Reports, select a report.

To filter the display listing by a report name within a report group, enter the name of the report in the blank entry line above the **Name** field and press Enter. For example, to view only reports beginning with **Account**, enter **Account** and press Enter.

To filter the display listing by a base query name within a report group, enter the query name in the blank entry line above **Query**.

4 Click Create Report Script.

Note: Make sure that password encryption has been set up before creating a report script. See "Using Password Encryption" on page 261

- 5 From the **Generate Report Script** screen, select the parameter values.
- 6 From **Report Output Format** drop down, select the output format of the batch.
- 7 Click OK.

A Report Batch Script window is displayed showing the parameters of the reports script.

Scripts are located in the EPM Oracle Instance/FinancialDataQuality directory

8 Copy the script that FDMEE generates, and on a Window command window or UNIX shell, paste the script and then run it.

For example at a Windows command line, you might specify:

```
runreport.bat <username><password> "Check Report With Warnings" "Category=Actual" "Period=March-05" "Location=PKA_TEST" "Report Output Format=PDF"
```

Note: When passing program arguments for a batch file execution, Jython removes double quotes (") unless arguments have a leading space in them. Jython uses double quotes for escaping. To avoid conflicts, add a leading space in the argument. For example, instead of passing "Period=Mar-2003", pass "Period=Mar-2003".

FDMEE Detail Reports

The following reports are available in FDMEE.

Audit Reports

An audit report displays all transactions for all locations that compose the balance of a target account. The data returned in this report depends on the location security assigned to the user.

Account Chase Wildcard (TargAcct, Per, Cat)

Shows imported accounts for all FDMEE locations, subtotaled by FDMEE location, based on an account selection that allows use of wildcards.

Runs for

All FDMEE locations

Parameters

Target account, Period, Category

Query

Account Chase Wildcard

Template

Account Chase WildCard.rtf

Account Chase - Freeform (TargAcct, Per, Cat)

Shows one imported account for all FDMEE locations; subtotaled by FDMEE location.

Runs for

All FDMEE locations

Parameters

Target account, Period, Category

Query

Account Chase Freeform

Template

Account Chase Free Form.rtf

Intersection Drill Down (Per, Cat)

Shows target accounts and amounts; and includes drill-down list of source accounts and amounts that are mapped to target accounts.

Runs for

Current FDMEE location

Parameters

Period, Category

Query

Intersection Drill Down

Template

Intersection Drill Down.rtf

Check Reports

Check reports provide information on the issues encountered when data load rules are run. Note that Check reports return target system values that include aggregation or calculations from the target system.

Check Report

Shows the results of the validation rules for the current location (indicates pass or fail status).

Runs for

Current FDMEE location

Parameters

Period, Location and Category

Query

Check Report

Template

Check Report.rtf

Check Report Period Range (Cat, Start Per, End Per)

Shows the results of the validation rules for selected periods.

Runs for

Current FDMEE location

Parameters

Category, Start Period, End Period

Query

Check Report Within Period Query

Template

Check Report With Period Range.rtf

Check Report With Warnings

Shows the results of the validation rules for the current location (warnings are recorded in validation rules).

Runs for

Current FDMEE location

Parameters

None

Query

Check Report With Warning

Template

Check Report With Warning.rtf

Check Report By Validation Entity Seq.

Shows the results of the validation rules for the current location (indicates pass or fail status); sorted by the sequence defined in the validation entity group.

Runs for

Current FDMEE location

Parameters

None

Query

Check Report By Validation Entity

Template

Check Report By Validation Entity Sequence.rtf

Base Trial Balance Reports

The Trial Balance reports provide detail on how source data is processed in FDMEE. It is used to display account balances in the General ledger system. As data is loaded from the source General Ledger system to the target EPM application, you can validate and compare the balances loaded with the source Trial Balance amounts.

Note: Before running the base Trial Balance Reports, confirm that the user who runs the base Trial Balance reports has access to the location associated with the report. (See "Defining Location Security" on page 47).

TB Current Location, with Targets (Cat, Per)

Shows imported source accounts (departments) and their corresponding accounts (entities).

Runs for

Current FDMEE location

Parameters

Category, Period

Query

Current Trial Balance With Location with Targets

Template

TB Location With Targets.rtf

TB Current Location with Rules (Cat, Per)

Shows imported source accounts (departments) and the mapping entity rule (map wildcard) in which the accounts are included.

Runs for

Current FDMEE location

Parameters

Category, Period

Query

TB Location With Query

Template

TB Location with Rules.rtf

TB Current Locations, All Dimensions-Targets, by Target Entity-Account (Cat, Per)

Shows all imported records with all dimensions and their respective targets: grouped by target entity and account.

Runs for

Current FDMEE location

Parameters

Category, Period

Query

Trial Balance Current Location with Targets

Template

TB/(All Dimensions with Targets) by Target Entity Account.rtf

TB Current Locations, All Dimensions-Targets (Cat, Per)

Shows all imported records with all dimensions and their respective targets.

Runs for

Current FDMEE location

Parameters

Category, Period

Query

Trial Balance Location All Dimension.

Template

TB with Transaction Currency.rtf

TB Current Location, by Target Acct (Cat, Per)

Shows imported accounts: subtotaled by target accounts.

Runs for

Current FDMEE location

Parameters

Category, Period

Query

Trial Balance Current Location Sorted By Target Account

Template

TB With Target Account.rtf

TB Current Location, By Target Entity Account (Cat, Per)

Shows all imported records with all dimensions and their respective targets; grouped by target entity and account.

Runs for

Current FDMEE location

Parameters

Category, Period

Query

Trial Balance Base Transaction Currency

Template

Base Trial Balance (All Dimensions with Targets).rtf

TB Converted Current Location by Target Entity/Account

Shows imported accounts and entities in addition to original and converted accounts: subtotaled by target entity.

Runs for

Current FDMEE location

Parameters

Category, Period

Trial Balance Converted by Target Entity/Account Query

Template

TB Converted Current Location by Target Entity Account.rtf

Listing Reports

Listing reports summarize metadata and settings (such as the import format, or check rule) by the current location.

Import Formats by Location

Displays a list of all import formats; sorted by FDMEE location.

Runs for

N/A

Parameters

None

Query

Import Format By Location

Template

Import Format by Location.rtf

Location Listing

Shows a list of all mapping rules for a selected period, category, or dimension.

Runs for

Current FDMEE location

Parameters

Any FDMEE Dimension, Period, Category

Location Listing Query

Template

Location Listing.rtf

Location Analysis

Location Analysis reports provide dimension mapping by the current location.

Dimension Map (Dimension)

Displays a list of all mapping rules for a selected dimension.

Runs for

Current FDMEE location

Parameters

Current FDMEE dimension

Query

Dimension Map

Template

Dimension Map.rtf

Dimension Map For POV (Dimension, Cat, Per)

Shows a list of all mapping rules for a selected period, category, or dimension.

Runs for

Current FDMEE location

Parameters

Any FDMEE Dimension, Period, Category

Dimension Map for POV

Template

Dimension Map.rtf

Process Monitor Reports

The Process Monitor Reports shows locations and their positions within the data conversion process. You can use the process monitor report to monitor the status of the closing process. The report is time-stamped. Therefore, it can be used to determine to which locations at which time data was loaded.

Process Monitor (Cat, Per)

Shows all locations and their current status (import, validate, export, load, or check). (Locations are displayed alphabetically.)

Runs for

All FDMEE locations

Parameters

Category, Period

Query

Process Monitor

Template

Process Monitor.rtf

Process Monitor With Lock (Cat, Per)

Shows a list of all locations and their current status (import, validate, export, load, or check), including the lock status of the locations.

Runs for

All FDMEE locations

Parameters

Category, Period

Query

ProcessMonitorwithLock

Template

ProcessMonitorwithlock.rtf

Process Status Period Range (Cat, Start Per, End Per)

Shows a list of all locations and the current load status of each location for each period of a period range.

Runs for

All FDMEE locations, period range

Parameters

Category, Start Period, End Period

Query

PMPeriodRange

Template

PMPeriodRange.rtf

Process Monitor All Categories (Cat, Per)

Shows a list of all locations and the current load status of every location for every category.

Runs for

All FDMEE categories and locations

Parameters

Period

Process Monitor All Categories

Template

Process Monitor All Category.rtf

Variance Reports

The Variance reports display source and trial balance accounts for one target account, showing data over two periods or categories.

Account Chase Variance

Displays source input accounts for one target input account, showing variances over two periods or categories.

Runs for

All FDMEE locations

Parameters

Target Account, Category 1, Period 1, Category 2, Period 2.

Query

Account Chase Variance

Template

Account Chase Variance.rtf

Trial Balance Variance

Shows source input accounts, subtotaled by target accounts, showing variances over two periods or categories.

Runs for

Current FDMEE location

Parameters

Category 1, Period 1, Category 2, Period 2

Query

Trial Balance Variance

Template

TB Variance.rtf



FDMEE Web Service

In This Appendix

executeDataRule	345
executeMetaDataRule	346
getDataRuleNames	346
getLocationNames	347
getPeriodNames	347

This appendix describes the FDMEE web service.

Note: It is recommended that you use batch scripts rather than the web services, For more information about batch scripts, see "Working with Batch Scripts" on page 260.

Access the FDMEE Web Services WSDL at http://server:19000/oracle-epm-erpi-webservices/RuleService?wsdl, where Host Name is the name of the host where Workspace is deployed.

To use the Web Service, configure the Oracle Web Services Manager. For more details, refer to Configuring Oracle Web Services Manager for EPM System Products in the *Oracle Enterprise Performance Management System Deployment Options Guide*.

executeDataRule

The executeDataRule method executes a data rule after validating the parameters passed with the rule. If the parameter is invalid, error messages are displayed.

Parameter Name	Valid Values	Associated Error Codes	Comments
String pUserName String pPassword String pSSOToken		AIF_ERR_00002:Invalid User Name/ Password combination. AIF_ERR_00001:Invalid Token	Of these three parameters you should pass either the username and password or a valid sso_token.
String plmportFromSource	Y/N	AIF_ERR_00005:Invalid value for Import From Source Parameter	

Parameter Name	Valid Values	Associated Error Codes	Comments
String pExportToTarget	Y/N	AIF_ERR_00006:Invalid value for Export to Target Parameter	
String pRuleName		AIF_ERR_00004:Invalid Data Rule Name	The name of the data rule for a location.
String pExecutionMode	INCREMENTAL/SNAPSHOT	AIF_ERR_00007:Invalid value for Execution mode	Data rule execution mode
String pLoadExchangeRateFlag	Y/N	AIF_ERR_00008:Invalid value for Load Exchange Rate Flag	This parameter can be used only in case of multi-currency applications.
String pStartPeriodName	Start Period name	AIF_ERR_00010:Invalid Period Keys	
String pEndPeriodName	End Period name	AIF_ERR_00010:Invalid Period Keys	
String pSyncMode	TRUE/FALSE	AIF_ERR_00009:Invalid value for sync mode	This parameter represents where the execution should take place synchronously or asynchronously. Value of True executes the job synchronously.

executeMetaDataRule

The executeMetaDataRule method executes a metadata rule after validating the locationName parameter. If the parameter is invalid, error messages are displayed.

Parameter Name	Valid Values	Associated Error Codes	Comments
String pUserName String pPassword String pSSOToken		AIF_COMMON_ERR_00001:Invalid token. AIF_COMMON_ERR_00002:Invalid User Name/Password combination.	Of these three parameters you should pass either the username and password or a valid sso_token.
String locationName		AIF_ERR_00015: Invalid Location Name	The location name
String pSyncMode	TRUE/FALSE	AIF_ODI_ERR_00002:Invalid value passed for synchronous mode parameter.	This parameter represents where the execution should take place synchronously or asynchronously. Value of True executes the job synchronously.

getDataRuleNames

The getDataRuleNames method retrieves a list of data rule names for the location. If no location is passed, an empty array is returned.

Parameter Name	Valid Values	Associated Error Codes	Comments
String pUserName String pPassword String pSSOToken		AIF_COMMON_ERR_00002:Invalid User Name/Password combination. AIF_COMMON_ERR_00001:Invalid token.	Of these three parameters you should pass either the username and password or a valid sso_token.
String locationName			The location name.

getLocationNames

Provides a list of locations for the executeMetaDataRule web service.

Parameter Name	Valid Values	Associated Error Codes	Comments
String pUserName		AIF_COMMON_ERR_00002:Invalid User Name/Password combination.	
String pPassword		AIF_COMMON_ERR_00001:Invalid token.	
String pSSOToken			

getPeriodNames

Provides a list of period names for the pStartPeriodName and pEndperiodName parameters of the executeDataRule service.

Parameter Name	Valid Values	Associated Error Codes	Comments
String pUserName		AIF_COMMON_ERR_00002:Invalid User Name/Password combination.	
String pPassword		AIF_COMMON_ERR_00001:Invalid token.	
String pSSOToken			



Source System Tables Used by FDMEE

In This Appendix

Fusion and E-Business Suite Source System Tables	349
PeopleSoft Enterprise Financial Management Source System Tables	351
PeopleSoft Human Capital Management Source System Tables	352
Creating Synonyms for FBS Tables	353

This section lists the source system tables used by FDMEE. FDMEE reads all tables listed and writes to GL_BUDGET_INTERFACE and GL_TRACK_DELTA_BALANCES.

It also describes how to create synonyms for EBS tables.

Fusion and E-Business Suite Source System Tables

These Fusion and E-Business Suite source system tables are used by FDMEE. All tables require read privileges, unless noted otherwise.

Table/View Name	Schema	Object Type	Privilege	Comments
FND_FLEX_VALIDATION_ QUALIFIERS	APPLSYS	Table	Read only	
FND_FLEX_VALIDATION_TABLES	APPLSYS	Table	Read only	
FND_FLEX_VALUES	APPLSYS	View	Read only	
FND_FLEX_VALUES_TL	APPLSYS	View	Read only	
FND_FLEX_VALUE_CHILDREN_V	APPS	View	Read only	View based on FND_FLEX_VALUE_ NORM_HIERARCHY, FND_FLEX_ VALUE_SETS, and FND_FLEX_ VALUES_VL
FND_FLEX_VALUE_NORM_ HIERARCHY	APPLSYS	Table	Read only	
FND_FLEX_VALUE_SETS	APPLSYS	Table	Read only	
FND_FORM_FUNCTIONS	APPLSYS	Table	Read only	
FND_ID_FLEXS	APPLSYS	Table	Read only	

Table/View Name	Schema	Object Type	Privilege	Comments
FND_ID_FLEX_SEGMENTS	APPLSYS	Table	Read only	
FND_ID_FLEX_SEGMENTS_TL	APPLSYS	Table	Read only	
FND_ID_FLEX_STRUCTURES	APPLSYS	Table	Read only	
FND_ID_FLEX_STRUCTURES_TL	APPLSYS	Table	Read only	
FND_LANGUAGES	APPLSYS	Table	Read only	
FND_NEW_MESSAGES	APPLSYS	Table	Read only	
FND_NUMBER	APPLSYS	PL/SQL Package	Execute	
FND_RESPONSIBILITY	APPLSYS	Table	Read only	
FND_RESPONSIBILITY_TL	APPLSYS	Table	Read only	
FND_SEGMENT_ATTRIBUTE_ VALUES	APPLSYS	Table	Read only	
GL_BALANCES	GL	Table	Read only	
GL_BALANCES_DELTA	GL	Table	Read only	
GL_BUDGETS	GL	Table	Read only	
GL_BUDGET_ENTITIES	GL	Table	Read only	
GL_BUDGET_INTERFACE	GL	Table	Read and write privileges are required.	
GL_BUDGET_VERSIONS	GL	Table	Read only	
GL_CODE_COMBINATIONS	GL	Table	Read only	
GL_CODE_COMBINATIONS_KFV	APPS	View	Read only	View based on GL_CODE_ COMBINATIONS
GL_DAILY_BALANCES	GL	Table	Read only	
GL_DAILY_CONVERSION_TYPES	GL	Table	Read only	
GL_DAILY_CONVERSION_TYPES_V	APPS	View	Read only	View based on GL_DAILY_ CONVERSION_TYPES
GL_DAILY_RATES	GL	Table	Read only	
GL_ENCUMBRANCE_TYPES	GL	Table	Read only	
GL_INTERFACE	GL	Table	Read/Write	
GL_JE_CATEGORIES	GL	Table	Read only	
GL_JE_CATEGORIES_TL	GL	Table	Read only	

Table/View Name	Schema	Object Type	Privilege	Comments
GL_JE_SOURCES_TL	GL	Table	Read only	
GL_LEDGERS		Table	Read only	Table (R12 only)
GL_PERIODS	GL	Table	Read only	
GL_PERIOD_SETS	GL	Table	Read only	
GL_PERIOD_STATUSES	GL	Table	Read only	
GL_PERIOD_TYPES	GL	Table	Read only	
GL_PERIOD_TYPES_V	APPS	View	Read only	View based on GL_PERIOD_TYPES
GL_SETS_OF_BOOKS	GL/APPS	View	Read only	Table (11i) / View based on GL_ LEDGERS (R12)
GL_STAT_ACCOUNT_UOM	GL	Table	Read only	
GL_TRACK_DELTA_BALANCES	GL	Table	Read/Write	Read and write privileges are required.

PeopleSoft Enterprise Financial Management Source System Tables

These PeopleSoft Enterprise Financial Management source system tables are used by FDMEE. All tables require read privileges, unless noted otherwise.

