Oracle® Financial Services Enterprise Financial Performance Analytics

User Guide

Release 8

Part No. E57807-01

September 2015

Oracle Financial Services Enterprise Financial Performance Analytics User Guide, Release 8

Part No. E57807-01

Primary Author: Surag Ramachandran, Jyothi G., Asma Anjum

Contributor: Tim Braman, Prashant Agrawal, Abhilash Sugunan, Padmini Krishnan

Contents

Send Us Your Comments

Preface

1 Introduction

2 Overview of the Process Flow

3 Dimension Loading Process

4 Time Dimension Population

5 Modification and Mapping of Reporting Lines

6 Multi-Currency Reporting

7 Fact Ledger Population

8 Fact Management Reporting Population

9 What-if Analysis

10 Cube Build Process

11 Predictive Modelling

12 Overview of OFSEFPA Reports

A How to Define a Dimension

B How to Define a Measure

C How to Develop a New Cube

D How to Define a Batch

Send Us Your Comments

Oracle Financial Services Enterprise Financial Performance Analytics User Guide, Release 8

Part No. E57807-01

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Send your comments to us using the electronic mail address: financialservices_ww@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Intended Audience

Welcome to Release 8 of the *Oracle Financial Services Enterprise Financial Performance Analytics User Guide.*

This product is intended for Business Analysts who support Financial Reporting and Management Reporting.

See Related Information Sources on page x for more Oracle product information.

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.

Structure

1 Introduction

2 Overview of the Process Flow

3 Dimension Loading Process

4 Time Dimension Population

Business data commonly represents information as of a point in time (for example, a balance as of a point in time) or as of a particular span of time (for example, income for the month of March). The rollup of a particular balance depending on their nature could be a simple additive rollup wherein the child member balances are added up to arrive at the parent node balance (for example, Ending Balance) or non additive rollups wherein a node formula is used to specify how to rollup the child member balances (for example, 3 month rolling average).

5 Modification and Mapping of Reporting Lines

6 Multi-Currency Reporting

7 Fact Ledger Population

Fact Ledger population involves populating the FCT_LEDGER_STAT table from the LEDGER_STAT table.

8 Fact Management Reporting Population

Fact Management Reporting Population involves populating the FCT_MGMT_REPORTING table from the LEDGER_STAT table.

9 What-if Analysis

10 Cube Build Process

11 Predictive Modelling

12 Overview of OFSEFPA Reports

A How to Define a Dimension

B How to Define a Measure

C How to Develop a New Cube

D How to Define a Batch

Related Information Sources

- Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) User Guide

- Oracle Financial Services Analytical Applications Data Model Document Generation White paper

- Oracle Financial Services Profitability Management (OFSPM) User Guide

1

Introduction

Overview of Oracle Financial Services Enterprise Financial Performance Analytics (OFSEFPA)

Oracle Financial Services Enterprise Financial Performance Analytics (OFSEFPA) (previously, Oracle Financial Services Profitability Analytics (OFSPA)) is a complete end-to-end web-based Business Intelligence solution for generating and analyzing Financial and Management Reports. It provides tools for data integration and includes customizable, pre-built dashboards and reports, a reporting data model, and user friendly functional subject areas for ad-hoc reporting.

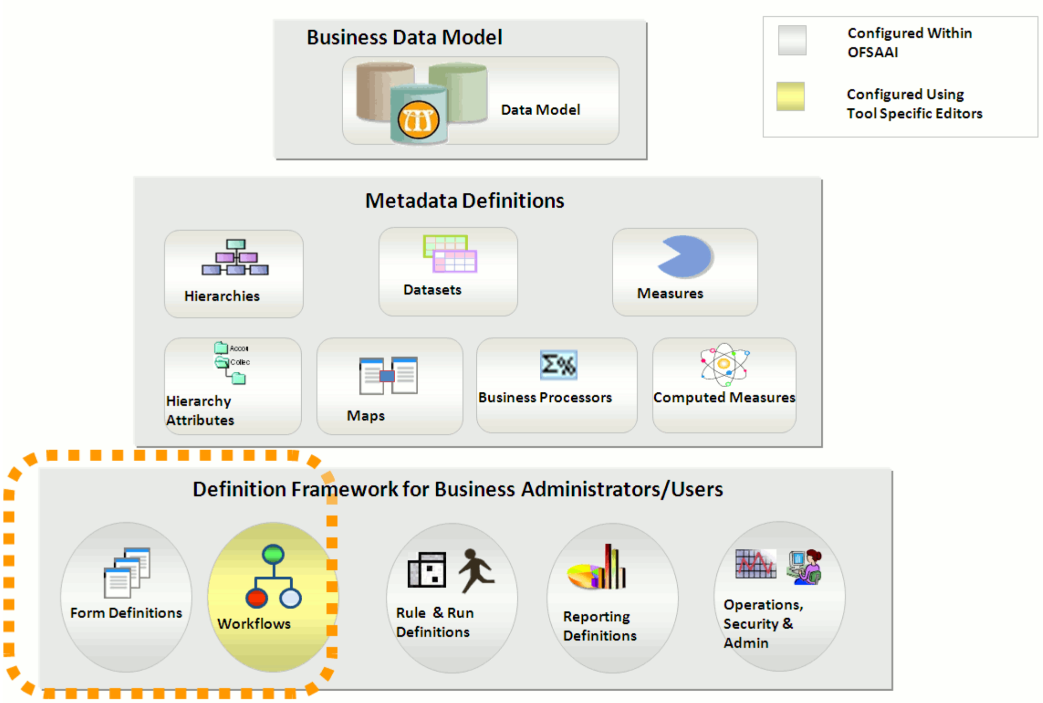
The OFSEFPA 8.0.0.0.0 is built using:

- OFSAAI 8.0.0.0.0 for ETL, Data Integration, and Cube Build activities.
- OBIEE 11.1.1.7.1 for Dashboard and Report activities.
- Essbase 11.1.2.1 for multi-dimensional Cube storage.