Table / View Name	Table / View Name	Table / View Name	Table / View Name
PS_AFFILIATE_LANG	PS_CHARTFIELD1_TBL	PS_LEDGER	PSDBFLDLABL
PS_AFFILIATE_VW	PS_CHARTFIELD2_TBL	PS_LEDGER_BUDG	PSDBFLDLABLLANG
PS_AFFINTRA1_LANG	PS_CHARTFIELD3_TBL	PS_LOCATION_TBL	PSKEYDEFN
PS_AFFINTRA1_VW	PS_CLASS_CF_LANG	PS_NAMES	PSOPTIONS
PS_AFFINTRA2_LANG	PS_CLASS_CF_TBL	PS_OPER_UNIT_LANG	PSRECDEFN
PS_AFFINTRA2_VW	PS_DEPT_TBL	PS_OPER_UNIT_TBL	PSRECFIELD
PS_ALTACCT_LANG	PS_DEPT_TBL_LANG	PS_PAYGROUP_TBL	PSTREEDEFN
PS_ALTACCT_TBL	PS_EARNINGS_TBL	PS_PC_BU_GL_VW	PSTREEDEFNLANG
PS_BD_SCENARIO_TBL	PS_FS_CF_TEMPLATE	PS_PC_INT_TMPL_GL	PSTREELEAF
PS_BOOK_CODE_TBL	PS_FS_FLD_PROMPT	PS_POSITION_DATA	PSTREENODE
PS_BU_BOOK_TBL	PS_FUND_LANG	PS_PROD_TBL_LANG	PSTREESTRCT

Table / View Name	Table / View Name	Table / View Name	Table / View Name
PS_BU_LED_GRP_TBL	PS_FUND_TBL	PS_PRODUCT_TBL	PSXLATITEM
PS_BUD_REF_LANG	PS_GL_ACCOUNT_LANG	PS_PROGRAM_LANG	PSXLATITEMLANG
PS_BUD_REF_TBL	PS_GL_ACCOUNT_TBL	PS_PROGRAM_TBL	
PS_BUS_UNIT_LANG	PS_HPYPB_ACCT_LN (Read and write privileges are required.)	PS_PROJECT	
PS_BUS_UNIT_TBL_FS	PS_HYP_KK_BD_HDR(Write privileges are required. Used for PeopleSoft Commitment Control)	PS_PROJECT_TBL	
PS_BUS_UNIT_TBL_GL	PS_HYP_KK_BD_LN (Write privileges are required. Used for PeopleSoft Commitment Control)	PS_REC_GROUP_REC	
PS_BUS_UNIT_TBL_HR	PS_JOB	PS_REC_GROUP_TBL	
PS_CAL_ADJP_TBL	PS_JOBCODE_TBL	PS_RT_DFLT_VW	
PS_CAL_DEFN_TBL	PS_LED_DEFN_LANG	PS_RT_INDEX_TBL	
PS_CAL_DETP_TBL	PS_LED_DEFN_TBL	PS_RT_TYPE_TBL	
PS_CAL_SUMP_TBL	PS_LED_GRP_LANG	PS_SET_CNTRL_REC	
PS_CF1_LANG	PS_LED_GRP_LED_TBL	PS_SET_CNTRL_TBL	
PS_CF2_LANG	PS_LED_GRP_TBL	PS_SET_CNTRL_TREE	
PS_CF3_LANG	PS_LED_TMPLT_TBL	PS_SETID_TBL	

PeopleSoft Human Capital Management Source System Tables

These PeopleSoft Human Capital Management source system tables are used by FDMEE. All tables require read privileges.

Table / View Name	Table / View Name	Table / View Name	Table / View Name
PS_ACCT_CD_TBL	PS_CHARTFIELD3_TBL	PS_JOB_EARNS_DIST	PS_SAL_PLAN_TBL
PS_AFFILIATE_VW	PS_CLASS_CF_TBL	PS_JOBCODE_TBL	PS_SAL_STEP_TBL
PS_ALTACCT_TBL	PS_DEPT_BUDGET_ERN	PS_LOCATION_TBL	PS_SET_CNTRL_REC
PS_BENEF_PLAN_TBL	PS_DEPT_CF_TBL	PS_NAMES	PS_UNION_TBL
PS_BP_JOB	PS_DEPT_TBL	PS_OPER_UNIT_TBL	PSOPTIONS
PS_BUD_REF_TBL	PS_EARNINGS_TBL	PS_PAYGROUP_TBL	PSRECFIELD
PS_BUS_UNIT_TBL_FS	PS_FREQUENCY_TBL	PS_POSITION_DATA	PSXLATITEM

Table / View Name	Table / View Name	Table / View Name	Table / View Name
PS_BUS_UNIT_TBL_GL	PS_FUND_TBL	PS_PRODUCT_TBL	PSXLATITEMLANG
PS_BUS_UNIT_TBL_HR	PS_GL_ACCOUNT_TBL	PS_PROGRAM_TBL	
PS_CHARTFIELD1_TBL	PS_GL_ACCOUNT_VW	PS_PROJECT_TBL	
PS_CHARTFIELD2_TBL	PS_JOB	PS_SAL_GRADE_TBL	

Creating Synonyms for EBS Tables

If a user other than the "apps" user is referenced, you can create synonyms for EBS tables.

- ➤ To create synonyms for EBS Tables:
- 1 Create synonyms for the EBS tables listed in this appendix.
- 2 Assign read access to all the synonyms that you created.
- 3 Assign write access to the following tables:
 - GL_Interface
 - GL_Budget_Interface
 - GL_Track_Delta_Balances

In ODI Topology for EBS_Data_Server, you can use the synonym for both the user and the schema.



Creating an Alternate Schema in an ERP Source System

Prebuilt integrations to an ERP source system in FDMEE use a basic filter for data extraction, and assume that appropriate security has been defined to allow access by FDMEE. In some environments, direct access to the source systems tables is prohibited because of system policies, or you may want to define a source system filter that is more detailed than what is provided in the FDMEE user interface.

For example, in the Account Reconciliation Manager (ARM) you may want to extract balance sheet accounts or active accounts only from the source system. To do this, create an alternate schema in the source system. This method provides a desired level of security, a different source system filter, or both.

- To create an alternate hierarchy for E-Business Suite (EBS) and Peoplesoft (PSFT) systems:
- 1 Create a new schema or user in the source system database.
- 2 Grant SELECT or SELECT/INSERT access to the list of source tables used by FDMEE to the new schema.
 Refer to the source table list provided inAppendix B, "Source System Tables Used by FDMEE".
- For Oracle E-Business Suite systems, create a new view named **GL_CODE_COMBINATIONS**, which includes the desired source filter.
 - For **Peoplesoft** systems, create a view using the **PS_LEDGER** table.
 - All columns from the source table must be included in the view.
- 4 Create synonyms for all remaining source tables that FDMEE references from the source system in the new schema.
 - Synonyms point to the base tables in the source system schema.
- 5 Update **Oracle Data Integrator (ODI)** to use the new schema in the physical schema for the related data server.

For example, the view created on the EBS GL_CODE_COMBINATIONS table may look like:

```
CREATE VIEW GL_COMBINATIONS (SEGMENT1, SEGMENT2,....)
SELECT SEGMENT1, SEGMENT2,....
FROM APPS.GL_CODE_COMBINATIONS
WHERE "ADD FILTERS"
```

Any views created in the new schema or synonyms must use the exact same name as specified in Appendix B, "Source System Tables Used by FDMEE". Because the table and view names are

the same as the core schema, FDMEE can access the updated contents with a change to the schema specification in ODI.

For SAP or JDE source systems, change the filter definition in the adapter rather than creating an alternate schema.

Before making any changes, it is recommended that you contact Oracle support to review the process to ensure a smooth update to the system.



Staging Tables

FDMEE Staging Tables

Subtopics

- Staging Tables Used for Import from Source
- Data Tables Used for Drill Through
- Views Used for Export to Target

The following sections describe FDMEE staging tables.

Staging Tables Used for Import from Source

Table / View Name	Description
TDATASEG_T	Temporary data table used for transforming data from source to target in a data rule execution. To ensure read consistency, data is extracted from the corresponding staging table (for example, AIF_EBS_GL_BALANCES_STG, AIF_EBS_GL_DAILY_BAL_STG, or AIF_PS_LEDGER) and copied to TDATASEG_T (partitioned by LOADID), which is then used for transforming source data to target data based on the data mappings specified in TDATAMAP_T.
TDATASEGW	Temporary data table used for transforming write-back data from source to target in a write-back rule execution. Write-back data is extracted from the temporary staging table (AIF_WRITEBACK_ESS_DATA_T) and copied over to TDATASEGW (partitioned by LOADID), which is then used for transforming source data to target data based on the data mappings specified in TDATAMAP_T.
	If you have selected to only "Import from Source," you can query this table by LOADID to view the data transformations that were applied during the write-back rule execution.

Data Tables Used for Drill Through

Table / View Name	Description
TDATASEG	Audit table describing the data transformations in a data rule execution. Only in a successful data rule execution data transformations stored in TDATASEG_T be copied over to TDATASEG.

Views Used for Export to Target

Table / View Name	Description
AIF_HS_BALANCES	Interface view used for exporting data to a target application in a data rule execution. This view queries the TDATASEG_T table directly by summarizing the target amounts across the unique target dimension member combinations.
AIF_EBS_GL_INTERFACE_V	Interface view used for exporting data back to the Fusion and E-Business Suite general ledger system in a write-back rule execution. This view queries the TDATASEGW table directly by summarizing the target amounts across the unique target segment value combinations.
AIF_PS_GL_INTERFACE_V	Interface view used for exporting data back to the PeopleSoft Enterprise Financial Management system in a write-back rule execution. This view queries the TDATASEGW table directly by summarizing the target amounts across the unique target ChartField value combinations.

PeopleSoft Human Capital Management Staging Tables

Subtopics

- PS90HCM
- CHARTFIELD

The following sections describe the columns in each PeopleSoft Human Capital Management staging table used by FDMEE.

PS90HCM

 Table 58
 AIF_PS_POSITION_DATA_STG Table

Column Name	Column Description
DESCR	Position Description

Table 59 AIF_PS_JOBCODE_TBL_STG Table

Column Name	Column Description
COMP_FREQUENCY	Compensation Frequency

Column Name	Column Description
DESCR	Description
EFFDT	Effective Date
EFF_STATUS	Status
STD_HOURS	Standard Hours

Table 60 AIF_PS_JOB_STG Table

Column Name	Column Description
ABSENCE_SYSTEM_CD	Absence System
ACCDNT_CD_FRA	Work Accident Code
ACCT_CD	Account Code
ADDS_TO_FTE_ACTUAL	Adds to FTE Actual Count
ANNL_BENEF_BASE_RT	Annual Benefit Base Rate
APPT_TYPE	Appointment Type
ASGN_END_DT	Assignment End Date
ASGN_START_DT	Assignment Start Date
AUTO_END_FLG	Auto End Job
BARG_UNIT	Bargaining Unit
BAS_ACTION	Benefits Administration Action
BAS_GROUP_ID	Benefits Administration Group ID
BENEFIT_SYSTEM	Benefits System
BORDER_WALKER	Cross Border Worker
CHANGE_AMT	Change Amount
CHANGE_PCT	Change Percent
CLASS_INDC	Classified/Unclassified Indicator
COBRA_ACTION	COBRA Action
COMPANY	Company
COMPRATE	Compensation Rate
COMP_FREQUENCY	Compensation Frequency
CONTRACT_NUM	Contract Number

Column Name	Column Description
CTG_RATE	Category Rate
CURRENCY_CD1	Currency Code 1
CUR_RT_TYPE	Currency Rate type
DAILY_RT	Daily Rate
DEPT_ENTRY_DT	Department Entry Date
DIRECTLY_TIPPED	Directly Tipped
EARNS_DIST_TYPE	Earnings Distribution Type
EEO_CLASS	EEO Class
ELIG_CONFIG1	Eligibility Configuration Field 1
ELIG_CONFIG2	Eligibility Configuration Field 2
ELIG_CONFIG3	Eligibility Configuration Field 3
ELIG_CONFIG4	Eligibility Configuration Field 4
ELIG_CONFIG5	Eligibility Configuration Field 5
ELIG_CONFIG6	Eligibility Configuration Field 6
ELIG_CONFIG7	Eligibility Configuration Field 7
ELIG_CONFIG8	Eligibility Configuration Field 8
ELIG_CONFIG9	Eligibility Configuration Field 9
EMPL_CTG	Employee Category
EMPL_CTG_L1	Employee Subcategory
EMPL_CTG_L2	Employee Subcategory 2
ENCUMB_OVERRIDE	Encumbrance Override
ENTRY_DATE	Entry Date
ESTABID	Establishment ID
EXEMPT_HOURS_MONTH	Exempted Hours Month
EXEMPT_JOB_LBR	Exempted
EXPECTED_END_DATE	Expected Job End Date
EXPECTED_RETURN_DT	Expected Return Date
FLSA_STATUS	FLSA Status

Column Name	Column Description
FORCE_PUBLISH	Force Publish
FUNCTION_CD	Function Code
GL_PAY_TYPE	GL Pay type
GP_ASOF_DT_EXG_RT	Use Rate As Of
GP_DFLT_CURRTTYP	Use Pay Group Rate Type
GP_DFLT_ELIG_GRP	Use Pay Group Eligibility
GP_DFLT_EXRTDT	Use Pay Group As Of Date
GP_ELIG_GRP	Eligibility Group
GP_PAYGROUP	Pay Group
GRADE_ENTRY_DT	Salary Grade Entry Date
HIRE_DT	Hire Date
HOLIDAY_SCHEDULE	Holiday Schedule
HOURLY_RT	Hourly Rate
HOURLY_RT_FRA	Hourly Rate France
HR_STATUS	HR Status
INTERCTR_WRKS_CNCL	Works Council Role
	Role that the worker has in the union group
JOB_DATA_SRC_CD	Job Source Code
JOB_ENTRY_DT	Job Entry Date
JOB_INDICATOR	Job Indicator
LABOR_AGREEMENT	Labor Agreement
LABOR_FACILITY_ID	Labor Facility ID
LABOR_TYPE_GER	Labor Type Germany
LASTUPDDTTM	Last Update Date/Time
LASTUPDOPRID	Last Update User ID
LAST_DATE_WORKED	Last Date Worked
LAST_HIRE_DT	Last Hire Date
LAYOFF_EXEMPT_FLAG	Layoff Exempt Flag
LAYOFF_EXEMPT_RSN	Layoff Exempt Reason

Column Name	Column Description
LBR_FAC_ENTRY_DT	Labor Facility Entry Date
LDW_OVR	Override Last Date Worked
LST_ASGN_START_DT	Last Assignment Start Date
LUMP_SUM_PAY	Lump Sum Pay
MAIN_APPT_NUM_JPN	Main Appointment Number
MATRICULA_NBR	Matricula Number
MONTHLY_RT	Monthly Rate
OFFICER_CD	Officer Code
PAID_FTE	Paid FTE
PAID_HOURS	Paid Hours
PAID_HRS_FREQUENCY	Paid Hours Frequency
PAYGROUP	Pay Group
PAY_SYSTEM_FLG	Pay System Flag
PAY_UNION_FEE	Pay Union Fee
PERFORM_GROUP_GER	Perform Group Germany
PER_ORG	Organizational Relationship
POI_TYPE	Person of Interest Type
POSITION_ENTRY_DT	Position Entry Date
POSITION_OVERRIDE	Position Override
POSN_CHANGE_RECORD	Position Change Record
PRORATE_CNT_AMT	Prorate Contract Change Amount
REG_REGION	Regulatory Region
REPORTS_TO	Reports To
SETID_DEPT	SetID Department
SETID_JOBCODE	SetID Job Code
SETID_LBR_AGRMNT	SetID Labor Agreement
SETID_LOCATION	SetID Location
SETID_SALARY	SetID Salary

Column Name	Column Description
SETID_SUPV_LVL	SetID Supervisor Level
SHIFT	Shift
SHIFT_FACTOR	Shift Factor
SHIFT_RT	Shift Rate
SOC_SEC_RISK_CODE	Social Security Risk Code
SPK_COMM_ID_GER	Spokesman Committee ID
STD_HRS_FREQUENCY	Standard Hours Frequency
STEP_ENTRY_DT	Step Entry Date
SUPERVISOR_ID	Supervisor ID
SUPV_LVL_ID	Supervisor Level ID
TARIFF_AREA_GER	Tariff Area Germany
TARIFF_GER	Tariff Germany
TAX_LOCATION_CD	Tax Location Code
TERMINATION_DT	Termination Date
UNION_FEE_AMOUNT	Union Fee Amount
UNION_FEE_END_DT	Union Fee End Date
UNION_FEE_START_DT	Union Fee Start Date
UNION_FULL_PART	Union Participation
UNION_POS	Union Position
UNION_SENIORITY_DT	Union Seniority Date
VALUE_1_FRA	Value 1
VALUE_2_FRA	Value 2
VALUE_3_FRA	Value 3
VALUE_4_FRA	Value 4
VALUE_5_FRA	Value 5
WORK_DAY_HOURS	Work Day Hours
WPP_STOP_FLAG	Stop Wage Progression
WRKS_CNCL_FUNCTION	Work Council Function

Column Name	Column Description
WRKS_CNCL_ROLE_CHE	Work Council Role

 Table 61
 AIF_PS_POSITION_DATA_STG Table

Column Name	Column Description
ADDS_TO_FTE_ACTUAL	Adds to FTE Actual Count
BARG_UNIT	Bargaining Unit
CLASS_INDC	Classified/Unclassified Indicator
COMPANY	Company
CONFIDENTIAL_POSN	Confidential Position
COUNTRY_CODE	Country Code
DESCRSHORT	Position Short Description
EG_ACADEMIC_RANK	Academic Rank
EG_GROUP	Service Calculation Group
ENCUMBER_INDC	Encumbrance Indicator
ENCUMB_SAL_AMT	Encumbrance Salary Amount
ENCUMB_SAL_OPTN	Encumbrance Salary Option
FLSA_STATUS	FLSA Status
FRI_HRS	Friday Hours
GRADE_FROM	Grade From
GRADE_TO	Grade To
GVT_AUDITED_BY_ID	Position Audited By
GVT_COMP_AREA	Competitive Area
GVT_COMP_LEVEL	Competitive Level
GVT_DRUG_TEST_REQ	Drug Test (Applicable)
GVT_DT_POSN_ESTAB	Date Position Established
GVT_EXEC_FIN_DISCL	Executive Financial Disclosure
GVT_FUND_SOURCE	Fund Source
GVT_INTEL_POSN	Intelligence Position
GVT_LAST_AUDIT_DT	Position Audit Date