OFSEFPA 8.0.0.0.0 supports generating reports from both Relational Database and/or Essbase. Hence, Essbase is no longer a prerequisite for OFSEFPA.

This manual deals with OFSAAI, required for OFSEFPA activities, process, and functional details about the dash boards and reports. Also it includes subject areas which could be used for ad-hoc reporting using OBIEE Answers tool.

OFSEFPA Configurability in OFSAAI



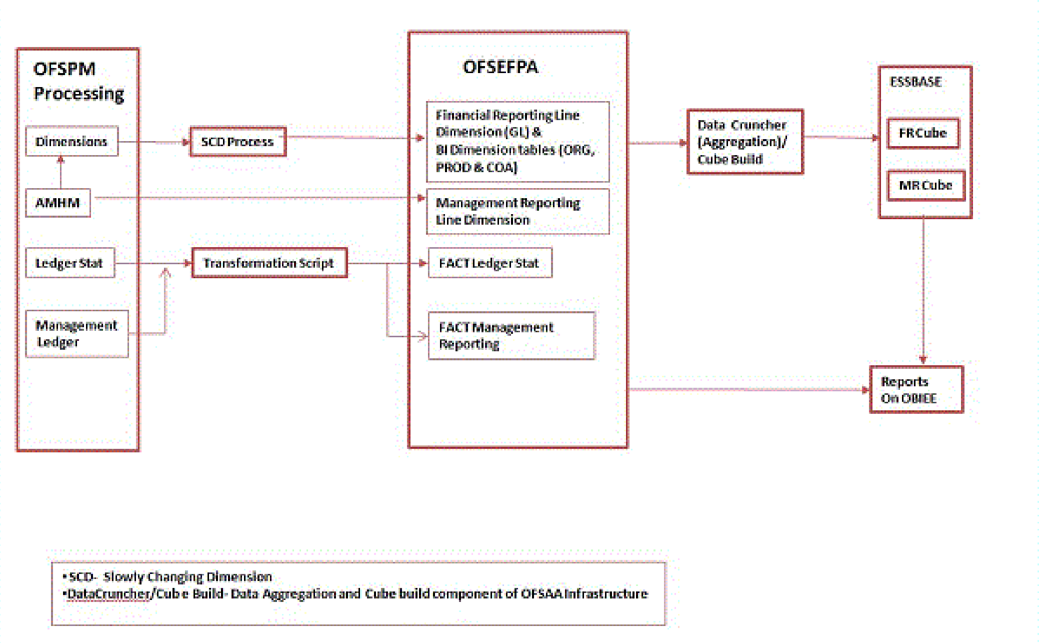
2

Overview of the Process Flow

Introduction

OFSEFPA 8.0.0.0.0 utilizes OBIEE technology to present financial reporting and management reporting. For details on OFSEFPA reports and how OBIEE is being utilized, refer to chapter Overview of OFSEFPA Reports, page 12-1. The OFSEFPA data model has been designed in such a way that it can receive fact and dimension data from Enterprise Performance Management (EPM) data model, which is installed through Oracle Financial Services Profitability Management 8.0.0.0.0 (OFSPM) seeded data flow processes or from other systems. The seeded data flow processes from the processing area of OFSPM to the data model of OFSEFPA utilizes the transformation and load components of OFSAAI 8.0.0.0.0. OFSEFPA 8.0.0.0.0 can be independently licensed and installed to work on top of the OFSAAI 8.0.0.0.0 infrastructure or can be licensed along with OFSPM 8.0 or higher to work in an integrated manner. The following diagram depicts the high-level data flow of OFSEFPA, when both OFSPM and OFSEFPA are installed.

OFSEFPA Data Flow


OFSPM Processing
Dimensions
AMHM
Ledger Stat
Management Ledger
SCD Process
Transformation Script
OFSEFPA
Financial Reporting Line Dimension (GL) & BI Dimension tables (ORG, PROD & COA)
Management Reporting Line Dimension
FACT Ledger Stat
FACT Management Reporting
Data Cruncher (Aggregation)/ Cube Build
ESSBASE
FR Cube
MR Cube
Reports On OBIEE
•SCD- Slowly Changing Dimension
•DataCruncher/Cub e Build- Data Aggregation and Cube build component of OFSAA Infrastructure

Data Flow: OFSPM Processing Area to OFSEFPA Reporting Area

The OFSPM processing area holds dimension data and fact data. Fact data is held primarily in the management ledger table (referred to as Ledger Stat in this document). The seeded data flow processes outlined in this section are for the management ledger table. Similarly the seeded dimension data movement process definitions are for the standard processing dimensions which come with the OFSPM Attributes Members Hierarchies Module (AMHM).

For more details on:

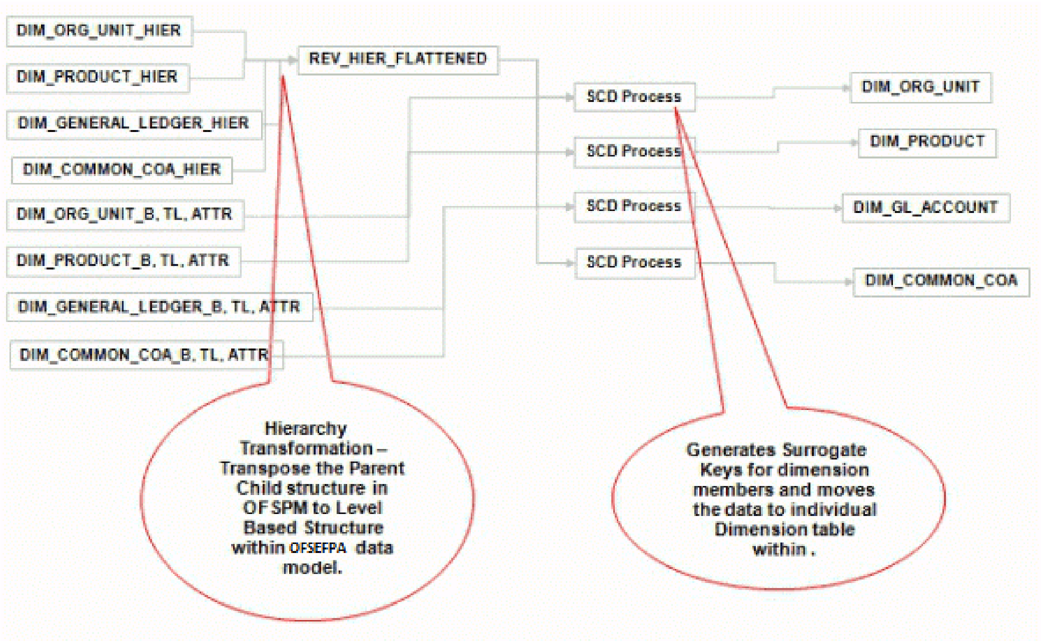
- Dimension data movement, refer to chapters Dimension Loading Process, page 3-1 and Time Dimension Population, page 4-1.
- Ledger Stat transformation, refer to chapter Fact Ledger Population, page 7-1.
- Fact Management Reporting Population, page 8-1.

Similar data movement process definitions can be created for any additional user-defined dimensions or custom instrument tables. These extensions are also described in this document.

The data movements from the OFSPM processing area to the OFSEFPA reporting area utilize the data transfer component of OFSAAI 8.0.0.0.0. Data transformation and loading is done with the Data Integrator module and is metadata driven. For more information on the Usage of the Data Integrator Component, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Dimension Data Flow

The Dimension data used for processing is stored in individual Member, Attribute, and Hierarchy tables for each dimension, and is part of the OFSPM AMHM data model. Hierarchies in OFSPM AMHM have a parent-child storage structure. These are flattened to a level-based structure as part of the data movement process. The flattened Hierarchies for each Dimension along with the Member and Attribute data undergo Slowly Changing Dimension (SCD) process to facilitate the movement of data to OFSEFPA dimension tables. The following diagram depicts the process flow:

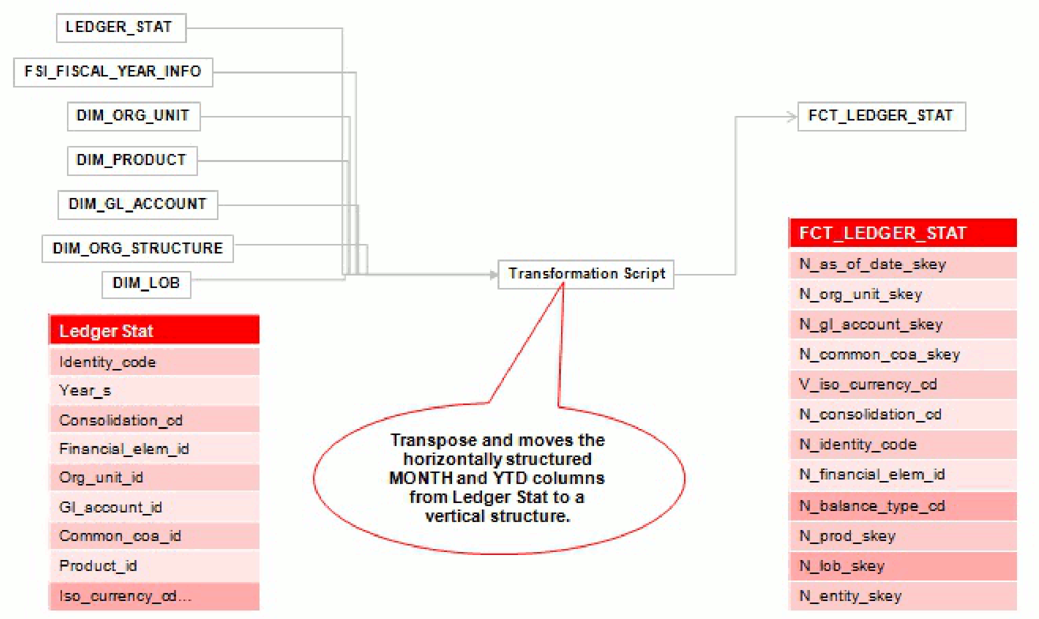


DIM_ORG_UNIT_HIER
DIM_PRODUCT_HIER
DIM_GENERAL_LEDGER_HIER
DIM_COMMON_COA_HIER
DIM_ORG_UNIT_B, TL, ATTR
DIM_PRODUCT_B, TL, ATTR
DIM_GENERAL_LEDGER_B, TL, ATTR
DIM_COMMON_COA_B, TL, ATTR
REV_HIER_FLATTENED
SCD Process
SCD Process
SCD Process
SCD Process
DIM_ORG_UNIT
DIM_PRODUCT
DIM_GL_ACCOUNT
DIM_COMMON_COA
Hierarchy Transformation – Transpose the Parent Child structure in OF SPM to Level Based Structure within OFSEFPA data model.
Generates Surrogate Keys for dimension members and moves the data to individual Dimension table within .