Column Name	Column Description
GVT_LEO_POSITION	LEO/Fire Position
GVT_MAINT_REVIEWCD	Maintenance Review
GVT_MANTREV_RESULT	Maintenance Review Results
GVT_MOB_POSN_IND	Mobility Position
GVT_NFC_FUNCTN_CD	NFC Function Code (NFC - National Finance Center)
GVT_NFC_PI_IND_OVR	Override NFC Indicator
GVT_NTE_DATE	Not to Exceed Date
GVT_OBLIGATED_TO	Obligated To ID
GVT_OBLIG_EXPIR_DT	Obligation Expiration
GVT_OCC_SERIES	Occupational Series
GVT_ORG_NUMCD1	Numeric Code 1
GVT_ORG_NUMCD2	Numeric Code 2
GVT_ORG_NUMCD3	Numeric Code 3
GVT_ORG_NUMCD4	Numeric Code 4
GVT_ORG_NUMCD5	Numeric Code 5
GVT_ORG_NUMCD6	Numeric Code 6
GVT_ORG_NUMCD7	Numeric Code 7
GVT_ORG_SUBAGENCY	Organizational Structure
GVT_ORG_TTL_CD	Organization Position Title Code
GVT_ORG_TTL_DESCR	Organization Position Title
GVT_PAREN_TITLE	Parenthetical Title
GVT_PAR_NBR	Personnel Action Request Number
GVT_PAY_PLAN	Pay Plan
GVT_PERF_PLAN	Performance Plan
GVT_PI_UPD_IND	PI Update Indicator
GVT_POI	Personnel Office ID
GVT_POSN_LOC	Position Location
GVT_POSN_OCCUPIED	Position Occupied

Column Name	Column Description
GVT_POSN_SENS_CD	Sensitivity Code
GVT_PRESIDENT_APPT	Presidential Appointment Position
GVT_PROC_INTEG_PSN	Procurement Integration Position
GVT_STAFF_LINE	Staff/Line Position
GVT_STATUS_TYPE	Status Type
GVT_SUB_AGENCY	Sub-Agency
GVT_TARGET_GRADE	Target Grade
GVT_VACY_REVW_CD	Vacancy Review Code
GVT_WIP_STATUS	Work-In-Progress Status
GVT_WORK_SCHED	Work Schedule
HEALTH_CERTIFICATE	Health Certificate
INCLUDE_SALPLN_FLG	Include Salary Plan Flag
JOB_SHARE	Job Share
KEY_POSITION	Key Position
LANGUAGE_SKILL	Language Skill
LASTUPDDTTM	Last Update Date/Time
LASTUPDOPRID	Last Update User ID
LOCATION	Location
MAIL_DROP	Mail Drop
MANAGER_LEVEL	Manager Level
MILITARY_SERVICE	Military Service
MIL_RANK	Military Rank
MIL_SVC_COMP_CD	Military Service Component
MON_HRS	Monday Hours
ORGCODE	Organization Hierarchy Code
ORGCODE_FLAG	Organization Hierarchy Flag
PHONE	Phone
PKG_RULE_ID	Package Rule ID

Column Name	Column Description
PKG_TEMPLATE_ID	Package Template ID
POSITION_POOL_ID	Position Pool ID
REG_REGION	Regulatory Region
REPORTS_TO	Reports To
REPORT_DOTTED_LINE	Report Dotted Line
SAT_HRS	Saturday Hours
SEASONAL	Seasonal
SEC_CLEARANCE_TYPE	Security Clearance Type
SHIFT	Shift
SIGN_AUTHORITY	Signing Authority
STATUS_DT	Status Date
STD_HRS_FREQUENCY	Standard Hours Frequency
STEP_FROM	Step From
STEP_TO	Step To
SUN_HRS	Sunday Hours
SUPV_LVL_ID	Supervisory Level ID
THURS_HRS	Thursday Hours
TRN_PROGRAM	Training Program
TUES_HRS	Tuesday Hours
UPDATE_INCUMBENTS	Update Incumbents
WED_HRS	Wednesday Hours

 Table 62
 AIF_PS_SAL_GRADE_TBL_STG Table

Column Name	Column Description
DESCR	Grade Description
DESCRSHORT	Grade Description Short
EFFDT	Effective Date
EFF_STATUS	Effective Status
ELIGIBILITY_POINTS	Eligibility Points

Column Name	Column Description
EMPL_CTG	Employee Category
EMPL_CTG_L1	Employee Subcategory
EMPL_CTG_L2	Employee Subcategory 2
GRADE	Grade
GRADE_TITLE_JPN	Grade Title Japan
GVT_MAX_RT_BIWK	Maximum Pay Rate - Bi-weekly
GVT_MID_RT_BIWK	Midpoint Pay Rate - Bi-weekly
GVT_MIN_RT_BIWK	Minimum Pay Rate - Bi-weekly
LABOR_AGREEMENT	Labor Agreement
MAX_YRS_GRADE	Max Years Grade
MIN_GRD_AGE	Minimum Grade Age
MIN_YRS_GRADE	Minimum Years Grade
REG_REGION	Regulatory Region
REVIEW_COUNT	Review Count
SALARY_MATRIX_CD	Salary Matrix Code
SAL_ADMIN_PLAN	Salary Administration Plan
SETID	SetID
STEP_INCREM_ACCUM	Step Increment Accumulated
STEP_INCREM_TYPE	Step Increment Type

 Table 63
 AIF_PS_SAL_STEP_TBL_STG Table

Column Name	Column Description
APS_MAX_CLS_AUS	Maximum Classification
APS_MIN_CLS_AUS	Minimum Classification
CLASSN_CURRENT_AUS	Job Classification
DAILY_RT	Daily Rate
EFFDT	Effective Date
GRADE	Grade
GVT_BIWEEKLY_RT	Bi-weekly Rate

Column Name	Column Description
GVT_INCREM_DAYS	Days to Next Step Increment
GVT_INCREM_WEEKS	Weeks to Next Step Increment
GVT_STEP	US Federal Step
GVT_WGI_ADJ	Within Grade Increase (WGI) Adjustment
INCREM_HOURS	Increment Hours
INCREM_MONTHS	Increment Months
SAL_ADMIN_PLAN	Salary Administration Plan
SETID	SetID
STEP_DESCR	Step Description

Table 64 AIF_PS_BP_JOB_STG Table

Column Name	Column Description
ACCDNT_CD_FRA	Account Code France
ACCT_CD	Account Code
ACTION	Action
ACTION_DT	Action Date
ACTION_REASON	Action Reason
ADDS_TO_FTE_ACTUAL	Adds to FTE Actual Count
ANNL_BENEF_BASE_RT	Annual Benefit Base Rate
ANNL_BEN_BASE_OVRD	Annual Benefit Base Overhead
ANNUAL_RT	Annual Rate
APPT_TYPE	Appointment Type
BARG_UNIT	Bargaining Unit
BAS_ACTION	Benefits Administration Action
BAS_GROUP_ID	Benefits Administration Group ID
BENEFIT_PROGRAM	Benefit Program
BENEFIT_SYSTEM	Benefit System
BEN_STATUS	Benefit Status
BORDER_WALKER	Cross Border Worker

Column Name	Column Description
BUSINESS_UNIT	Business Unit
CHANGE_AMT	Change Amount
CHANGE_PCT	Change Percent
CLASS_INDC	Classified/Unclassified Indicator
COBRA_ACTION	COBRA Action
COMPANY	Company
COMPRATE	Compensation Rate
COMP_FREQUENCY	Compensation Frequency
CONTRACT_NUM	Contract Number
CTG_RATE	Category Rate
CURRENCY_CD	Currency Code
CURRENCY_CD1	Currency Code 1
CUR_RT_TYPE	Currency Rate type
DAILY_RT	Daily Rate
DEPTID	Department
DEPT_ENTRY_DT	Department Entry Date
DIRECTLY_TIPPED	Directly Tipped
EARNS_DIST_TYPE	Earnings Distribution Type
EEO_CLASS	EEO_Class
EFFDT	Effective Date
EFFSEQ	Effective Sequence
ELIG_CONFIG1	Eligibility Configuration Field 1
ELIG_CONFIG2	Eligibility Configuration Field 2
ELIG_CONFIG3	Eligibility Configuration Field 3
ELIG_CONFIG4	Eligibility Configuration Field 4
ELIG_CONFIG5	Eligibility Configuration Field 5
ELIG_CONFIG6	Eligibility Configuration Field 6
ELIG_CONFIG7	Eligibility Configuration Field 7

Column Name	Column Description	
ELIG_CONFIG8	Eligibility Configuration Field 8	
ELIG_CONFIG9	Eligibility Configuration Field 9	
EMPLID	Employee ID	
EMPL_CLASS	Employee Classification	
EMPL_CTG	Employee Category	
EMPL_CTG_L1	Employee Subcategory	
EMPL_CTG_L2	Employee Subcategory 2	
EMPL_RCD	Employee Record Number	
EMPL_STATUS	Employee Status	
EMPL_TYPE	Employee Type	
ENCUMB_OVERRIDE	Encumbrance Override	
ENTRY_DATE	Entry Date	
ESTABID	Establishment ID	
EXEMPT_HOURS_MONTH	Exempted Hours Month	
EXEMPT_JOB_LBR	Exempted	
FICA_STATUS_EE	FICA Status	
FLSA_STATUS	FLSA Status	
FORCE_PUBLISH FTE	Force Publish FTE	
FULL_PART_TIME	Full or Part-Time	
FUNCTION_CD	Function Code	
GL_PAY_TYPE	GL Pay type	
GP_ASOF_DT_EXG_RT	Use Rate As Of	
GP_DFLT_CURRTTYP	Use Pay Group Rate Type	
GP_DFLT_ELIG_GRP	Use Pay Group Eligibility	
GP_DFLT_EXRTDT	Use Pay Group As Of Date	
GP_ELIG_GRP	Eligibility Group	
GP_PAYGROUP	Pay Group	
GRADE	Grade	

Column Name	Column Description
GRADE_ENTRY_DT	Grade Entry Date
GVT_ANNL_RT_NO_LOC	Annuity Rate
GVT_ANNUITY_OFFSET	Annuity Offset
GVT_ANNUIT_COM_DT	Annuity Date
GVT_ANN_IND	Annuity
GVT_BASIC_LIFE_RED	Basic Life Red
GVT_BIWEEKLY_RT	Biweekly Rate
GVT_BW_RT_NO_LOC	Biweekly Rate No Location
GVT_CHANGE_FLAG	Change Flag
GVT_COMPRATE	Compensate Rate
GVT_CONT_EMPLID	Contract Employee ID
GVT_CSRS_FROZN_SVC	CSRS Frozen Service
GVT_DAILY_RT	Daily Rate
GVT_DED_PRORT_DT	DED Prorate Date
GVT_DETL_POSN_NBR	Position Number
GVT_DLY_RT_NO_LOC	Daily Rate No Location
GVT_EFFDT	Effective Date
GVT_EFFDT_PROPOSED	Effective Date Proposed
GVT_ELIG_FEHB	Eligibility FEHB
GVT_FEGLI	FEGLI Code
GVT_FEGLI_BASC_PCT	FEGLI Basic Percent
GVT_FEGLI_LIVING	FEGLI Living Benefits
GVT_FEGLI_OPT_PCT	FEGLI Opted Percent
GVT_FEHB_DT	FEHB Date
GVT_FEHB_PCT	FEHB Percent
GVT_FERS_COVERAGE	FERS Coverage
GVT_HRLY_RT_NO_LOC	Hourly Rate No Location
GVT_LEG_AUTH_1	Legal Authority 1

Column Name	Column Description
GVT_LEG_AUTH_2	Legal Authority 2
	,
GVT_LEO_POSITION	LEO/Fire Position (LEO - Law enforcement officer)
GVT_LIVING_AMT	Living Amount
GVT_LOCALITY_ADJ	Locality Adjustment
GVT_MNLY_RT_NO_LOC	Monthly Rate No Location
GVT_NID_CHANGE	NID Change
GVT_NOA_CODE	NOA Code
GVT_OTH_PAY_CHG	Other Pay Change
GVT_PAR_AUTH_D1	Authority 1 Description - Part 1
GVT_PAR_AUTH_D1_2	Authority 1 Description - Part 2
GVT_PAR_AUTH_D2	Authority 2 Description - Part 1
GVT_PAR_AUTH_D2_2	Authority 2 Description - Part 1
GVT_PAR_NTE_DATE	NTE Date
GVT_PAY_BASIS	Pay Basis
GVT_PAY_FLAG	Pay Flag
GVT_PAY_PLAN	Pay Plan
GVT_PAY_RATE_DETER	Pay Rate Determinant
GVT_PI_UPD_IND	PI Update Indicator
GVT_POI	Personnel Office ID
GVT_POSN_OCCUPIED	Position Occupied
GVT_PREV_RET_COVRG	Previous Retirement Coverage
GVT_RETIRE_PLAN	Retire Plan
GVT_RETRO_BSE_FLAG	Retrospective Base Flag
GVT_RETRO_DED_FLAG	Retrospective Deduction Flag
GVT_RETRO_FLAG	Retrospective Flag
GVT_RETRO_JOB_FLAG	Retrospective Job Flag
GVT_ROUTE_NEXT	Route to Next
GVT_RTND_GRADE	Retained Grade

Column Name	Column Description	
GVT_RTND_GVT_STEP	Retained Government Step	
GVT_RTND_PAY_PLAN	Retained Pay Plan	
GVT_RTND_SAL_PLAN	Retained Salary Plan	
GVT_RTND_STEP	Retained Step	
GVT_S113G_CEILING	S113G Ceiling	
GVT_SF52_NBR	SF52 Number	
GVT_STATUS_TYPE	Status Type	
GVT_STEP	Step	
GVT_SUB_AGENCY	Sub Agency	
GVT_TRANS_NBR	Transaction Number	
GVT_TRANS_NBR_SEQ	Transaction Number Sequence	
GVT_TSP_UPD_IND	TSP Status	
GVT_TYPE_OF_APPT	Type of Appointment	
GVT_WIP_STATUS	WIP Status	
GVT_WORK_SCHED	Work Schedule	
GVT_XFER_FROM_AGCY	Transfer From Agency	
GVT_XFER_TO_AGCY	Transfer To Agency	
HOLIDAY_SCHEDULE	Holiday Schedule	
HOURLY_RT	Hourly Rate	
HOURLY_RT_FRA	Hourly Rate France	
INTERCTR_WRKS_CNCL	Works Council Role	
JOBCODE	Job Code	
JOB_DATA_SRC_CD	Job Source Code	
JOB_ENTRY_DT	Job Entry Date	
JOB_INDICATOR	Job Indicator	
LABOR_AGREEMENT	Labor Agreement	
LABOR_FACILITY_ID	Labor Facility ID	
LABOR_TYPE_GER	Labor Type Germany	

Column Name	Column Description
LAYOFF_EXEMPT_FLAG	Layoff Exempt Flag
LAYOFF_EXEMPT_RSN	Layoff Exempt Reason
LBR_FAC_ENTRY_DT	Labor Facility Entry Date
LOCATION	Location
LUMP_SUM_PAY	Lump Sum Pay
MAIN_APPT_NUM_JPN	Main Appointment Number
MATRICULA_NBR	Matricula Number
MONTHLY_RT	Monthly Rate
OFFICER_CD	Officer Code
PAID_FTE	Paid FTE
PAID_HOURS	Paid Hours
PAID_HRS_FREQUENCY	Paid Hours Frequency
PAYGROUP	Pay Group
PAY_SYSTEM_FLG	Pay System Flag
PAY_UNION_FEE	Pay Union Fee
PERFORM_GROUP_GER	Perform Group Germany
POSITION_ENTRY_DT	Position Entry Date
POSITION_NBR	Position Number
POSITION_OVERRIDE	Position Override
POSN_CHANGE_ RECORD	Position Change Record
PRORATE_CNT_AMT	Prorate Contract Change Amount
REG_REGION	Regulatory Region
REG_TEMP	Regular or Temporary
REPORTS_TO	Reports To
SAL_ADMIN_PLAN	Salary Administration Plan
SETID_DEPT	SetID Department
SETID_JOBCODE	SetID Job Code
SETID_LBR_AGRMNT	SetID Labor Agreement

Column Name	Column Description
SETID_LOCATION	SetID Location
SETID_SALARY	SetID Salary
SHIFT	Shift
SHIFT_FACTOR	Shift Factor
SHIFT_RT	Shift Rate
SOC_SEC_RISK_CODE	Social Security Risk Code
SPK_COMM_ID_GER	Spokesman Committee ID
STD_HOURS	Standard Hours
STD_HRS_FREQUENCY	Standard Hours
STEP	Frequency Step
STEP_ENTRY_DT	Step Entry Date
SUPERVISOR_ID	Supervisor ID
TARIFF_AREA_GER	Tariff Area Germany
TARIFF_GER	Tariff Germany
TAX_LOCATION_CD	Tax Location Code
UNION_CD	Union Code
UNION_FEE_AMOUNT	Union Fee Amount
UNION_FEE_END_DT	Union Fee End Date
UNION_FEE_START_DT	Union Fee Start Date
UNION_FULL_PART	Union Participation
UNION_POS	Union Position
UNION_SENIORITY_DT	Union Seniority Date
UPDATE_PAYROLL	Update Payroll
VALUE_1_FRA	Value 1
VALUE_2_FRA	Value 2
VALUE_3_FRA	Value 3
VALUE_4_FRA	Value 4
VALUE_5_FRA	Value 5

Column Name	Column Description
WORK_DAY_HOURS	Work Day Hours
WPP_STOP_FLAG	Stop Wage Progression
WRKS_CNCL_FUNCTION	Work Council Function
WRKS_CNCL_ROLE_CHE	Work Council Role

CHARTFIELD

 Table 65
 Table Type—CHARTFIELD

Table Name	Column Name	Column Description
AIF_PS_GL_ACCOUNT_TBL_STG	ACCOUNT	Account
AIF_PS_AFFILIATE_VW_STG	AFFILIATE	Affiliate
AIF_PS_ALTACCT_TBL_STG	ALTACCT	Alternate Account
AIF_PS_BUD_REF_TBL_STG	BUDGET_REF	Budget Ref
AIF_PS_CHARTFIELD1_TBL_STG	CHARTFIELD1	Chartfield 1
AIF_PS_CHARTFIELD2_TBL_STG	CHARTFIELD2	Chartfield 2
AIF_PS_CHARTFIELD3_TBL_STG	CHARTFIELD3	Chartfield 3
AIF_PS_CLASS_CF_TBL_STG	CLASS_FLD	Class
AIF_PS_DEPT_CF_TBL_STG	DEPTID_CF	Department
AIF_PS_FUND_TBL_STG	FUND_CODE	Fund Code
AIF_PS_OPER_UNIT_TBL_STG	OPERATING_UNIT	Operating Unit
AIF_PS_PRODUCT_TBL_STG	PRODUCT	Product
AIF_PS_PROGRAM_TBL_STG	PROGRAM_CODE	Program
AIF_PS_PROJECT_TBL_STG	PROJECT	Project



Working with Open Interface Adapters

In This Appendix

Setting up the Open Interface Adapter......379

Setting up the Open Interface Adapter

The Open Interface Adapter is used to map data from the open interface table (AIF_OPEN_INTERFACE) to the FDMEE staging tables. Typically, the Open Interface Adapter is used to load from a source not supported with a pre-packaged adapter. The Open Interface Adapter eliminates the need to process and manage data files, but the user is responsible for loading the table. Load methods include:

- Event script in FDMEE
- ODI
- SQL Loader
- anything

You configure this adapter using FDMEE to populate the staging table with the related data from the open interface table. You load the open interface table with the tool of your choice like a pl/sql routine, a sql loader script or a custom program. When you use ODI to load the open interface table, you can customize the Open Interface Adapter ODI package shipped with FDMEE to call a custom ODI interface to load the open interface table as a step in the data load process.

Using the Open Interface Table

The open interface table is called AIF_OPEN_INTERFACE. The table definition is as follows:

Table 66 AIF_OPEN_INTERFACE Table Definitions

Column Name	Data Type	Comment
BATCH_NAME	VARCHAR2(200 BYTE)	Required. Used to filter the data
YEAR	NUMBER(15,0)	Optional. Required when period mapping is Explicit
PERIOD	VARCHAR2(30 BYTE)	Optional. Required when period mapping is Explicit
PERIOD_NUM	NUMBER(15,0)	Optional. Required when period mapping is Explicit

Column Name	Data Type	Comment
CURRENCY	VARCHAR2(10 BYTE)	Required.
DATAVIEW	VARCHAR2(8 BYTE)	Used for HFM applications. Assign value of YTD or Periodic. Default value is YTD
DESC1	VARCHAR2(75 BYTE)	Optional. Mapped to DESC1
DESC2	VARCHAR2(75 BYTE)	Optional. Mapped to DESC2
AMOUNT	NUMBER(29,12)	Required. Mapped to AMOUNT
AMOUNT_YTD	NUMBER(29,12)	Optional. Mapped to AMOUNT_YTD
AMOUNT_PTD	NUMBER(29,12)	Optional. Mapped to AMOUNT_PTD
COL01	VARCHAR2(75 BYTE)	Input Source Column 1
COL02	VARCHAR2(75 BYTE)	Input Source Column 2
COL03	VARCHAR2(75 BYTE)	Input Source Column 3
COL30	VARCHAR2(75 BYTE)	Input Source Column 30
ATTR1	VARCHAR2(20 BYTE)	Optional. Mapped to ATTR1
ATTR2	VARCHAR2(20 BYTE)	Optional. Mapped to ATTR2
ATTR13	VARCHAR2(20 BYTE)	Optional Mapped to ATTR13
ATTR14	VARCHAR2(20 BYTE)	Reserved. Do not map.

Loading Source Data using Custom Loaders

- To load data using the Open Interface Table and Adapter:
- 1 Populate the Open Interface table as need.
- 2 Create any new period mappings as needed.
- 3 Create a new source system:
 - a. From the Task pane, select Setup, then Source System.
 - b. Select **Add** to add a new source system.
 - c. Specify the Name, Source System Type, and ODI Context Code.

The ODI context code refers to the context defined in Oracle Data Integrator. A context groups the source and target connection information.

For the Source system Type, select "Others," and for the ODI Context Code, enter the default "GLOBAL." If the ODI has been set up differently from the defaults, enter the appropriate ODI Context Code when it is not "GLOBAL."

- 4 From **Source Adapter**, copy the Open Interface Adapter to a new adapter.
 - a. Provide a new adapter key.

The adapter key can be ten characters long. Once the adapter key has been created, it cannot be modified.

b. Change the "Adapter Name" so that it is different from the copied adater name.

The adapter name is used in pick lists, and needs to be changed so that the copied adapter can be distinguished from the original adapter.

- 5 On the Import Format screen, define a new import format.
 - a. Select the source system created in Step 3 and the adapter from Step 4.
 - b. Map the source columns to the appropriate target dimensions.
- 6 When the import format mappings are complete, select Regenerate ODI Scenario.

This step generates the ODI Scenario based on the import format mappings.

- 7 On the **Locations** screen, define a new location using the import format.
- 8 On the Data Rule screen, define the new data rule.
 - a. Select the appropriate period mapping type.

Provide a value for the batch name. The batch name is used by the system as a filter on the open interface table. This allows multiple users to load data into a single interface table but then segregate the data by the batch name.

- b. Enter Y or N for the Delete Data After Import option.
- 9 Execute the data rule using one of three methods:
 - Data Rule Execution
 - Workbench
 - Batch Execution Script

Customizing ODI to Load Data from a Custom Source

You can modify the Open Interface Adapter to meet your business needs, and then once fully tested, move the updated content to a different environment for further testing or use in production. The steps to perform the customization and migration are listed below.

Use ODI Studio to perform the following steps. Before performing these steps, backup you work repository, custom project, and model.

When exporting and importing objects, use the following options:

- When exporting, select the **Child Components Export** option.
- When importing, use **Synonym Mode INSERT_UPDATE** as the Import Type.

- To modify the Open Interface Adapter:
- 1 From the FDMEE repository, export the Open Interface Adapter Model and ERPI Target Model folders.
- 2 From the FDMEE repository, export the Open Interface Adapter Project.
- 3 Create a new work repository with a new ID.
 - Use a repository ID of 900 or higher to avoid conflicts with Oracle shipped components. This is your development repository when performing customization. It is referred to as the "custom repository" in the following steps.
- In the custom repository, import the model folders named: Open Interface Adapter Model and ERPI Target Model.
- 5 In the custom repository, import the Open Interface Adapter Project.
- 6 Perform any customizations in the custom repository.
 - When customizing objects, do not modify FDMEE seeded objects imported from the FDMEE repository. Instead, copy needed objects, use the "Custom" suffix in names to identify them, and then modify them.
 - For example, copy the "Open Interface Balances" package as "Open Interface Balances Custom," and make any edits to it.
- 7 Export customized objects (for example, Project, Packages, Interfaces) from the custom repository and import them into the FDMEE repository.
- 8 Using the FDMEE Source Adapter user interface, copy the Open Interface Adapter to a new adapter and complete the following:
 - For the new adapter, provide a new adapter key. The adapter key can be ten characters long. Once it has been created, it cannot be modified.
 - Make any necessary modifications to the Display Names. Display Name are shown in the Import Format and Data Load Workbench.
 - Specify the ODI Package Name and ODI Project Code as specified in the custom project.

Migrating the Customization

- To migrate a customization from a test or production repository:
- 1 Export the model and project from the custom repository.
- Import the model and project into the target FDMEE repository (using the Synonym Mode INSERT_UPDATE as the Import Type).



PeopleSoft Commitment Control

PeopleSoft's Commitment Control is a budgetary control feature in the PeopleSoft General Ledger product, which supports the posting of budgets and tests transactions against budgetary balances according to the rules configured by users. Using Commitment Control, you can define an alternate structure from the General Ledger to manage budgets based on the Chartfield (account segments) and calendars. For example, you can choose to control budgets at a Division level by Quarter whereas actual expenses are recorded at the Cost Center level by month.

Additionally, you can make large scale changes to budgets, and write revisions as journals back to source systems for posting while maintaining distinctions in PeopleSoft budgets between original proposals, adjustments, revisions, and adjusted budgets. Hyperion Planning, for example, can be used to prepare the initial budget at the start of the fiscal year. It can also be used to make revisions to the budgets throughout the fiscal year. As budgets are prepared and revised, they have to be validated and posted to Commitment Control, which manages and controls the actual revenue and expenses.

FDMEE integrates with Commitment Control by allowing you to use Hyperion Planning to prepare and maintain the budgets. The integration involves:

- loading the Actual from Commitment Control
- validating budgets during preparation and revision against Commitment Control budget definitions
- posting the initial budget to Commitment Control
- posting ongoing budget revisions to Commitment Control

To use Commitment Control within the context of FDMEE, complete the following:

- 1. In **Source System**, register your General Ledger and HRMS source systems.
- 2. In **Source System**, to use PeopleSoft Commitment Control, select **Enable Commitment Control**.
- 3. In **Target Application**, register the target application.
- 4. In **Import Format**, define an import format that specifies how to map PeopleSoft chart fields to dimensions in the Public Sector Planning and Budgeting applications.
- 5. In **Data Load Rule**, define a location that identifies the PeopleSoft accounting entity (business unit) from which to load data.
- 6. In **Data Load Rule**, specify a period mapping type of "Budget."
- 7. In **Data Load Rule** define global, application, and source mappings that specify how period keys in the PeopleSoft calendar and time periods correspond to periods in your Public Sector

Planning and Budgeting budget application such as months, quarters, and years. Options include:

- select an "As of Date"
 - The "As of Date" to determine effective dated records, for example, Budget Definition.
- select a "Target for Blank Period" when the budget is based on a project period.
- Optionally, in **Period Mappings**, map a budget period by mapping a calendar and period to an FDMEE period.

Commitment Control allows different calendars to be used for different rules. The calendar can be of different granularity and duration. For example, you can map the Chartfield used for rule ranges to the Hyperion Planning Entity dimension. When you specify an explicit period mapping, you can map a Budget period by pointing to a calendar and period to an FDMEE period. The same FDMEE period can be mapped to multiple source periods from different calendars.

- 8. Run the data load rules.
- 9. In **Import Format** define an import format that contains write back mappings that identify the Public Sector Planning and Budgeting data to write to the PeopleSoft accounting entity chart fields.
- 10. In Write-Back Rule, define a write-back rule.

Write-back period mapping is based on explicit period maps. Every Entity is associated with a different Budget Calendar.

- 11. In **Write-Back Rule**, define write back filters that identify the portions of budget data to load to PeopleSoft.
- 12. Run the write back rule.

G

Integrating FDM

In This Appendix

Enabling FDM Applications	385
Drilling Through to the FDM Landing Page	386

The FDM features are available only when a customer upgrades from 11.1.1.4 or 11.1.2.X.

Enabling FDM Applications

You can enable applications registered in FDM to be registered as target applications in FDMEE. Use this for Oracle Hyperion Strategic Finance or Oracle Hyperion Enterprise® are registered in FDM, but are not registered in Oracle Hyperion Shared Services. After you enable the applications, define import formats, locations, and data load rules in FDMEE.

Note: FDMEE supports loading data and drill through for FDM applications, but does not support loading metadata or writing-back data for applications registered in FDM.

- To enable FDM applications:
- 1 From the **Setup** tab, and then under **Register**, select **FDM Application**.

This feature is for upgrade customers, who integrated from ERPI to FDM.

- 2 Click Add.
- 3 In the Name field, enter the FDM application schema name.
- 4 Enter the User Name and Password for the application schema.
- 5 Optional: Select Remote Database.

If FDM accesses a database that is different from the FDMEE database, select *Remote*.

If FDM accesses the *same* database as FDMEE, do not select Remote Database since FDMEE obtains the connection information from the Oracle Hyperion Shared Services Registry.

Note: FDM and FDMEE must be on the same database platform. Either on Oracle or SQL Server.

6 For SQL Server databases only: Enter the SQL Server Database Name.

- If you selected Remote Database, in JDBC URL, select a generic JDBC URL address. Then, modify the <host>, <port>, and <SID>. For example, server.oracle.com:1521:orcl.
- 8 To test the connection, click Test Connection.
- 9 Click Save.
- 10 Define the import format. See "Working with Import Formats" on page 83.

Drilling Through to the FDM Landing Page

For information on drill through capabilities to the Oracle Hyperion Financial Data Quality Management landing page, see the. *Oracle Hyperion Financial Data Quality Management, Administrator's Guide.*



Report Query Table Reference

In This Appendix

TDATASEG Table Reference	387
TLOGPROCESS Table Reference	391

When creating reports, you can use a base SQL query and a parameter SQL query to enhance the data shown in the report. The base SQL query can be used to pull data from various tables and display the data as report output. For example, using one query definition, you can show different columns or groupings. In one report, you can list amounts by account and group by entity, and in another list amount by entity and group by account.

The parameter SQL query enables you to run a query against the parameters in the report definition. For example, the query can be used to select the Location, Period, Category, or Account groupings. In one report, you can list amounts by account and group by entity, and in another list amount by entity and group by account.

The FDMEE tables that are used in the base and parameter SQL query are:

- TDATASEG
- TLOGPROCESS

TDATASEG Table Reference

The TDATASEG table is used to store the data loaded by the user, and the transformation between the source dimension members and results of the mapping process.

Table 67 TDATASEG Table Reference

Column Name	Definition	Description
DATAKEY	NUMBER(31,0) NOT NULL ENABLE	System generated unique key for each row of data
PARTIONNKEY	NUMBER(10,0) NOT NULL ENABLE	Location key. Join to TPOVPARTITION to retrieve location information.
CATKEY	NUMBER(10,0) NOT NULL ENABLE	Category Key. Join to TPOVCATEGORY to retrieve category information.
PERIODKEY	DATE NOT NULL ENABLE	Period Key. Join to TPOVPERIOD to retrieve FDMEE to EPM period mapping details.

Column Name	Definition	Description
DATAVIEW	VARCHAR2(8 CHAR) DEFAULT 'YTD' NOT NULL ENABLE	Hard coded to YTD for file, and set to YTD for balance sheet and PTD for income statement when pulling data from ERP.
CURKEY	VARCHAR2(10 CHAR) DEFAULT	Currency code of the data.
CALCACCTTYPE	NUMBER(6,0) DEFAULT 9 NOT NULL ENABLE	Indicates when row was imported from source or computed by Logic Group: • 9=Imported • 5=Calculated and Exported • 1=Calculated, and Not Exported
CHANGESIGN	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Indicates that the sign of the imported amount should be reversed: • 0=No Change • 1=Reverse Sign
JOURNALID	VARCHAR2(10 CHAR) DEFAULT	ID for the Journal. User provided value
AMOUNT	NUMBER(29,12) DEFAULT 0 NOT NULL ENABLE	Amount loaded from source
AMOUNTX	NUMBER(29 12) DEFAULT 0 NOT NULL ENABLE	Amount after any transformation rules. This value is loaded to the target application.
DESC1	VARCHAR2(75 CHAR) DEFAULT	Description can be imported from file
DESC2	VARCHAR2(75 CHAR) DEFAULT	Description can be imported from file
ACCOUNT	VARCHAR2(75 CHAR) NOT NULL ENABLE	Account member from source
ACCOUNTX	VARCHAR2(4000 CHAR) DEFAULT	Account member after mapping rules processed
ACCOUNTR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to Mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG.
ACCOUNTF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map types: • 1=Exception • 3=Between • 4=Range
ENTITY	VARCHAR2(75 CHAR) DEFAULT	Entity member from source
ENTITYX	VARCHAR2(75 CHAR) DEFAULT	Entity member after mapping rules processed. This is value that will be exported.
ENTITYR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to Mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG
ENTITYF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map types: • 1=Exception • 3=Between • 4=Range

Column Name	Definition	Description
ICP	VARCHAR2(75 CHAR) DEFAULT	ICP from source
ICPX	VARCHAR2(75 CHAR) DEFAULT	ICP after mapping rules processed. This value is exported.
ICPR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG.
ICPF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map type: • 1=Exception • 3=Between • 4=Range
UD1	VARCHAR2(75 CHAR) DEFAULT	UD1 from source
UD2	VARCHAR2(75 CHAR) DEFAULT	UD2 from source
UD3	VARCHAR2(75 CHAR) DEFAULT	UD3 from source
UD4	VARCHAR2(75 CHAR) DEFAULT	UD4 from source
UD5	VARCHAR2(75 CHAR) DEFAULT	UD5 from source
UD6	VARCHAR2(75 CHAR) DEFAULT	UD6 from source
UD7	VARCHAR2(75 CHAR) DEFAULT	UD7 from source
UD8	VARCHAR2(75 CHAR) DEFAULT	UD8 from source
UD9	VARCHAR2(75 CHAR) DEFAULT	UD9 from source
UD10	VARCHAR2(75 CHAR) DEFAULT	UD10 from source
UD11	VARCHAR2(75 CHAR) DEFAULT	UD11 from source
UD12	VARCHAR2(75 CHAR) DEFAULT	UD12 from source
UD13	VARCHAR2(75 CHAR) DEFAULT	UD13 from source
UD14	VARCHAR2(75 CHAR) DEFAULT	UD14 from source
UD15	VARCHAR2(75 CHAR) DEFAULT	UD15 from source
UD16	VARCHAR2(75 CHAR) DEFAULT	UD16 from source
UD17	VARCHAR2(75 CHAR) DEFAULT	UD17 from source
UD18	VARCHAR2(75 CHAR) DEFAULT	UD18 from source
UD19	VARCHAR2(75 CHAR) DEFAULT	UD19 from source
UD20	VARCHAR2(75 CHAR) DEFAULT	UD20 from source
UD1X	VARCHAR2(75 CHAR) DEFAULT	UD1 after mapping rules processed. This value is exported.