Note: For more information on dimension management, refer to *Oracle Financial Services Profitability Management (OFSPM) User Guide.*

Ledger Stat Data flow

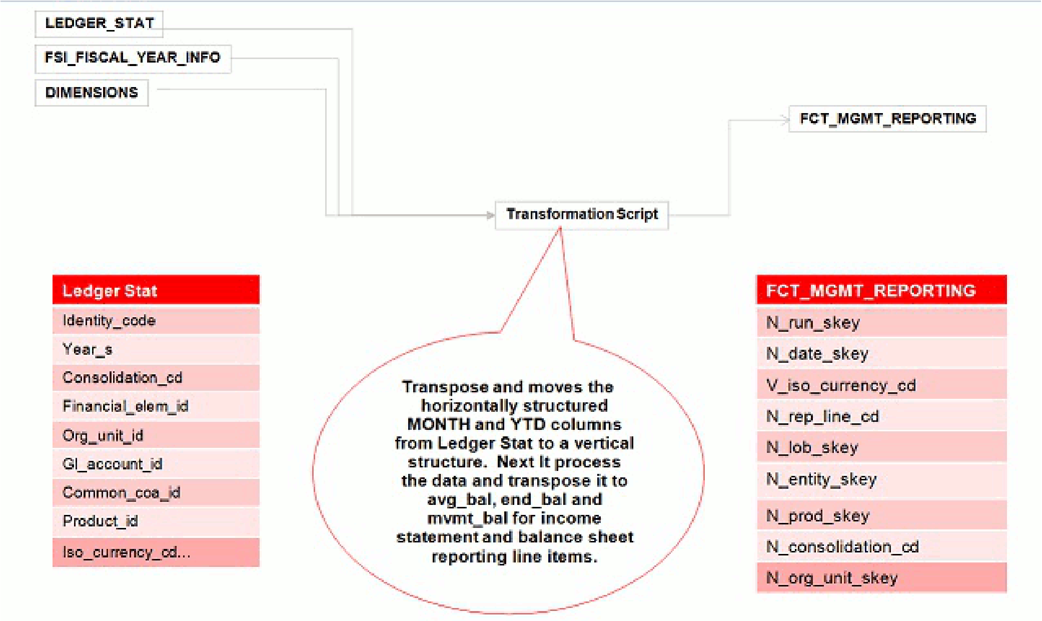
The ledger transformation program moves data from Ledger Stat in the OFSPM processing area to Fact Ledger Stat Entity of OFSEFPA. The transformation program joins data for all the required dimensions. The following diagram provides a high-level overview of this process. For more details on the process and its execution, refer to chapter Fact Ledger Population, page 7-1.



Optionally Data can be loaded from FSI_D_MANAGEMENT_LEDGER to FCT_MGMT_REPORTING.

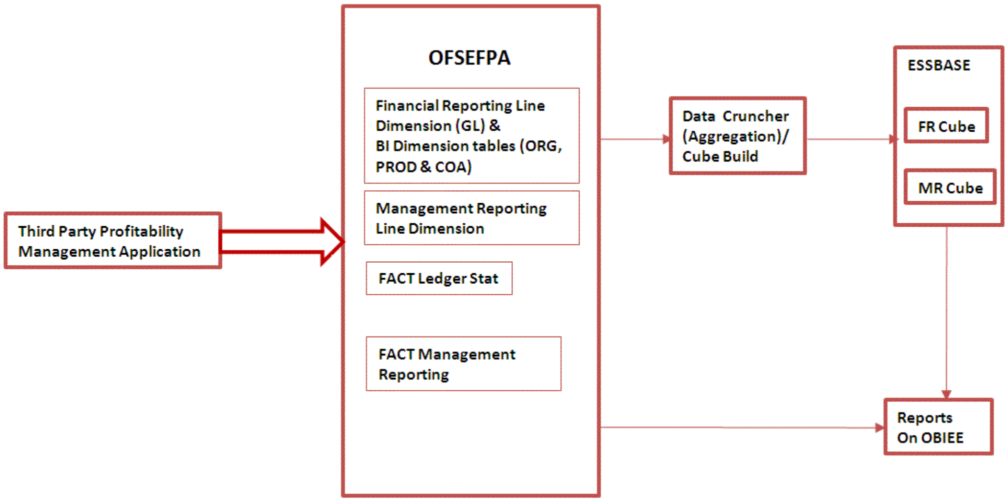
Management Reporting Data Flow

The management transformation program moves data from the Ledger Stat in OFSPM processing area to Fact Management Reporting Entity of OFSEFPA. The transformation program joins data for all the required dimensions. The following diagram provides a high-level overview of this process. For more details on the process and its execution, refer to chapter Fact Management Reporting Population, page 8-1.



Optionally Data can be loaded from FSI_D_MANAGEMENT_LEDGER to FCT_LEDGER_STAT

OFSEFPA Data Flow: Third Party Profitability Management Applications to OFSEFPA Reporting Area

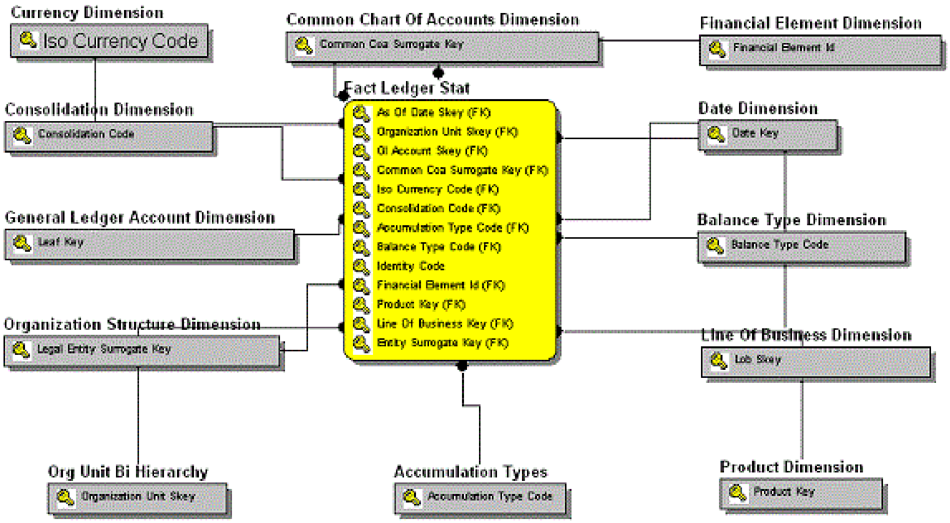

OFSEFPA
Financial Reporting Line Dimension (GL) & BI Dimension tables (ORG, PROD & COA)
Management Reporting Line Dimension
FACT Ledger Stat
FACT Management Reporting
Third Party Profitability Management Application
Data Cruncher (Aggregation)/ Cube Build
ESSBASE
FR Cube
MR Cube
Reports On OBIEE

Data Model

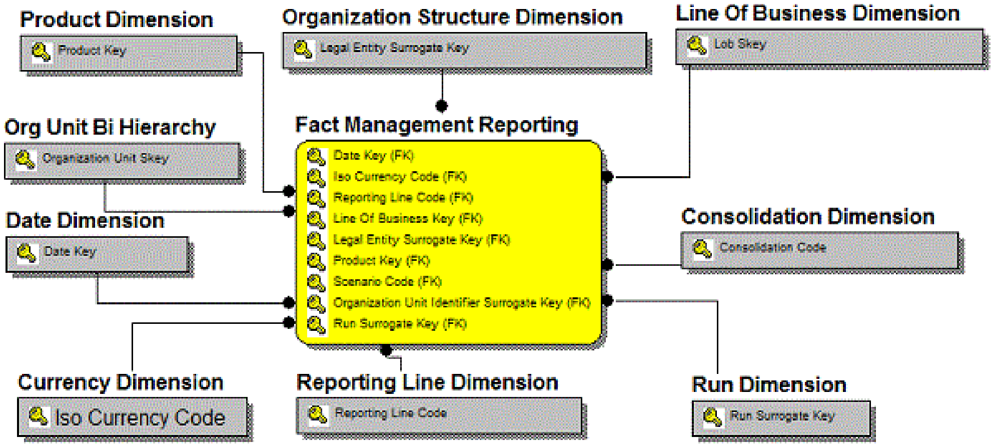
The data model of OFSEFPA is a star schema for the fact tables FCT_LEDGER_STAT and FCT_MGMT_REPORTING.

The Entity diagrams of the data model are provided below:

- Fact Ledger Stat



- Fact Management Reporting



Hierarchy

Standard dimensions such as Product, Organization, Common COA, and General Ledger are populated through the AMHM layer of Profitability Management application (OFSPM). OFSEFPA assumes that only one hierarchy is present for each dimension in the AMHM layer. This Hierarchy data for each Dimension is populated in the Dimension Tables through SCDs.

Entry in SETUP_MASTER table is required for the hierarchy for which the members are to be loaded in the dimension tables. One properly constructed Hierarchy ID should be given in the COMPONENT_VALUE column for each dimension.

3

Dimension Loading Process

Overview of Dimension Loading Process

The hierarchy data is natively stored in a parent-child structure within the *Dimension Management* component of OFSAAI. Dimension population involves the movement of dimension data from processing dimension tables (maintained by dimension management component of OFSAAI) to the reporting dimension tables used in Business Intelligence (BI) applications.

This data movement process is applicable only when OFSEFPA is installed along with OFSPM application.

Dimension loading process has the following two components:

1. Hierarchy Transformation, page 3-2
2. Dimension Tables Population, page 3-8

The Dimension loading process for the different hierarchies is discussed in the following sections:

- General Ledger:

  The members of this hierarchy are stored within the infrastructure metadata tables for OFSEFPA. Hence the hierarchy needs to be maintained separately for OFSPM and OFSEFPA. This hierarchy need not be included as part of the hierarchy transformation process but should be part of the SCD process to move the General Ledger (GL) dimension members from OFSPM to OFSEFPA.

- Financial Element:

  The members of this hierarchy are inserted through RDBMS insert scripts by the OFSEFPA solution installer. This hierarchy has to be maintained separately for OFSPM and OFSEFPA.

- Organization Unit:

This hierarchy in OFSPM first gets flattened by the Hierarchy Transformation and is then moved to the hierarchy table for Organization Unit (DIM_ORG_UNIT) by the Dimension table population component (as explained in chapter Dimension Tables Population, page 3-8). The OFSPM and OFSEFPA hierarchies can be kept in sync using the above two components.

- Product:

  This hierarchy in OFSPM first gets flattened through Hierarchy Transformation and is then moved to the hierarchy table for Product (DIM_PRODUCT) by the Dimension table population component (as explained in chapter Dimension Tables Population, page 3-8). The OFSPM and OFSEFPA hierarchies can be kept in sync by using the above two components.

- Time:

  The hierarchy table (DIM_DATES) for this hierarchy is loaded by the Time dimension population process (for more details, refer to chapter Time Dimension Population, page 4-1).

- Consolidation and Currency:

  The hierarchy data for these hierarchies are loaded through RDBMS insert scripts by the OFSEFPA solution installer. These hierarchies have to be maintained separately for OFSPM and OFSEFPA.

- Legal Entity:

  The hierarchy data of Legal Entity is loaded through insert scripts based on the user defined hierarchy.