Column Name	Definition	Description
UD1R	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to Mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG
UD1F	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map type: • 1=Exception • 3=Between • 4=Range
ARCHIVEID	NUMBER(31,0) DEFAULT 0 NOT NULL ENABLE	Future use
HASMEMOITEM	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Future use
STATICDATAKEY	NUMBER(31,0) DEFAULT 0 NOT NULL ENABLE	Future use
ATTR1	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR2	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR3	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR4	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR5	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR6	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR7	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR8	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR9	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
ATTR10	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR11	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR12	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR13	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through

Column Name	Definition	Description
ATTR14	VARCHAR2(20 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill- through
CODE_COMBINATION_ID	VARCHAR2(155 CHAR)	Used for integration with EBS
AMOUNT_YTD	NUMBER(29,12)	YTD Amount. Used for EBS, Peoplesoft, Fusion data sources
AMOUNT_PTD	NUMBER(29,12)	PTD Amount. Used for EBS, Peoplesoft, Fusion data sources
LOADID	NUMBER(15,0)	Process ID that created or updated this row.
RULE_ID	NUMBER(15,0)	Data Rule ID used to create this row. Join to AIF_BALANCE_RULES for details.
STAT_BALANCE_FLAG	VARCHAR2(1 CHAR)	Indicates when balance is a statistic: • Y=Stat • N=Balance
VALID FLAC	VARCUARO(4 CUAR)	Indicates when row has valid mappings: • Y=Valid • N=Not Valid
VALID_FLAG	VARCHAR2(1 CHAR)	I=Ignore

TLOGPROCESS Table Reference

The TLOGPROCESS table is used to store the workflow process status for a location, category, and period.

 Table 68
 TLOGPROCESS Table Reference

Column Name	Definition	Description
PARTITIONKEY	NUMBER(10,0) NOT NULL ENABLE	Location key. Join to TPOVPARTITION to retrieve location information.
CATKEY	NUMBER(10,0) NOT NULL ENABLE	Category Key. Join to TPOVCATEGORY to retrieve category information.
PERIODKEY	DATE NOT NULL ENABLE	Period Key. Join to TPOVPERIOD to retrieve FDMEE to EPM period mapping details.
RULE_ID	NUMBER(15,0) NOT NULL ENABLE	Data Rule ID. Join to AIF_BALANCE_RULES for details.
PROCESSIMP	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Import step: O=Not started or failed 1=Successful

Column Name	Definition	Description
PROCESSIMPNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Validate status: Import Successful Recalculated OK Import Error Recalculate Failed MultiLoad BypassDataLoad Logic Calc Err Map Calc Err
PROCESSVAL	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE,	Status for Validate step: O=Not started or failed 1=Successful
PROCESSVALNOTE	VARCHAR2(50 CHAR) DEFAULT NULL,	Textual note on Validate step: Validate Successful Error= x records (Where X = how many members did not have map rules) BypassDataLoad
PROCESSEXP	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE,	Status for Export step: O=Not started or failed 1=Successful
PROCESSEXPNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Export step: Last successful export Export -B Successful Export Successful BypassDataLoad
PROCESSENTLOAD	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Load step: O=Not started or failed 1=Successful
PROCESSENTLOADNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Load status: Time Date stamp for success Load Error BypassDataLoad
PROCESSENTVAL	NUMBER(1,0) DEFAULT O NOT NULL ENABLE,	Status for Check step: O=Not started or failed 1=Successful

Column Name	Definition	Description
PROCESSENTVALNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Check step: Check Successful Check Error BypassDataLoad
PROCESSCERT	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE,	Status for Certification step: • 0=Not started or unsubmitted • 1=Submitted
PROCESSCERTNOTE	VARCHAR2(50 CHAR) DEFAULT NULL,	Textual note on Load status: Controls Submitted Controls Cleared Not Submitted No Controls Found for Profile No Controls Group Assigned Error Setting Status
PROCESSASSES	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE,	Status for Assessment (process explorer) step: O=Not started or unsubmitted 1=Submitted
PROCESSASSESNOTE	VARCHAR2(50 CHAR) DEFAULT NULL,	Textual note on Load status: Controls Submitted Controls Cleared Not Submitted No Controls Found for Profile No Controls Group Assigned Error Setting Status
PROCESSCHILDDONE	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Certification status for parent locations step: O=Not started or all children not complete 1=All children complete
PROCESSCHILDDONENOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Certification status for parent location: Children Submitted No Children
PROCESSUD1	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not used
PROCESSUD1NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not used
PROCESSUD2	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not used

Column Name	Definition	Description
PROCESSUD2NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not used
PROCESSUD3	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not used
PROCESSUD3NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not used
PROCESSUD4	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not used
PROCESSUD4NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not used
PROCESSENDTIME	DATE DEFAULT TO_ DATE('01/01/1900', 'MM/ DD/YYYY') NOT NULL ENABLE	Last update time/date
BLNWCDIRTY	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Flag used to indicate that maps (WC = WildCard) must be recalculated prior to validating: • 0=0K • 1=Re-caculate location
		Map rules have changed after data was imported. This causes the Calculate flag to be displayed
BLNLOGICDIRTY	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Flag used to indicate the LOGIC must be recalculated prior to validating
BLNVALDIRTY	NUMBER(1,0) DEFAULT 1 NOT NULL ENABLE	Flag used to indicate when Validation workflow should be re-run: • 0=0K • 1=re-process validations Map rules have changed after data was imported. This causes the Calculate flag to be displayed.
INTLOCKSTATE	NUMBER(6,0) DEFAULT 50 NOT NULL ENABLE	Location POV lock status: • 50=open • 60=locked
PROCESSSTATUS	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Current state of workflow for location/category/per. Valid status from tLogProcessStates



Setting up Jython, Eclipse and Python

This appendix explains how to setup Jython, Eclipse and PyDev for use with the FDMEE scripting component.

This section is specific to the 11.1.2.3.100 release of Oracle Hyperion Financial Data Quality Management, Enterprise Edition. Because Jython, Eclipse and Python are not Oracle products, always refer to these products' documentation for the latest updates and changes. Reference the following URL for more information: http://pydev.org/manual 101 root.html

- ➤ To set up Jython:
- 1 Create a working directory.

For example, create: C:\FDMEE.

2 Download Jython to the working directory.

The download is available athttp://www.jython.org/downloads.html

Double click the Jython installer jar and select the following options:

- English Language
- Standard Install
- Default Location (C:\FDMEE\jython2.5.1)
- Current Java Home
- 3 Download Eclipse to working directory and extract.

The Eclipse download is available at http://www.eclipse.org/downloads

- 4 Launch Eclipse from the C:\FDMEE\eclipse\eclipse.exe.
- 5 If the Welcome page is displayed, skip it by selecting the Workbench link in the top right corner.
- 6 Select the following menu option to update your preferences:
 - a. Select **Windows** and then **Preferences**.
 - b. Add **PyDev** to the Available Software Sites:

To do this:

- i. Select Install/Update, then Available Software Sites, and then click Add.
- ii. In Name, enter PyDev.
- iii. In **Location**, enter http://pydev.org/updates/.
- 7 Change the default text editor options:

To do this:

- a. Select **General**, then **Editors**, and then **Text Editors**.
- b. For the **Displayed tab width**, enter **2**.
- c. Enable the **Insert spaces for tabs** option.
- d. Enable the **Show print margin** option.
- e. Enable the **Show line numbers** option.

8 Select the Menu option.

To do this:

- a. Select Help, and then Install New Software...
- b. In the **Work with** dropdown, select **PyDev**.
- c. From the PYDev availabl software list, select the PyDev node, and click Next.
- d. Confirm the PyDev for Eclipse installation, and click Next
- e. Accept the licensing terms, and click Finish
- f. Once the install starts, you are prompted about Aptana PyDev; and PyDev; Aptana certificates. Make sure you select the certificates and click **OK**.
- g. Restart **Eclipse** to complete the PyDev installation.

9 Update your PyDev preferences:

To do this:

- Select Windows and then Preferences.
- b. Configure the Jython interpreter used by PyDev:
 - i. Select **PyDev**, and then **Interpreter-Jython**.
 - ii. Click New.
 - iii. Click Browse.
 - iv. Select C:\FDMEE\jython2.5.1\jython.jar
 - v. Click **OK**.
 - vi. When prompted to select the folders to add to the SYSTEM python path, do not change any selection and click **OK**.
 - vii. Click **OK** to close the Preferences window.

Glossary

! See bang character.

#MISSING See missing data.

access permissions A set of operations that a user can perform on a resource.

account blocking The process by which accounts accept input data in the consolidated file. Blocked accounts do not receive their value through the additive consolidation process.

account eliminations Accounts which have their values set to zero in the consolidated file during consolidation.

account type A property that determines how an account's value flows over time and its sign behavior. Account type options can include expense, income, asset, liability, and equity.

accountability map A visual, hierarchical representation of the responsibility, reporting, and dependency structure of the accountability teams (also known as critical business areas) in an organization.

active service A service whose Run Type is set to Start rather than to Hold.

active-active high availability system A system in which all the available members can service requests, and no member is idle. An active-active system generally provides more scalability options than an active-passive system. Contrast with active-passive high availability system.

active-passive high availability system A system with active members, which are always servicing requests, and passive members that are activated only when an active member fails. Contrast with active-active high availability system.

activity-level authorization Defines user access to applications and the types of activities they can perform on applications, independent of the data that will be operated on.

ad hoc report An online analytical query that an end user creates dynamically.

adapter Software that enables a program to integrate with data and metadata from target and source systems.

adaptive states Interactive Reporting Web Client level of permission.

adjustment See journal entry.

Advanced Relational Access The integration of a relational database with an Essbase multidimensional database so that all data remains in the relational database and is mapped to summary-level data in the Essbase database.

agent An Essbase server process that starts and stops applications and databases, manages connections from users, and handles user-access security. The agent is referred to as ESSBASE EXE.

aggregate cell A cell comprising several cells. For example, a data cell that uses Children(Year) expands to four cells containing Quarter 1, Quarter 2, Quarter 3, and Quarter 4 data.

aggregate function A type of function, such as sum or calculation of an average, that summarizes or performs analysis on data.

aggregate limit A limit placed on an aggregated request line item or aggregated metatopic item.

aggregate storage database The database storage model designed to support large-scale, sparsely distributed data which is categorized into many, potentially large dimensions. Upper level members and formulas are dynamically calculated, and selected data values are aggregated and stored, typically with improvements in overall aggregation time.

aggregate view A collection of aggregate cells based on the levels of the members within each dimension. To reduce calculation time, values are pre-aggregated and stored as aggregate views. Retrievals start from aggregate view totals and add up from there.

aggregation The process of rolling up and storing values in an aggregate storage database; the stored result of the aggregation process.

aggregation script In aggregate storage databases only, a file that defines a selection of aggregate views to be built into an aggregation.

alias table A table that contains alternate names for members.

alternate hierarchy A hierarchy of shared members. An alternate hierarchy is based upon an existing hierarchy in a database outline, but has alternate levels in the dimension. An alternate hierarchy allows the same data to be seen from different points of view.

ancestor A branch member that has members below it. For example, the members Qtr2 and 2006 are ancestors of the member April.

appender A Log4j term for destination.

application 1) A software program designed to run a specific task or group of tasks such as a spreadsheet program or database management system; 2) A related set of dimensions and dimension members that are used to meet a specific set of analytical requirements, reporting requirements, or both.

application administrator A person responsible for setting up, configuring, maintaining, and controlling an application. Has all application privileges and data access permissions.

application currency The default reporting currency for the application.

Application Migration Utility A command-line utility for migrating applications and artifacts.

application server cluster A loosely joined group of application servers running simultaneously, working together for reliability and scalability, and appearing to users as one application server instance. See also vertical application cluster and horizontal application cluster.

area A predefined set of members and values that makes up a partition.

arithmetic data load A data load that performs operations on values in the database, such as adding 10 to each value.

artifact An individual application or repository item; for example, scripts, forms, rules files, Interactive Reporting documents, and financial reports. Also known as an object.

assemblies Installation files for EPM System products or components.

asset account An account type that stores values that represent a company's assets.

assignment The association of a source and destination in the allocation model that controls the direction of allocated costs or revenue flow.

asymmetric topology An Oracle Fusion Middleware Disaster Recovery configuration that is different across tiers on the production site and standby site. For example, an asymmetric topology can include a standby site with fewer hosts and instances than the production site.

attribute A characteristic of a dimension member. For example, Employee dimension members may have attributes of Name, Age, or Address. Product dimension members can have several attributes, such as a size and flavor.

attribute association A relationship in a database outline whereby a member in an attribute dimension describes a characteristic of a member of its base dimension. For example, when product 100-10 has a grape flavor, the product 100-10 has the Flavor attribute association of grape. Thus, the 100-10 member of the Product dimension is associated with the Grape member of the Flavor attribute dimension.

Attribute Calculations dimension A system-defined dimension that performs these calculation operations on groups of members: Sum, Count, Avg, Min, and Max. This dimension is calculated dynamically and is not visible in the database outline. For example, using the Avg member, you can calculate the average sales value for Red products in New York in January.

attribute dimension A type of dimension that enables analysis based on the attributes or qualities of dimension members.

attribute reporting A reporting process based on the attributes of the base dimension members. See also base dimension.

attribute type A text, numeric, Boolean, date, or linked-attribute type that enables different functions for grouping, selecting, or calculating data. For example, because the Ounces attribute dimension has the type numeric, the number of ounces specified as the attribute of each product can be used to calculate the profit per ounce for that product.

authentication Verification of identity as a security measure. Authentication is typically based on a user name and password. Passwords and digital signatures are forms of authentication.

authentication service A core service that manages one authentication system.

auto-reversing journal A journal for entering adjustments that you want to reverse in the next period.

automated stage A stage that does not require human intervention; for example, a data load.

axis 1) A straight line that passes through a graphic used for measurement and categorization; 2) A report aspect used to arrange and relate multidimensional data, such as filters, pages, rows, and columns. For example, for a data query in Simple Basic, an axis can define columns for values for Qtr1, Qtr2, Qtr3, and Qtr4. Row data would be retrieved with totals in the following hierarchy: Market, Product.

backup A duplicate copy of an application instance.

balance account An account type that stores unsigned values that relate to a particular time.

balanced journal A journal in which the total debits equal the total credits.

bang character (!) A character that terminates a series of report commands and requests information from the database. A report script must be terminated with a bang character; several bang characters can be used within a report script.

base currency The currency in which daily business transactions are performed.

base dimension A standard dimension that is associated with one or more attribute dimensions. For example, assuming products have flavors, the Product dimension is the base dimension for the Flavors attribute dimension.

base entity An entity at the bottom of the organization structure that does not own other entities.

batch calculation Any calculation on a database that is done in batch; for example, a calculation script or a full database calculation. Dynamic calculations are not considered to be batch calculations.

batch file An operating system file that can call multiple ESSCMD scripts and run multiple sessions of ESSCMD. On Windows-based systems, batch files have BAT file extensions. On UNIX, batch files are written as a shell script.

Batch Loader An FDM component that enables the processing of multiple files.

batch POV A collection of all dimensions on the user POV of every report and book in the batch. While scheduling the batch, you can set the members selected on the batch POV.

batch processing mode A method of using ESSCMD to write a batch or script file that can be used to automate routine server maintenance and diagnostic tasks. ESSCMD script files can execute multiple commands and can be run from the operating system command line or from within operating system batch files. Batch files can be used to call multiple ESSCMD scripts or run multiple instances of ESSCMD.

block The primary storage unit which is a multidimensional array representing the cells of all dense dimensions.

block storage database The Essbase database storage model categorizing and storing data based on the sparsity of data values defined in sparse dimensions. Data values are stored in blocks, which exist only for sparse dimension members for which there are values.

Blocked Account An account that you do not want calculated in the consolidated file because you want to enter it manually.

book 1) In Financial Reporting, a container that holds a group of similar documents. Books may specify dimension sections or dimension changes; 2) In Data Relationship Management, a collection of exports that can be run together as a group. Export results can be combined together or output separately.

book POV The dimension members for which a book is run.

bookmark A link to a reporting document or a Web site, displayed on a personal page of a user. The types of bookmarks are My Bookmarks and image bookmarks.

bounding rectangle The required perimeter that encapsulates the Interactive Reporting document content when embedding Interactive Reporting document sections in a personal page, specified in pixels for height and width or row per page.

broadcast message A simple text message sent by an administrator to a user who is logged on to a Planning application. The message details information such as system availability, notification of application refresh, or application backups.

build method A method used to modify database outlines. Choice of a build method is based on the format of data in data source files.

business process A set of activities that collectively accomplish a business objective.

business rules Logical expressions or formulas that are created within an application to produce a desired set of resulting values.

cache A buffer in memory that holds data temporarily.

calc script A set of commands that define how a database is consolidated or aggregated. A calculation script may also contain commands that specify allocation and other calculation rules separate from the consolidation process.

Calculated Accounts Accounts with formulas that you cannot alter. These formulas are fixed to maintain the accounting integrity of the model that you are building. For example, the formula for Net Income, a Calculated Account, is modeled into Strategic Finance and cannot be changed in historical or forecast periods.

calculated member in MaxL DML A member designed for analytical purposes and defined in the optional WITH section of a MaxL DML query.

Calculation Manager A module of Enterprise Performance Management Architecture (EPMA) that Planning and Financial Management users can use to design, validate, and administrate business rules in a graphical environment. c

calculation status A consolidation status that indicates that some values or formula calculations have changed. You must reconsolidate to get the correct values for the affected entity.

calendar User-defined time periods and their relationship to each other. Q1, Q2, Q3, and Q4 comprise a calendar or fiscal year.

cascade The process of creating multiple reports for a subset of member values.

Catalog pane An area that displays a list of elements available to the active section. If Query is the active section, a list of database tables is displayed. If Pivot is the active section, a list of results columns is displayed. If Dashboard is the active section, a list of embeddable sections, graphic tools, and control tools are displayed.

categories Groupings by which data is organized. For example, Month.

cause and effect map A map that depicts how the elements that form your corporate strategy relate and how they work together to meet your organization's strategic goals. A Cause and Effect map tab is automatically created for each Strategy map.