The above components in detail and the execution methods are explained in the following sections.

Hierarchy Transformation

The following topics are covered in this section:

- Overview of Hierarchy Flattening Process, page 3-3
- Prerequisites, page 3-3
- Tables Used by the Hierarchy Flattening Transformation , page 3-5
- Executing the Hierarchy Flattening Transformation, page 3-5
- Checking the Execution Status, page 3-7

Overview of Hierarchy Flattening Process

Hierarchy Flattening Transformation is used to move the hierarchy data from the parent child storage structure in OFSPM AMHM model to a level based storage structure in OFSEFPA. In OFSPM AMHM model, hierarchy data for any hierarchy created on seeded or user defined dimensions using the AMHM is stored within hierarchy tables of respective dimensions. These are moved to the REV_HIER_FLATTENED table of OFSEFPA after flattening by the Hierarchy flattening process.

Example

The hierarchy data of one or more Product Hierarchies created on Product dimension (a seeded dimension) are stored in DIM_PRODUCTS_HIER table.

The hierarchy data in the preceding example would be moved to REV_HIER_FLATTENED in the OFSEFPA model by the Hierarchy Flattening Process.

Database components used by this transformation are:

1. REV_BATCHHIERFLATTEN - Oracle database function

2. REV_HIER_TRANSFORMATON_BIAPPS - Oracle database Package called by the preceding function.

Some of the features of the Hierarchy Flattening Transformation are:

- The user has the choice to process a single hierarchy or all hierarchies belonging to a particular dimension as part of a single execution.

- Any changes made in the hierarchy using the *AMHM Hierarchy Maintenance* screen will change the FLATTENED_ROWS_COMPLETION_CODE flag in REV_HIER_DEFINITIONS table to 'Pending'. This improves the processing efficiency, since the Transformation process will avoid hierarchies that have not been modified.

Prerequisites

- All the post install steps mentioned in the *Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration guide* and the *Solution Installation Manuals* of *Profitability Management* (only if OFSPM is installed) and *Enterprise Financial Performance Analytics* have to be completed successfully.

- Seeded Hierarchies which come with the install and any hierarchy created using the OFSAAI framework should have proper data in the Tables used by the Hierarchy Flattening Transformation, page 3-5. Hierarchy is maintained in the Dimension Management component of OFSAAI. (**Financial Services Application >Master Maintenance > Dimension Management > Hierarchies** screen).

> **Note:** The following debugging steps need to be performed only if the hierarchy flattening process has failed.

1. Check in the database (atomic schema) if the FLATTENED_ROWS_COMPLETION_CODE column of the REV_HIER_DEFINITIONS table has value 'Pending' for the Hierarchy ID to be processed.

   This column will have the value 'Pending' for any new hierarchy created or modified using the OFSAAI Hierarchy Management User Interface.

2. Check if the REV_DIMENSIONS_B table has a row for the dimension that is being processed.

   Execute the following query in the database to find the value and use the value in the dimension ID column for the dimension name/description to be processed.

   ```
   Select b.dimension_id,t.dimension_name,t.description from
   rev_dimensions_b b inner join rev_dimensions_tl t on
   b.dimension_id = t.dimension_id and t.dimension_name like
   '<dimension name>'
   ```

3. Check if the REV_HIERARCHIES table has a row for the hierarchy id that is being processed.

   ```
   SELECT * FROM rev_hierarchies rh where dimension_id = <dimension
   id>
   ```

- Map the application user to BATPRO role, that has seeded batch execution function.

- Create a Batch. For more information refer to Executing the Hierarchy Flattening Transformation, page 3-5.

- Before executing a batch check if the following services are running on the application server.

  - Iccserver

  - Router

  - AM Server

  - Messageserver

  - Olapdataserver

> **Note:** For more information on how to check if the services are up

and on, and how to start the services if you find them not running, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Tables Used by the Hierarchy Flattening Transformation

- REV_HIERARCHIES - This is the master table for hierarchies with one row per hierarchy.
- REV_DIMENSIONS_B - This is the master table for dimensions with one row per dimension.
- REV_HIER_DEFINITIONS - FLATTENED_ROWS_COMPLETION_CODE column is checked to determine whether the hierarchy is to be processed.
- DIM_<DIMENSIONNAME>_HIER - This table stores the hierarchy data and is the source for the transformation.

  Example
  DIM_PRODUCTS_HIER - This table stores the hierarchy data of one or more Product Hierarchies created on Product dimension (a seeded dimension).
- REV_HIER_FLATTENED - This is the output table for the transformation into which the flattened hierarchy data gets populated.

Executing the Hierarchy Flattening Transformation

You can execute the Data Transformation from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI.

The Hierarchy Flattening Transformation for OFSEFPA 6.1 has been seeded with the Batch ID **<INFODOM>_Hierarchy_Transformation**, which can be executed from the *Batch Execution* section of OFSAAI. In the Parameter List, Enter Dimension ID and Hierarchy ID. For example, '2', '1000003710'.

Queries to obtain the Dimension ID and Hierarchy ID are provided below.

You can also define a new Batch and an underlying Task definition from the *Batch Maintenance* window of OFSAAI. For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

To define a new task for a Batch definition:

- Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.
- Click **Add** (+) button from the Task Details grid.

The *Task Definition* window is displayed.

- Enter the **Task ID** and **Description**.
- Select **TRANSFORM DATA** component from the drop down list.
- Select the following from the **Dynamic Parameters** list:
  - **Datastore Type** - Select the appropriate datastore type from the list
  - **Datastore Name** - Select the appropriate datastore name from the list
  - **IP address** - Select the IP address from the list
  - Rule **Name** - Select **BATCH_HIERTRANSFORMATION** from the drop down list of available transformations. (This is a seeded Data Transformation which is installed as part of the OFSEFPA Solution Installer. If you don't see this in the list, contact Oracle Support.)
  - **Parameter List** - Enter Dimension ID and Hierarchy ID.

    For the Parameter List, the values are:

    - **Dimension ID –** Execute the following query in the database to find the value, and use the value in the Dimension ID column for the dimension name/description to be processed.

      ```
      Select b.dimension_id,t.dimension_name,t.description from
      rev_dimensions_b b inner join rev_dimensions_tl t on
      b.dimension_id = t.dimension_id and t.dimension_name like
      '<dimension name>'
      ```

      Replace <dimension name> in the preceding query with the Dimension Name you find in the UI (**Financial Services Application > Master Maintenance > Dimension Management**) for the dimension on which the Hierarchy you want to flatten.

    - **Hierarchy ID** - If all the hierarchies belonging to a dimension are to be processed, then provide **null** as the parameter value. Else, provide the System Identifier of the hierarchy that needs to be transformed.

      Execute the following query in the database, only if a single hierarchy has to be processed, and use the value in HIERARCHY_ID column as parameter for the hierarchy to be processed.

      ```
      select b.object_definition_id,short_desc,long_desc from
      fsi_m_object_definition_b b inner join
      fsi_m_object_definition_tl t on b.object_definition_id =
      t.object_definition_id and b.id_type = 5
      ```

      For OFSEFPA, it is assumed that only one Hierarchy is processed at a time.

Example
If all the hierarchies for GL Account dimension must be processed, the parameter list should be given as:

'2',null

Where '2' is the Dimension ID for the seeded dimension GL Account.

Example
If a particular hierarchy with code 1000018112 must be processed, the parameter list should be given as follows:

'2', '1000018112'

Where '1000018112' is the code obtained by executing the preceding query in the database.

- Click **Save**.

  The Task definition is saved on the selected Batch.

You can execute the batch from *Batch Execution* window by choosing the Batch created following the steps mentioned in the preceding sections. For more details, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Hierarchy Transformation can also be executed directly on the database through SQLPLUS. The details are:

- **Function Name**: REV_BATCHHIERFLATTEN
- **Parameters**: BATCH_RUN_ID, MIS_DATE, PDIMENSIONID, and PHIERARCHYID
- **Sample Parameter Values:** 'Batch1', '20091231', '2', '1000018112'

> **Note:** Execute the Hierarchy Transformation Batch only when a new Hierarchy is defined or an existing Hierarchy is modified.

Checking the Execution Status

The Batch execution status can be monitored through *Batch Monitor* section of *OFSAAI Operations* module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S – Success

The Event Log window in Batch Monitor section provides execution logs, in which the top row is the most recent. Any errors during the Batch execution are listed in the logs.

> **Tip:** It is advisable to check the Event Log for any errors, even if the execution status returns 'Success'.

The execution log can also be accessed on the application server in the directory *$FIC_DB_HOME/log/date*, where file name will have the Batch Execution ID.

> **Note:** Check the **.profile** file in the installation home if you are unable to find this path.

The database level operations log can be accessed by querying the FSI_MESSAGE_LOG table. The Batch Run ID column can be filtered for identifying the relevant log. (This is the same log you see in the Event Log window.)

Dimension Tables Population

Dimensional data changes are handled by OFSEFPA solution using the SCD component.

The following topics are covered in this section:

- Overview of SCD Process , page 3-8
- Prerequisites, page 3-10
- Tables Used by the SCD Component , page 3-11
- Executing the SCD Component, page 3-16
- Checking the Execution Status, page 3-18

Overview of SCD Process

SCDs are dimensions that have data that changes slowly, rather than changing on a time-based, regular schedule.

For more information on SCDs, refer to:

- *Oracle Data Integrator Best Practices for a Data Warehouse* at

  <http://www.oracle.com/technetwork/middleware/data-integrator/overview/odi-bestpractices-datawarehouse-whi-129686.pdf >

- *Oracle Warehouse Builder Data Modeling, ETL, and Data Quality Guide* at

  <http://download.oracle.com/docs/cd/E16338_01/owb.112/e10935/dim_objects.htm>

Additional online sources include:

- <http://en.wikipedia.org/wiki/Slowly_changing_dimension>
- <http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/10g/r2/owb/owb10gr2_gs/owb/lesson3/slowlychangingdimensions.htm>
- <http://www.oraclebidwh.com/2008/11/slowly-changing-dimension-scd/>
- <http://www.informationweek.com/news/software/bi/showArticle.jhtml?articleID=204800027&pgno=1>
- <http://www.informationweek.com/news/software/bi/showArticle.jhtml?articleID=59301280>

You can also refer to *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling* by Ralph Kimball and Margy Ross.

The SCD component of the platform is delivered via a C++ executable. The types of SCD handled by the OFSAAI SCD component for OFSEFPA solution are Type 1 and Type 2.

Type 1 Methodology

The Type 1 methodology overwrites old data with new data, and therefore does not track historical data. This is useful for making changes to dimension data.

Example

N_PRODUCT_SKEY	V_PRODUCT_NAME	D_START_DATE	D_END_DATE	F_LATEST_RECORD_INDICATOR
1	PL	5/31/2010	12/31/9999	Y

In this example,

N_PRODUCT_SKEY is the surrogate key column which is a unique key for each record in the dimension table.

V_PRODUCT_NAME is the product name.

D_START_DATE indicates the date from which this product record is valid.

D_END_DATE indicates the date till which this product record is valid.

F_LATEST_RECORD_INDICATOR with value 'Y', which indicates this is the latest record in the dimension table for this product and 'N' indicates it is not.