CDF See custom-defined function.

CDM See custom-defined macro.

cell 1) The data value at the intersection of dimensions in a multidimensional database; the intersection of a row and a column in a worksheet; 2) A logical group of nodes belonging to one administrative domain.

cell note A text annotation for a cell in an Essbase database. Cell notes are a type of LRO.

CHANGED status Consolidation status that indicates data for an entity has changed.

chart template A template that defines the metrics to display in Workspace charts.

child A member with a parent above it in the database outline.

choice list A list of members that a report designer can specify for each dimension when defining the report's point of view. A user who wants to change the point of view for a dimension that uses a choice list can select only the members specified in that defined member list or those members that meet the criteria defined in the function for the dynamic list.

clean block A data block in which the database is fully calculated, when a calculation script calculates all dimensions at once, or when the SET CLEARUPDATESTATUS command is used in a calculation script.

cluster An array of servers or databases that behave as a single resource which share task loads and provide failover support; eliminates one server or database as a single point of failure in a system.

cluster interconnect A private link used by a hardware cluster for heartbeat information, to detect node failure.

cluster services Software that manages cluster member operations as a system. With cluster services, you can define a set of resources and services to monitor through a heartbeat mechanism between cluster members and to move these resources and services to a different cluster member as efficiently and transparently as possible.

clustered bar charts Charts in which categories are viewed side-by-side; used only with vertical bar charts.

code page A mapping of bit combinations to a set of text characters. Different code pages support different sets of characters. Each computer contains a code page setting for the character set requirements of the language of the computer user. In the context of this document, code pages map characters to bit combinations for non-Unicode encodings. See also encoding.

column In Data Relationship Management, a field of data associated with an import source or the results of a query, compare, validation, or export.

committed access An Essbase Kernel Isolation Level setting that affects how Essbase handles transactions. Under committed access, concurrent transactions hold long-term write locks and yield predictable results.

computed item A virtual column (as opposed to a column that is physically stored in the database or cube) that can be calculated by the database during a query, or by Interactive Reporting Studio in the Results section. Computed items are calculations of data based on functions, data items, and operators provided in the dialog box and can be included in reports or reused to calculate other data.

connection file See Interactive Reporting connection file (.oce)

consolidated file (Parent) A file into which all of the business unit files are consolidated; contains the definition of the consolidation.

consolidation The process of aggregating data from dependent entities to parent entities. For example, when the dimension Year consists of the members Qtr1, Qtr2, Qtr3, and Qtr4, its consolidation is Year.

consolidation file (*.cns) A graphical interface that enables you to add, delete, or move Strategic Finance files in the consolidation process using either a Chart or Tree view. It also enables you to define and modify the consolidation.

consolidation rule The rule that is executed during the consolidation of the node of the hierarchy. This rule can contain customer-specific formulas appropriate for the correct consolidation of parent balances. Elimination processing can be controlled within these rules.

content Information stored in the repository for any type of file.

content browser A component that earbles users to browse and select content to be placed on a Workspace Page.

context variable A variable that is defined for a particular task flow to identify the context of the taskflow instance.

contribution The value added to a parent from a child entity. Each child has a contribution to its parent.

controls groups Groupings used in FDM to maintain and organize certification and assessment information, especially helpful for meeting Sarbanes-Oxley requirements.

conversion rate See exchange rate.

cookie A segment of data placed on your computer by a Web site.

correlated subqueries Subqueries that are evaluated once for every row in the parent query; created by joining a topic item in the subquery with a topic in the parent query.

critical business area (CBA) An individual or a group organized into a division, region, plant, cost center, profit center, project team, or process; also called accountability team or business area.

critical success factor (CSF) A capability that must be established and sustained to achieve a strategic objective; owned by a strategic objective or a critical process and is a parent to one or more actions.

crosstab reporting Reporting that categorizes and summarizes data in table format. The table cells contain summaries of the data that fit within the intersecting categories. For example, a crosstab report of product sales information could show size attributes, such as Small and Large, as column headings and color attributes, such as Blue and Yellow, as row headings. The cell in the table where Large and Blue intersect could contain the total sales of all Blue products that are sized Large.

cube A block of data that contains three or more dimensions. An Essbase database is a cube.

cube deployment In Essbase Studio, the process of setting load options for a model to build an outline and load data into an Essbase application and database.

cube schema In Essbase Studio, the metadata elements, such as measures and hierarchies, representing the logical model of a cube.

currency conversion A process that converts currency values in a database from one currency into another. For example, to convert one U. S. dollar into the European euro, the exchange rate (for example, 0.923702) is multiplied by the dollar (1* 0.923702). After conversion, the European euro amount is .92.

Currency Overrides A feature allowing the selected input method for any input period to be overridden to enable input of that period's value as Default Currency/Items. To override the input method, enter a pound sign (#) before or after the number.

currency partition A dimension type that separates local currency members from a base currency, as defined in an application. Identifies currency types, such as Actual, Budget, and Forecast.

custom calendar Any calendar created by an administrator.

custom dimension A dimension created and defined by users. Channel, product, department, project, or region could be custom dimensions.

custom property A property of a dimension or dimension member that is created by a user.

custom report A complex report from the Design Report module, composed of any combination of components.

custom-defined function (CDF) Essbase calculation functions developed in Java and added to the standard Essbase calculation scripting language using MaxL. See also custom-defined macro.

custom-defined macro (CDM) Essbase macros written with Essbase calculator functions and special macro functions. Custom-defined macros use an internal Essbase macro language that enables the combination of calculation functions and they operate on multiple input parameters. See also custom-defined function.

cycle through Perform multiple passes through a database while calculating it.

dashboard A collection of metrics and indicators that provide an interactive summary of your business. Dashboards enable you to build and deploy analytic applications.

data cache A buffer in memory that holds uncompressed data blocks.

data cell See cell.

data file cache A buffer in memory that holds compressed data (PAG) files.

data form A grid display that enables users to enter data into the database from an interface such as a Web browser, and to view and analyze data or related text. Certain dimension member values are fixed, giving users a specific view into the data.

data function Function that computes aggregate values, including averages, maximums, counts, and other statistics that summarize groupings of data.

data load location In FDM, a reporting unit responsible for submitting source data into the target system. Typically, one FDM data load location exists for each source file loaded to the target system.

data load rules A set of criteria that determines how to load data from a text-based file, a spreadsheet, or a relational data set into a database.

data lock A feature that prevents changes to data according to specified criteria, such as a period or scenario.

data model A representation of a subset of database tables.

data value See cell.

database connection A file that stores definitions and properties used to connect to data sources and enables database references to be portable and widely used.

date measure In Essbase, a member tagged as Date in the dimension where measures are represented. The cell values are displayed as formatted dates. Dates as measures can be useful for analysis types that are difficult to represent using the Time dimension. For example, an application may need to track acquisition dates for a series of capital assets, but the acquisition dates span too large a period to allow for feasible Time dimension modeling. See also typed measure.

Default Currency Units The unit scale of data. For example, If you select to define your analysis in thousands and enter 10, this unit is interpreted as 10,000.

dense dimension In block storage databases, a dimension likely to contain data for every combination of dimension members. For example, time dimensions are often dense because they can contain all combinations of all members. Contrast with sparse dimension.

dependent entity An entity that is owned by another entity in the organization.

derived text measure In Essbase Studio, a text measure whose values are governed by a predefined rule expressed as a range. For example, a derived text measure, called Sales Performance Index, based on a measure Sales, could consist of the values High, Medium, and Low. This derived text measure is defined to display High, Medium, and Low, depending on the range in which the corresponding sales values fall. See also text measure.

descendant Any member below a parent in the database outline. In a dimension that includes years, quarters, and months, the members Qtr2 and April are descendants of the member Year.

Design Report An interface in Web Analysis Studio for designing custom reports, from a library of components.

destination 1) In Business Rules, a block of the database where calculated values are stored; 2) In Profitability and Cost Management, the association of a source and destination in the allocation model that controls the direction of allocated costs or revenue flow.

destination currency The currency to which balances are converted. You enter exchange rates and convert from the source currency to the destination currency. For example, when you convert from EUR to USD, the destination currency is USD.

detail chart A chart that provides the detailed information that you see in a Summary chart. Detail charts appear in the Investigate Section in columns below the Summary charts. If the Summary chart shows a Pie chart, then the Detail charts below represent each piece of the pie.

dimension A data category used to organize business data for the retrieval and preservation of values. Dimensions usually contain hierarchies of related members grouped within them. For example, a Year dimension often includes members for each time period, such as quarters and months.

dimension build The process of adding dimensions and members to an Essbase outline.

dimension build rules Specifications, similar to data load rules, that Essbase uses to modify an outline. The modification is based on data in an external data source file.

dimension tab In the Pivot section, the tab that enables you to pivot data between rows and columns.

dimension table 1) A table that includes numerous attributes about a specific business process; 2) In Essbase Integration Services, a container in the OLAP model for one or more relational tables that define a potential dimension in Essbase.

dimension type A dimension property that enables the use of predefined functionality. Dimensions tagged as time have a predefined calendar functionality.

dimensionality In MaxL DML, the represented dimensions (and the order in which they are represented) in a set. For example, the following set consists of two tuples of the same dimensionality, because they both reflect the dimensions (Region, Year): { (West, Feb), (East, Mar) }

direct rate A currency rate that you enter in the exchangerate table. The direct rate is used for currency conversion. For example, to convert balances from JPY to USD, in the exchange-rate table, enter a rate for the period/scenario where the source currency is JPY and the destination currency is USD.

dirty block A data block containing cells that have been changed since the last calculation. Upper-level blocks are marked as dirty when their child blocks are dirty (that is, when they have been updated).

Disaster Recovery The ability to safeguard against natural or unplanned outages at a production site by having a recovery strategy for applications and data to a geographically separate standby site.

display type One of three Web Analysis formats saved to the repository: spreadsheet, chart, and pinboard.

dog-ear The flipped page corner in the upper-right corner of the chart header area.

drill-down Navigation through the query result set using the dimensional hierarchy. Drilling down moves the user perspective from aggregated data to detail. For example, drilling down can reveal hierarchical relationships between years and quarters or quarters and months.

drill-through The navigation from a value in one data source to corresponding data in another source.

driver In Profitability and Cost Management, an allocation method that describes the mathematical relationship between the sources that use the driver and the destinations to which those sources allocate cost or revenue. For Business Modeling, see also cost driver and activity driver.

duplicate alias name A name that occurs more than once in an alias table and can be associated with more than one member in a database outline. Duplicate alias names can be used with duplicate member outlines only.

duplicate member name Multiple occurrences of a member name in a database, with each occurrence representing a different member. For example, a database has two members named New York. One member represents New York state and the other member represents New York city.

duplicate member outline A database outline containing duplicate member names.

Dynamic Calc and Store members Members in a block storage outline that Essbase calculates only upon the first retrieval of the value. Essbase then stores the calculated value in the database. Subsequent retrievals do not require calculating.

Dynamic Calc members Members in a block storage outline that Essbase calculates only at retrieval time. Essbase discards calculated values after completing the retrieval request.

dynamic calculation In Essbase, a calculation that occurs only when you retrieve data on a member that is 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.

dynamic hierarchy In aggregate storage database outlines only, a hierarchy in which members are calculated at retrieval time.

dynamic member list A system-created named member set that is based on user-defined criteria. The list is refreshed automatically whenever it is referenced in the application. As dimension members are added and deleted, the list automatically reapplies the criteria to reflect the changes.

dynamic reference A pointer in the rules file to header records in a data source.

dynamic report A report containing data that is updated when you run the report.

Dynamic Time Series A process that performs period-to-date reporting in block storage databases.

dynamic view account An account type indicating that account values are calculated dynamically from the data that is displayed.

Eliminated Account An account that does not appear in the consolidated file.

elimination The process of zeroing out (eliminating) transactions between entities within an organization.

employee A user responsible for, or associated with, specific business objects. Employees need not work for an organization; for example, they can be consultants. Employees must be associated with user accounts, for authorization purposes.

encoding A method for mapping bit combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41. See also code page, locale.

ending period A period enabling you to adjust the date range in a chart. For example, an ending period of "month" produces a chart showing information through the end of the current month.

Enterprise View An Administration Services feature that enables management of the Essbase environment from a graphical tree view. From Enterprise View, you can operate directly on Essbase artifacts.

entity A dimension representing organizational units. Examples: divisions, subsidiaries, plants, regions, products, or other financial reporting units.

EPM Oracle home A subdirectory of Middleware home containing the files required by EPM System products. The EPM Oracle home location is specified during installation with EPM System Installer.

EPM Oracle instance A directory containing active, dynamic components of EPM System products (components that can change during run-time). You define the EPM Oracle instance directory location during configuration with EPM System Configurator.

Equity Beta The riskiness of a stock, measured by the variance between its return and the market return, indicated by an index called "beta." For example, when a stock's return normally moves up or down 1.2% when the market moves up or down 1%, the stock has a beta of 1.2.

essbase.cfg An optional configuration file for Essbase. Administrators may edit this file to customize Essbase Server functionality. Some configuration settings may also be used with Essbase clients to override Essbase Server settings.

EssCell A function entered into a cell in Essbase Spreadsheet Add-in to retrieve a value representing an intersection of specific Essbase database members.

ESSCMD A command-line interface for performing Essbase operations interactively or through batch script files.

ESSLANG The Essbase environment variable that defines the encoding used to interpret text characters. See also encoding.

ESSMSH See MaxL Shell.

exceptions Values that satisfy predefined conditions. You can define formatting indicators or notify subscribing users when exceptions are generated.

exchange rate type An identifier for an exchange rate. Different rate types are used because there may be multiple rates for a period and year. Users traditionally define rates at period end for the average rate of the period and for the end of the period. Additional rate types are historical rates, budget rates, forecast rates, and so on. A rate type applies to a specific time.

expense account An account that stores periodic and year-to-date values that decrease net worth when they are positive.

Explorer A client/server-based tool that delivers query, analysis, and reporting capabilities for power users who need to directly access data sources or to explore the information organized in prebuilt data models stored in the repository.

Extensible Markup Language (XML) A language comprising a set of tags used to assign attributes to data that can be interpreted between applications according to a schema.

external authentication Logging on to Oracle EPM System products with user information stored outside the application. The user account is maintained by the EPM System, but password administration and user authentication are performed by an external service, using a corporate directory such as Oracle Internet Directory (OID) or Microsoft Active Directory (MSAD).

externally triggered events Non-time-based events for scheduling job runs.

Extract, Transform, and Load (ETL) Data-source-specific programs for extracting data and migrating it to applications.

extraction command An Essbase reporting command that handles the selection, orientation, grouping, and ordering of raw data extracted from a database; begins with the less-than (<) character.

fact table The central table in a star join schema, characterized by a foreign key and elements drawn from a dimension table. This table typically contains numeric data that can be related to all other tables in the schema.

failover The ability to switch automatically to a redundant standby database, server, or network when the primary database, server, or network fails or is shut down. A system that is clustered for failover provides high availability and fault tolerance through server redundancy and fault-tolerant hardware, such as shared disks.

Favorites gadget A gadget that contains links to Reporting and Analysis documents and URLs. See also gadget.

file delimiter A character, such as a comma or tab, that separates fields in a data source.

filter A constraint on data sets that restricts values to specific criteria; for example, to exclude certain tables, metadata, or values, or to control access.

flow account An unsigned account that stores periodic and year-to-date values.

footer Text or images at the bottom of report pages, containing dynamic functions or static text such as page numbers, dates, logos, titles or file names, and author names.

format string 1) In Essbase, a method for transforming the way cell values are displayed; 2) In Data Relationship Management, a parameter of a Format or Formatted Date derived property that indicates the format in which a property value should be returned.

formula In Data Relationship Management, business logic used by a derived property to dynamically calculate a property value.

frame An area on the desktop. Two main areas: the navigation and workspace frames.

free-form grid An object for presenting, entering, and integrating data from different sources for dynamic calculations.

free-form reporting Creating reports by entering dimension members or report script commands in worksheets.

function In Data Relationship Management, a syntactic element of a derived property formula that accepts parameters and returns dynamic values.

gadget A simple, specialized, lightweight application that provides easy viewing of EPM content and enables access to core Reporting and Analysis functionality.

geneology data Additional data that is optionally generated after allocation calculations. This data enables reporting on all cost or revenue flows from start to finish through all allocation steps.

generation A layer in a hierarchical tree structure that defines member relationships in a database. Generations are ordered incrementally from the top member of the dimension (generation 1) down to the child members. Use the unique generation name to identify a layer in the hierarchical tree structure.

generic jobs Non-SQR Production Reporting or non-Interactive Reporting jobs.

global report command A command in a running report script that is effective until it is replaced by another global command or the file ends.

grid POV A means for specifying dimension members on a grid without placing dimensions in rows, columns, or page intersections. A report designer can set POV values at the grid level, preventing user POVs from affecting the grid. If a dimension has one grid value, you put the dimension into the grid POV instead of the row, column, or page.

group A container for assigning similar access permissions to multiple users.

GUI Graphical user interface

hardware cluster a collection of computers that provides a single view of network services (for example, an IP address) or application services (such as databases and Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

high availability A system attribute that enables an application to continue to provide services in the presence of failures. This is achieved through removal of single points of failure, with fault-tolerant hardware, as well as server clusters; when one server fails, processing requests are routed to another server.

Historical Average An average for an account over a number of historical periods.

holding company An entity that is part of a legal entity group, with direct or indirect investments in all entities in the group.

horizontal application server cluster A cluster with application server instances on different machines.

host A server on which applications and services are installed.

host properties Properties pertaining to a host, or when the host has multiple Oracle EPM homes, to an Oracle EPM home.

Hybrid Analysis An analysis mapping low-level data stored in a relational database to summary-level data stored in Essbase, combining the mass scalability of relational systems with multidimensional data.

hyperlink A link to a file, a Web page, or an intranet HTML page.

Hypertext Markup Language (HTML) A programming language specifying how Web browsers display data.

identity A unique identification for a user or group in external authentication.

image bookmarks Graphic links to Web pages or repository items.

IMPACTED status A status that indicates changes in child entities consolidating into parent entities.

implied share A member with one or more children but only one that is consolidated, so the parent and child share a value.

import format In FDM, the definition of the structure of the source file that enables the loading of a source data file to an FDM data-load location.

inactive group A group for which an administrator has deactivated system access.

INACTIVE status A status that indicates entities deactivated from consolidation for the current period.

inactive user A user whose account was deactivated by an administrator.

income account An account storing periodic and year-to-date values that, when positive, increase net worth.

index 1) A method where Essbase uses sparse-datacombinations to retrieve data in block storage databases. 2)The index file.

index cache A buffer containing index pages.

index entry A pointer to an intersection of sparse dimensions. Index entries point to data blocks on disk and use offsets to locate cells.

index file An Essbase file storing block storage data retrieval information, residing on disk, and containing index pages.

index page A subdivision in an index file. An index page contains pointers to data blocks.

input data Data loaded from a source rather than calculated.

installation assemblies Product installation files that plug in to EPM System Installer.

integration A process that is run to move data between Oracle's Hyperion applications using Shared Services. Data integration definitions specify the data moving between a source application and a destination application, and they enable the data movements to be grouped, ordered, and scheduled.

intelligent calculation A calculation method tracking updated data blocks since the last calculation.

Interactive Reporting connection file (.oce) Files encapsulating database connection information, including the database API (ODBC, SQL*Net, and so on), database software, the database server network address, and database user name. Administrators create and publish Interactive Reporting connection (.oce) files.

intercompany elimination See elimination.

intercompany matching The process of comparing balances for pairs of intercompany accounts within an application. Intercompany receivables are compared to intercompany payables for matches. Matching accounts are used to eliminate intercompany transactions from an organization's consolidated totals.

intercompany matching report A report that compares intercompany account balances and indicates whether the accounts are in balance.

interdimensional irrelevance A situation in which a dimension does not intersect with other dimensions. Because the data in the dimension cannot be accessed from the nonintersecting dimensions, the nonintersecting dimensions are not relevant to that dimension.

intersection A unit of data representing the intersection of dimensions in a multidimensional database; also, a worksheet cell.

intrastage assignment An assignment in the financial flow to an object within the same stage.

introspection A deep inspection of a data source to discover hierarchies based on the inherent relationships in the database. Contrast with scraping.

Investigation See drill-through.

isolation level An Essbase Kernel setting that determines the lock and commit behavior of database operations. Choices are: committed access and uncommitted access.

iteration A pass of the budget or planning cycle in which the same version of data is revised and promoted.

Java Database Connectivity (JDBC) A client-server communication protocol used by Java-based clients and relational databases. The JDBC interface provides a call-level API for SQL-based database access.

Java web application server cluster An active-active application server cluster of Java Virtual Machines (JVMs).

job output Files or reports produced from running a job.

jobs Documents with special properties that can be launched to generate output. A job can contain Interactive Reporting, SQR Production Reporting, or generic documents.

join A link between two relational database tables or topics based on common content in a column or row. A join typically occurs between identical or similar items within different tables or topics. For example, a record in the Customer table is joined to a record in the Orders table because the Customer ID value is the same in each table.

journal entry (JE) A set of debit-credit adjustments to account balances for a scenario and period.

JSP Java Server Page.

KeyContacts gadget A gadget that contains a group of Smart Space users and provides access to Smart Space Collaborator. For example, you can have a KeyContacts gadget for your marketing team and another for your development team. See also gadget.

latest A spreadsheet keyword used to extract data values from the member defined as the latest time period.

layer 1) The horizontal location of members in a hierarchical structure, specified by generation (top down) or level (bottom up); 2) Position of objects relative to other objects. For example, in the Sample Basic database, Qtr1 and Qtr4 are in the same layer, so they are also in the same generation, but in a database with a ragged hierarchy, Qtr1 and Qtr4 might not be in same layer, though they are in the same generation.

layout area An area on a Workspace Page where content can be placed.

legend box A box containing labels that identify the data categories of a dimension.

level A layer in a hierarchical tree structure that defines database member relationships. Levels are ordered from the bottom dimension member (level 0) up to the parent members.

level 0 block A data block for combinations of sparse, level 0 members.

level 0 member A member that has no children.

liability account An account type that stores "point in time" balances of a company's liabilities. Examples: accrued expenses, accounts payable, and long-term debt.

lifecycle management The process of migrating an application, a repository, or individual artifacts across product environments.

line item detail The lowest level of detail in an account.

lineage The relationship between different metadata elements showing how one metadata element is derived from one or more other metadata elements, ultimately tracing the metadata element to its physical source. In Essbase Studio, a lineage viewer displays the relationships graphically. See also traceability.

link 1) A reference to a repository object. Links can reference folders, files, shortcuts, and other links; 2) In a taskflow, the point where the activity in one stage ends and another begins.

link condition A logical expression evaluated by the taskflow engine to determine the sequence of launching taskflow stages.

linked data model Documents that are linked to a master copy in a repository

linked partition A shared partition that enables you to use a data cell to link two databases. When a user clicks a linked cell in a worksheet, Essbase opens a new sheet displaying the dimensions in the linked database. The user can then drill down those dimensions.

linked reporting object (LRO) A cell-based link to an external file such as cell notes, URLs, or files with text, audio, video, or pictures. (Only cell notes are supported for Essbase LROs in Financial Reporting.) Contrast with local report object.

load balancer Hardware or software that directs the requests to individual application servers in a cluster and is the only point of entry into the system.

load balancing Distribution of requests across a group of servers, which helps to ensure optimal end user performance.

local currency An input currency type. When an input currency type is not specified, the local currency matches the entity's base currency.

local report object A report object that is not linked to a Financial Reporting report object in Explorer. Contrast with linked reporting object.

local results A data model's query results. Results can be used in local joins by dragging them into the data model. Local results are displayed in the catalog when requested.

locale A computer setting that specifies a location's language, currency and date formatting, data sort order, and the character set encoding used on the computer. Essbase uses only the encoding portion. See also encoding, ESSLANG.

locale header record A text record at the beginning of some non-Unicode-encoded text files, such as scripts, that identifies the encoding locale.

location alias A descriptor that identifies a data source. The location alias specifies a server, application, database, user name, and password. Location aliases are set by DBAs at the database level using Administration Services Console, ESSCMD, or the API.

locked A user-invoked process that prevents users and processes from modifying data.

locked data model A data model that cannot be modified by a user.

LOCKED status A consolidation status indicating that an entity contains data that cannot be modified.

Log Analyzer An Administration Services feature that enables filtering, searching, and analysis of Essbase logs.

logic group In FDM, one or more logic accounts generated after a source file is loaded into FDM. Logic accounts are calculated accounts derived from the source data.

logical address for web applications An aliased reference used to identify the internal host name, port, and context of a Web application. In a clustered or high-availability environment, this is the alias name that establishes a single internal reference for the distributed components. In EPM System, a nonclustered logical address for a web application defaults to the physical host running the web application.

LRO See linked reporting object.

managed server An application server process running in its own Java Virtual Machine (JVM).

manual stage A stage that requires human intervention.

Map File A file that stores the definition for sending data to or retrieving data from an external database. Map files have different extensions (.mps to send data; .mpr to retrieve data).

Map Navigator A feature that displays your current position on a Strategy, Accountability, or Cause and Effect map, indicated by a red outline.

Marginal Tax Rate The rate used to calculate the after-tax cost of debt; represents the tax rate applied to the last earned income dollar (the rate from the highest tax bracket into which income falls) and includes federal, state, and local taxes. Based on current level of taxable income and tax bracket, you can predict marginal tax rate.

Market Risk Premium The additional rate of return paid over the risk-free rate to persuade investors to hold "riskier" investments than government securities. Calculated by subtracting the risk-free rate from the expected market return. These figures should closely model future market conditions.

master data model An independent data model that is referenced as a source by multiple queries. When used, "Locked Data Model" is displayed in the Query section's Content pane; the data model is linked to the master data model displayed in the Data Model section, which an administrator may hide.

mathematical operator A symbol that defines how data is calculated in formulas and outlines. Can be any of the standard mathematical or Boolean operators; for example, +, -, *, /, and %.

MaxL The multidimensional database access language for Essbase, consisting of a data definition language (MaxL DDL) and a data manipulation language (MaxL DML). See also MaxL DDL, MaxL DML, and MaxL Shell

MaxL DDL The data definition language used by Essbase for batch or interactive system-administration tasks.

MaxL DML The data manipulation language used in Essbase for data query and extraction.

MaxL Perl Module A Perl module (essbase.pm) that is part of Essbase MaxL DDL. This module can be added to the Perl package to provide access to Essbase databases from Perl programs.

MaxL Script Editor A script-development environment in Administration Services Console. MaxL Script Editor is an alternative to using a text editor and the MaxL Shell for administering Essbase with MaxL scripts.

MaxL Shell An interface for passing MaxL statements to Essbase Server. The MaxL Shell executable file is located in the Essbase bin directory (UNIX: essmsh; Windows: essmsh.exe).

MDX (multidimensional expression) A language used for querying and calculation in multidimensional-compliant databases.

measures Numeric values in an OLAP database cube that are available for analysis. Measures are margin, cost of goods sold, unit sales, budget amount, and so on. See also fact table.

member A discrete component within a dimension. A member identifies and differentiates the organization of similar units. For example, a time dimension might include members Jan, Feb, and Qtr1.

member list A named system- or user-defined group that references members, functions, or member lists within a dimension.

member load In Essbase Integration Services, the process of adding dimensions and members (without data) to Essbase outlines.

member selection report command A type of Report Writer command that selects member ranges based on outline relationships, such as sibling, generation, and level.

member-specific report command A type of Report Writer formatting command that is executed as it is encountered in a report script. The command affects only its associated member and executes the format command before processing the member.

merge A data load option that clears values only from the accounts specified in the data load file and replaces them with values in the data load file.

metadata A set of data that defines and describes the properties and attributes of the data stored in a database or used by an application. Examples of metadata are dimension names, member names, properties, time periods, and security.

metadata elements Metadata derived from data sources and other metadata that is stored and cataloged for Essbase Studio use.

metadata sampling The process of retrieving a sample of members in a dimension in a drill-down operation.

metadata security Security set at the member level to restrict users from accessing certain outline members.

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

Middleware home A directory that includes the Oracle WebLogic Server home and can also include the EPM Oracle home and other Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.

migration audit report A report generated from the migration log that provides tracking information for an application migration.

migration definition file (.mdf) A file that contains migration parameters for an application migration, enabling batch script processing.

migration log A log file that captures all application migration actions and messages.

migration snapshot A snapshot of an application migration that is captured in the migration log.

MIME Type An attribute that describes the data format of an item, so that the system knows which application should open the object. A file's MIME (Multipurpose Internet Mail Extension) type is determined by the file extension or HTTP header. Plug-ins tell browsers which MIME types they support and which file extensions correspond to each MIME type.

minireport A report component that includes layout, content, hyperlinks, and the query or queries to load the report. Each report can include one or more minireports.

minischema A graphical representation of a subset of tables from a data source that represents a data modeling context.

missing data (#MISSING) A marker indicating that data in the labeled location does not exist, contains no value, or was never entered or loaded. For example, missing data exists when an account contains data for a previous or future period but not for the current period.

model 1) A file or content string containing an application-specific representation of data. Models are the basic data managed by Shared Services, of two major types: dimensional and nondimensional application objects; 2) In Business Modeling, a network of boxes connected to represent and calculate the operational and financial flow through the area being examined.

multidimensional database A method of organizing, storing, and referencing data through three or more dimensions. An individual value is the intersection point for a set of dimensions. Contrast with relational database.

Multiload An FDM feature that allows the simultaneous loading of multiple periods, categories, and locations.

My Workspace Page Customizable Workspace Pages created by users. They are marked specially so that they can be easily accessed from one single place without having to navigate the repository.

named set In MaxL DML, a set with its logic defined in the optional WITH section of a MaxL DML query. The named set can be referenced multiple times in the query.

native authentication The process of authenticating a user name and password from within the server or application.

nested column headings A report column heading format that displays data from multiple dimensions. For example, a column heading that contains Year and Scenario members is a nested column. The nested column heading shows Q1 (from the Year dimension) in the top line of the heading, qualified by Actual and Budget (from the Scenario dimension) in the bottom line of the heading.

NO DATA status A consolidation status indicating that this entity contains no data for the specified period and account.

non-dimensional model A Shared Services model type that includes application objects such as security files, member lists, calculation scripts, and Web forms.

non-unique member name See duplicate member name.

null value A value that is absent of data. Null values are not equal to zero.

numeric attribute range A feature used to associate a base dimension member that has a discrete numeric value with an attribute that represents a value range. For example, to classify customers by age, an Age Group attribute dimension can contain members for the following age ranges: 0-20, 21-40, 41-60, and 61-80. Each Customer dimension member can be associated with an Age Group range. Data can be retrieved based on the age ranges rather than on individual age values.

ODBC Open Database Connectivity. A database access method used from any application regardless of how the database management system (DBMS) processes the information.

OK status A consolidation status indicating that an entity has already been consolidated, and that data has not changed below it in the organization structure.

OLAP Metadata Catalog In Essbase Integration Services, a relational database containing metadata describing the nature, source, location, and type of data that is pulled from the relational data source.

OLAP model In Essbase Integration Services, a logical model (star schema) that is created from tables and columns in a relational database. The OLAP model is then used to generate the structure of a multidimensional database. See also online analytical processing (OLAP).

online analytical processing (OLAP) A multidimensional, multiuser, client-server computing environment for users who analyze consolidated enterprise data in real time. OLAP systems feature drill-down, data pivoting, complex calculations, trend analysis, and modeling.

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

Oracle home A directory containing the installed files required by a specific product, and residing within the directory structure of Middleware home. See also Middleware home.

organization An entity hierarchy that defines each entity and their relationship to others in the hierarchy.

origin The intersection of two axes.

outline The database structure of a multidimensional database, including all dimensions, members, tags, types, consolidations, and mathematical relationships. Data is stored in the database according to the structure defined in the outline.

outline synchronization For partitioned databases, the process of propagating outline changes from one database to another database.

P&L accounts (P&L) Profit and loss accounts. P&L refers to a typical grouping of expense and income accounts that comprise a company's income statement.

page A display of information in a grid or table often represented by the Z-axis. A page can contain data from one field, derived data from a calculation, or text.

page file An Essbase data file.

page heading A report heading type that lists members represented on the current page of the report. All data values on the page have the members in the page heading as a common attribute.

page member A member that determines the page axis.

palette A JASC-compliant file with a .PAL extension. Each palette contains 16 colors that complement each other and can be used to set the dashboard color elements.

parallel calculation A calculation option. Essbase divides a calculation into tasks and calculates some tasks simultaneously.

parallel data load In Essbase, the concurrent execution of data load stages by multiple process threads.

parallel export The ability to export Essbase data to multiple files. This may be faster than exporting to a single file, and it may resolve problems caused by a single data file becoming too large for the operating system to handle.

parent adjustments The journal entries that are posted to a child in relation to its parent.

parents The entities that contain one or more dependent entities that report directly to them. Because parents are entities associated with at least one node, they have entity, node, and parent information associated with them.

partition area A subcube within a database. A partition is composed of one or more areas of cells from a portion of the database. For replicated and transparent partitions, the number of cells within an area must be the same for the data source and target to ensure that the two partitions have the same shape. If the data source area contains 18 cells, the data target area must also contain 18 cells to accommodate the number of values.

partitioning The process of defining areas of data that are shared or linked between data models. Partitioning can affect the performance and scalability of Essbase applications.

pattern matching The ability to match a value with any or all characters of an item entered as a criterion. Missing characters may be represented by wild-card values such as a question mark (?) or an asterisk (*). For example, "Find all instances of apple" returns apple, but "Find all instances of apple*" returns apple, applesauce, applecranberry, and so on.

percent consolidation The portion of a child's values that is consolidated to its parent.

percent control The extent to which an entity is controlled within the context of its group.

percent ownership The extent to which an entity is owned by its parent.

performance indicator An image file used to represent measure and scorecard performance based on a range you specify; also called a status symbol. You can use the default performance indicators or create an unlimited number of your own.

periodic value method (PVA) A process of currency conversion that applies the periodic exchange rate values over time to derive converted results.

permission A level of access granted to users and groups for managing data or other users and groups.

persistence The continuance or longevity of effect for any Essbase operation or setting. For example, an Essbase administrator may limit the persistence of user name and password validity.

personal pages A personal window to repository information. You select what information to display and its layout and colors.

personal recurring time events Reusable time events that are accessible only to the user who created them.

personal variable A named selection statement of complex member selections.

perspective A category used to group measures on a scorecard or strategic objectives within an application. A perspective can represent a key stakeholder (such as a customer, employee, or shareholder/financial) or a key competency area (such as time, cost, or quality).

pinboard One of the three data object display types. Pinboards are graphics composed of backgrounds and interactive icons called pins. Pinboards require traffic lighting definitions.

pins Interactive icons placed on graphic reports called pinboards. Pins are dynamic. They can change images and traffic lighting color based on the underlying data values and analysis tools criteria.

pivot Alter the perspective of retrieved data. When Essbase first retrieves a dimension, it expands data into rows. You can then pivot or rearrange the data to obtain a different viewpoint.

planner A user who can input and submit data, use reports that others create, execute business rules, use task lists, enable e-mail notification for themselves, and use Smart View. Planners comprise the majority of users.

planning unit A data slice at the intersection of a scenario, version, and entity; the basic unit for preparing, reviewing, annotating, and approving plan data.

plot area The area bounded by X, Y, and Z axes; for pie charts, the rectangular area surrounding the pie.

plug account An account in which the system stores any outof-balance differences between intercompany account pairs during the elimination process.

post stage assignment Assignments in the allocation model that are assigned to locations in a subsequent model stage.

POV (point of view) A feature for setting data focus by selecting members that are not already assigned to row, column, or page axes. For example, selectable POVs in FDM could include location, period, category, and target category. In another example, using POV as a filter in Smart View, you could assign the Currency dimension to the POV and select the Euro member. Selecting this POV in data forms displays data in Euro values.

precalculation Calculating the database before user retrieval.

precision Number of decimal places displayed in numbers.

predefined drill paths Paths used to drill to the next level of detail, as defined in the data model.

presentation A playlist of Web Analysis documents, enabling reports to be grouped, organized, ordered, distributed, and reviewed. Includes pointers referencing reports in the repository.

preserve formulas User-created formulas kept within a worksheet while retrieving data.

primary measure A high-priority measure important to your company and business needs. Displayed in the Contents frame.

Process Monitor Report A list of locations and their positions within the FDM data conversion process. You can use the process monitor report to monitor the status of the closing process. The report is time-stamped. Therefore, it can be used to determine to which locations at which time data was loaded.

product In Shared Services, an application type, such as Planning or Performance Scorecard.

Production Reporting See SQR Production Reporting.

project An instance of Oracle's Hyperion products grouped together in an implementation. For example, a Planning project may consist of a Planning application, an Essbase cube, and a Financial Reporting Server instance.

provisioning The process of granting users and groups specific access permissions to resources.

proxy server A server acting as an intermediary between workstation users and the Internet to ensure security.

public job parameters Reusable named job parameters created by administrators and accessible to users with requisite access privileges.

public recurring time events Reusable time events created by administrators and accessible through the access control system.

PVA See periodic value method.

qualified name A member name in a qualified format that differentiates duplicate member names in a duplicate member outline. For example, [Market].[East].[State]. [New York] or [Market].[East].[City].[New York].

query governor An Essbase Integration Server parameter or Essbase Server configuration setting that controls the duration and size of queries made to data sources.

reciprocal assignment An assignment in the financial flow that also has the source as one of its destinations.

reconfigure URL A URL that is used to reload servlet configuration settings dynamically when users are already logged on to the Workspace.

record In a database, a group of fields making up one complete entry. For example, a customer record may contain fields for name, address, telephone number, and sales data.

recurring template A journal template for making identical adjustments in every period.

recurring time event An event specifying a starting point and the frequency for running a job.

redundant data Duplicate data blocks that Essbase retains during transactions until Essbase commits updated blocks.

regular journal A feature for entering one-time adjustments for a period. A regular journal can be balanced, balanced by entity, or unbalanced.

Related Accounts Accounts related to the main account and grouped under the same main account number. The account structure groups all main and related accounts under the same main account number. The main account is distinguished from related accounts by the first suffix of the account number.

relational database A type of database that stores data in related two-dimensional tables. Contrast with multidimensional database.

replace A data load option that clears existing values from all accounts for periods specified in the data load file and loads values from the data load file. If an account is not specified in the load file, its values for the specified periods are cleared.

replicated partition A portion of a database, defined through Partition Manager, used to propagate an update to data mastered at one site to a copy of data stored at another site. Users can access the data as though it were part of their local database.

Report Extractor An Essbase component that retrieves report data from the Essbase database when report scripts are run.

report object In report designs, a basic element with properties defining behavior or appearance, such as text boxes, grids, images, and charts.

report script A text file containing Essbase Report Writer commands that generate one or more production reports.

Report Viewer An Essbase component that displays complete reports after report scripts are run.

reporting currency The currency used to prepare financial statements, and converted from local currencies to reporting currencies.

repository Storage location for metadata, formatting, and annotation information for views and queries.

resources Objects or services managed by the system, such as roles, users, groups, files, and jobs.

restore An operation to reload data and structural information after a database has been damaged or destroyed, typically performed after shutting down and restarting the database.

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

result frequency The algorithm used to create a set of dates to collect and display results.

review level A Process Management review status indicator representing the process unit level, such as Not Started, First Pass, Submitted, Approved, and Published.

Risk Free Rate The rate of return expected from "safer" investments such as long-term U.S. government securities.

role The means by which access permissions are granted to users and groups for resources.

roll-up See consolidation.

root member The highest member in a dimension branch.

runtime prompt A variable that users enter or select before a business rule is run.

sampling The process of selecting a representative portion of an entity to determine the entity's characteristics. See also metadata sampling.

saved assumptions User-defined Planning assumptions that drive key business calculations (for example, the cost per square foot of office floor space).

scalability The ability to resize a system, making it larger or smaller. With regard to increases, "scale vertically" or "scale up" refers to expanding a single machine's capability. To "scale horizontally" or "scale out" refers to adding more machines.

scaling Scaling determines the display of values in whole numbers, tens, hundreds, thousands, millions, and so on.

scenario A dimension for classifying data; for example, Actuals, Budget, Forecast1, or Forecast2.

schema In relational databases, a logical model that represents the data and the relationships between the data.

scope The area of data encompassed by any Essbase operation or setting; for example, the area of data affected by a security setting. Most commonly, scope refers to three levels of granularity, where higher levels encompass lower levels. The levels, from highest to lowest: the entire system (Essbase Server), applications on Essbase Server, or databases within Essbase Server applications. See also persistence.

score The level at which targets are achieved, usually expressed as a percentage of the target.

scorecard A business object that represents the progress of an employee, strategy element, or accountability element toward goals. Scorecards ascertain this progress based on data collected for each measure and child scorecard added to the scorecard.

scraping An inspection of a data source to derive the most basic metadata elements from it. Contrast with introspection.

secondary measure A low-priority measure, less important than primary measures. Secondary measures do not have Performance reports but can be used on scorecards and to create dimension measure templates.

security agent A Web access management provider (for example, Oracle Access Manager, Oracle Single Sign-On, or CA SiteMinder) that protects corporate Web resources.

security platform A framework enabling Oracle EPM System products to use external authentication and single sign-on.

serial calculation The default calculation setting. Divides a calculation pass into tasks and calculates one task at a time.

services Resources that enable business items to be retrieved, changed, added, or deleted. Examples: Authorization and Authentication.

servlet A piece of compiled code executable by a Web server. **shared disks** See shared storage.

shared member A member that shares storage space with another member of the same name, preventing duplicate calculation of members that occur multiple times in an Essbase outline.

Shared Services Registry The part of the Shared Services repository that manages EPM System deployment information for most EPM System products, including installation directories, database settings, computer names, ports, servers, URLs, and dependent service data.

shared storage A set of disks containing data that must be available to all nodes of a failover cluster; also called shared disks.

Shared Workspace Pages Workspace Pages shared across an organization that are stored in a special System folder and can be accessed by authorized users from the Shared Workspace Pages Navigate menu.

sibling A child member at the same generation as another child member and having the same immediate parent. For example, the members Florida and New York are children of East and each other's siblings.

silent response files Files providing data that an installation administrator would otherwise be required to provide. Response files enable EPM System Installer or EPM System Configurator to run without user intervention or input.

single point of failure Any component in a system that, when it fails, prevents users from accessing the normal functionality.

single sign-on (SSO) The ability to log on once and then access multiple applications without being prompted again for authentication.

smart tags Keywords in Microsoft Office applications that are associated with predefined actions available from the Smart Tag menu. In Oracle EPM System products, smart tags can also be used to import Reporting and Analysis content and to access Financial Management and Essbase functions.

SmartCut A link to a repository item, in URL form.

snapshot Read-only data from a specific time.

source currency The currency from which values originate and are converted through exchange rates to the destination currency.

sparse dimension In block storage databases, a dimension unlikely to contain data for all member combinations when compared to other dimensions. Contrast with dense dimension. For example, not all customers have data for all products.

SPF files Printer-independent files created by an SQR Production Reporting server, containing a representation of the actual formatted report output, including fonts, spacing, headers, footers, and so on.

Spotlighter A tool that enables color coding based on selected conditions.

SQL spreadsheet A data object that displays the result set of a SQL query.

SQR Production Reporting A specialized programming language for data access, data manipulation, and creating SQR Production Reporting documents.

stage 1) A task description that forms one logical step within a taskflow, usually performed by an individual. A stage can be manual or automated; 2) For Profitability, logical divisions within the model that represent the steps in the allocation process within your organization.

stage action For automated stages, the invoked action that executes the stage.

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

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

standard dimension A dimension that is not an attribute dimension.

standard journal template A journal function used to post adjustments that have common adjustment information for each period. For example, you can create a standard template that contains the common account IDs, entity IDs, or amounts, and then use the template as the basis for many regular journals.

Status bar The bar at the bottom of the screen that displays helpful information about commands, accounts, and the current status of your data file.

stored hierarchy In aggregate storage databases outlines only, a hierarchy in which the members are aggregated according to the outline structure. Stored hierarchy members have certain restrictions; for example, they cannot contain formulas.

strategic objective (S0) A long-term goal defined by measurable results. Each strategic objective is associated with one perspective in the application, has one parent, the entity, and is a parent to critical success factors or other strategic objectives.

Strategy map Represents how the organization implements high-level mission and vision statements into lower-level, constituent strategic goals and objectives.

structure view Displays a topic as a simple list of component data items.

Structured Query Language A language used to process instructions to relational databases.

Subaccount Numbering A system for numbering subaccounts using nonsequential whole numbers.

subscribe Flags an item or folder to receive automatic notification whenever the item or folder is updated.

Summary chart In the Investigates Section, a chart that rolls up detail charts shown below in the same column, plotting metrics at the summary level at the top of each chart column.

supervisor A user with full access to all applications, databases, related files, and security mechanisms for a server.

supporting detail Calculations and assumptions from which the values of cells are derived.

suppress rows A setting that excludes rows containing missing values and underscores characters from spreadsheet reports.

symmetric multiprocessing (SMP) A server architecture that enables multiprocessing and multithreading. Performance is not significantly degraded when a large number of users simultaneously connect to an single instance.

symmetric topology An Oracle Fusion Middleware Disaster Recovery configuration that is identical across tiers on the production site and standby site. In a symmetric topology, the production site and standby site have the identical number of hosts, load balancers, instances, and applications. The same ports are used for both sites. The systems are configured identically and the applications access the same data.

sync Synchronization of Shared Services and application models.

synchronized The condition that exists when the latest version of a model resides in both the application and in Shared Services. See also model.

system extract A feature that transfers data from application metadata into an ASCII file.

tabs Navigable views of accounts and reports in Strategic Finance.

target Expected results of a measure for a specified period of time (day, quarter, and so on).

task list A detailed status list of tasks for a particular user.

taskflow The automation of a business process in which tasks are passed from one taskflow participant to another according to procedural rules.

taskflow definition Business processes in the taskflow management system that consist of a network of stages and their relationships; criteria indicating the start and end of the taskflow; and information about individual stages, such as participants, associated applications, associated activities, and so on.

taskflow instance A single instance of a taskflow including its state and associated data.

taskflow management system A system that defines, creates, and manages the execution of a taskflow, including definitions, user or application interactions, and application executables.

taskflow participant The resource that performs the task associated with the taskflow stage instance for both manual and automated stages.

Taxes - Initial Balances Strategic Finance assumes that the Initial Loss Balance, Initial Gain Balance, and Initial Balance of Taxes Paid entries have taken place in the period before the first Strategic Finance time period.

TCP/IP See Transmission Control Protocol/Internet Protocol.

text list In Essbase, an object that stores text values mapped to numeric identifiers. Text Lists enable the use of text measures.

text measure In Essbase, a member tagged as Text in the dimension where measures are represented. The cell values are displayed as predefined text. For example, the text measure Satisfaction Index may have the values Low, Medium, and High. See also typed measure, text list, derived text measure.

time dimension The time period that the data represents, such as fiscal or calendar periods.

time events Triggers for job execution.

time scale A scale that displays metrics by a specific time span, such as monthly or quarterly.

time series reporting A process for reporting data based on a calendar date (for example, year, quarter, month, or week).

Timeline Viewer An FDM feature that enables users to view dates and times of completed process flow steps for specific locations.

Title bar A bar that displays the Strategic Finance name, the file name, and the scenario name Version box.

toast message A message that fades in the lower-right corner of the screen.

token An encrypted identification of one valid user or group on an external authentication system.

top and side labels Column and row headings on the top and sides of a Pivot report.

top-level member A dimension member at the top of the tree in a dimension outline hierarchy, or the first member of the dimension in sort order when there is no hierarchical relationship among dimension members. If a hierarchical relationship exists, the top-level member name is generally the same as the dimension name.

trace allocations A Profitability feature that enables you to visually follow the flow of financial data, either forwards or backwards, from a single intersection throughout the model.

trace level The level of detail captured in a log file.

traceability The ability to track a metadata element to its physical source. For example, in Essbase Studio, a cube schema can be traced from its hierarchies and measure hierarchies to its dimension elements, date/time elements, measures, and, ultimately, to its physical source elements. See also lineage.

traffic lighting Color-coding of report cells, or pins based on a comparison of two dimension members, or on fixed limits.

transformation A process that transforms artifacts so that they function properly in the destination environment after application migration.

translation See currency conversion.

Transmission Control Protocol/Internet Protocol (TCP/IP) A standard set of communication protocols linking computers with different operating systems and internal architectures. TCP/IP utilities are used to exchange files, send mail, and store data to various computers that are connected to local and wide area networks.

transparent login A process that logs in authenticated users without launching the login screen.

transparent partition A shared partition that enables users to access and change data in a remote database as though it is part of a local database.

triangulation A means of converting balances from one currency to another through a third common currency. For example, to convert balances from the Danish krone to the British pound, balances could be converted from the krone to the euro and from the euro to the pound.

triggers An Essbase feature whereby data is monitored according to user-specified criteria that, when met, cause Essbase to alert the user or system administrator.

trusted user Authenticated user.

tuple MDX syntax element that references a cell as an intersection of a member from each dimension. If a dimension is omitted, its top member is implied. Examples: (Jan); (Jan, Sales); ([Jan], [Sales], [Cola], [Texas], [Actual]).

two-pass An Essbase property that is used to recalculate members that are dependent on the calculated values of other members. Two-pass members are calculated during a second pass through the outline.

typed measure In Essbase, a member tagged as Text or Date in the dimension where measures are represented. The cell values are displayed as predefined text or dates.

unary operator A mathematical indicator (+, -, *, /, %) associated with an outline member. The unary operator defines how the member is calculated during a database rollup.

Unicode-mode application An Essbase application wherein character text is encoded in UTF-8, enabling users with computers set up for different languages to share application data.

unique member name A nonshared member name that exists only once in a database outline.

unique member outline A database outline that is not enabled for duplicate member names.

upgrade The process of deploying a new software release and moving applications, data, and provisioning information from an earlier deployment to the new deployment.

upper-level block A type of data block wherein at least one of the sparse members is a parent-level member.

user directory A centralized location for user and group information, also known as a repository or provider. Popular user directories include Oracle Internet Directory (OID), Microsoft Active Directory (MSAD), and Sun Java System Directory Server.

user variable A variable that dynamically renders data forms based on a user's member selection, displaying only the specified entity. For example, a user variable named Department displays specific departments and employees.

user-defined attribute (UDA) An attribute, associated with members of an outline to describe a characteristic of the members, that can be used to return lists of members that have the specified associated UDA.

user-defined member list A named, static set of members within a dimension defined by the user.

validation The process of checking a business rule, report script, or partition definition against the outline to ensure that the object being checked is valid.

validation rules Rules used in FDM to enforce data integrity. For example, in FDM, validation rules ensure that certain conditions are met after data is loaded from FDM to the target application.

value dimension A dimension that is used to define input value, translated value, and consolidation detail.

variance The difference between two values (for example, between planned and actual values).

version A possible outcome used within the context of a scenario of data. For example, Budget - Best Case and Budget - Worst Case where Budget is scenario and Best Case and Worst Case are versions.

vertical application server cluster A cluster with multiple application server instances on the same machine.

view A year-to-date or periodic display of data.

visual cue A formatted style, such as a font or a color, that highlights specific data value types. Data values may be dimension members; parent, child, or shared members; dynamic calculations; members containing a formula; readonly data cells; read-and-write data cells; or linked objects.

WebLogic Server home A subdirectory of Middleware home containing installed files required by a WebLogic Server instance. WebLogic Server home is a peer of Oracle homes.

weight A value assigned to an item on a scorecard that indicates the relative importance of that item in the calculation of the overall scorecard score. The weighting of all items on a scorecard accumulates to 100%. For example, to recognize the importance of developing new features for a product, the measure for New Features Coded on a developer's scorecard would be assigned a higher weighting than a measure for Number of Minor Defect Fixes.

wild card Character that represents any single character (?) or group of characters (*) in a search string.

WITH section In MaxL DML, an optional section of the query used for creating reusable logic to define sets or members. Sets or custom members can be defined once in the WITH section and then referenced multiple times during a query.

workbook An entire spreadsheet file with many worksheets.

workflow The steps required to process data from start to finish in FDM. The workflow consists of Import (loading data from the GL file), Validate (ensures that all members are mapped to a valid account), Export (loads the mapped members to the target application), and Check (verifies accuracy of data by processing data with user-defined validation rules).

Workspace Page A page created with content from multiple sources including documents, URL, and other content types. Enables a user to aggregate content from Oracle and non-Oracle sources.

write-back The ability for a retrieval client, such as a spreadsheet, to update a database value.

ws.conf A configuration file for Windows platforms.

wsconf_platform A configuration file for UNIX platforms.

XML See Extensible Markup Language.

XOLAP An Essbase multidimensional database that stores only the outline metadata and retrieves all data from a relational database at query time. XOLAP supports aggregate storage databases and applications that contain duplicate member names.

Y axis scale A range of values on Y axis of charts displayed in Investigate Section. For example, use a unique Y axis scale for each chart, the same Y axis scale for all Detail charts, or the same Y axis scale for all charts in the column. Often, using a common Y axis improves your ability to compare charts at a glance.

Zero Administration A software tool that identifies version number of the most up-to-date plug-in on the server.

ZoomChart A tool for viewing detailed information by enlarging a chart. A ZoomChart enables you to see detailed numeric information on the metric that is displayed in the chart.