If the V_PRODUCT_NAME column is set as a Type 1 SCD column and if there is a change in the product name to 'Personal Loan' from 'PL' in the above example, in the next processing period, then when SCD is executed for the new processing period the record in the above example changes to:

N_PRODUCT_SKEY	V_PRODUCT_NAME	D_START_DATE	D_END_DATE	F_LATEST_RECORD_INDICATOR
1	Personal Loan	6/30/2010	12/31/9999	Y

Type 2 Methodology

The Type 2 method tracks historical data by creating multiple records for a given natural key in the dimensional tables with separate surrogate keys. With Type 2, the historical changes in dimensional data are preserved. In the above example for the change in product name from 'PL' to 'Personal Loan' if history has to be preserved, then the V_PRODUCT_NAME column has to be set as Type 2 when SCD is processed for the processing period and the change inserts a new record as shown in the following example:

Example

N_PRODUCT_SKEY	V_PRODUCT_NAME	D_START_DATE	D_END_DATE	F_LATEST_RECORD_INDICATOR
1	PL	5/31/2010	12/31/9999	N
1	Personal Loan	6/30/2010	12/31/9999	Y

A new record is inserted to the product dimension table with the new product name. The latest record indicator for this is set as 'Y', indicating this is the latest record for the personal loan product. The same flag for the earlier record was set to 'N'.

Prerequisites

1. The Hierarchy Flattening Transformation should have been executed successfully.

2. The SCD executable should be present under <*installation home*>*ficdb/bin*. The file name is **scd** and the user executing the SCD component should have execute rights on this file.

3. The setup tables accessed by SCD component (SETUP_MASTER, SYS_TBL_MASTER, and SYS_STG_JOIN_MASTER) should have the required entries. The SETUP_MASTER table does not come seeded with the installation; the required entries must be added manually. The required columns are mentioned in the Tables Used by the SCD Component, page 3-11. The tables SYS_TBL_MASTER and SYS_STG_JOIN_MASTER are seeded for the Org unit, GL Account, Product, Common COA (Chart of Accounts) dimensions along with solution installation and you must only add entries in these tables, if you add new dimensions.

4. Database Views with name DIM_<Dimension Name>_V come seeded, for the seeded dimensions which come as part of installation. These views source data from the Profitability dimension tables as well as the flattened hierarchy data.

   DIM_PRODUCT_V is the view available for the product dimension.

   New views will have to be added for any new dimension, added in addition to the seeded dimensions.

Tables Used by the SCD Component

The following are the database tables and columns used by the SCD component:

- SETUP_MASTER
  - V_COMPONENT_CODE - This column is not used by the OFSEFPA solution.
  - V_COMPONENT_DESC - This column value is hard coded in the database view definitions for DIM_PRODUCT_V, DIM_GL_ACCOUNT_V, DIM_COMMON_COA_V, and DIM_ORG_UNIT_V to obtain the Hierarchy ID from the REV_HIER_FLATTENED table. For this reason, the value for this column should be unique.

    > **Note:** The value in V_COMPONENT_DESC must exactly match with the value used in the SQL to create the DIM_<dimension>_V view. The View SQL contains a section referencing the SETUP_MASTER table. You must use the same upper and/or lower case letters in V_COMPONENT_DESC as used in this section of the View SQL.

  - V_COMPONENT_VALUE - This is the hierarchy ID to be processed and this can be obtained by executing the following query:

    ```
    select b.object_definition_id,short_desc,long_desc from
    fsi_m_object_definition_b b inner join fsi_m_object_definition_tl
    t on b.object_definition_id = t.object_definition_id and
    b.id_type = 5
    ```

    Example:

V_COMPONENT_CODE	V_COMPONENT_DESC	V_COMPONENT_VALUE
COMMON_COA_HIER	COMMON_COA_HIER1	1000063952
GL_ACCOUNT_HIER	GL_ACCOUNT_HIER1	200000808

V_COMPONENT_CODE	V_COMPONENT_DESC	V_COMPONENT_VALUE
ORG_HIER	ORG_UNIT_HIER1	200282
PRODUCT_HIER	PRODUCT_HIER1	1000004330

> **Note:** *For any newly defined Hierarchy, a row will have to be inserted to this table manually for SCD to process that Hierarchy. You can only specify one Hierarchy for each dimension.*

- SYS_TBL_MASTER

  The solution installer populates one row per dimension for the seeded dimensions in this table.

Column Name	Data Type	Column Description
MAP_REF_NUM	NUMBER(3) NOT NULL	The Mapping Reference Number for this unique mapping of a Source to a Dimension Table.
TBL_NM	VARCHAR2(30) NOT NULL	Dimension Table Name.
STG_TBL_NM	VARCHAR2(30) NOT NULL	Staging Table Name.
SRC_PRTY	NUMBER(2) NULL	Priority of the Source when multiple sources are mapped to the same target.
SRC_PROC_SEQ	NUMBER(2) NOT NULL	The sequence in which the various sources for the DIMENSION will be taken up for processing.

Column Name	Data Type	Column Description
SRC_TYP	VARCHAR2(30) NULL	The type of the Source for a Dimension, that is, Transaction Or Master Source.
DT_OFFSET	NUMBER(2) NULL	The offset for calculating the Start Date based on the Functional Requirements Document (FRD).
SRC_KEY	NUMBER(3) NULL	

Example:

This is the row inserted by the solution installer for the product dimension.

MAP_REF_NUM	128
TBL_NM	DIM_PRODUCT
STG_TBL_NM	DIM_PRODUCT_V
SRC_PRTY	
SRC_PROC_SEQ	1
SRC_TYP	MASTER
DT_OFFSET	0

> **Note:** *For any newly defined dimension, a row will have to be inserted to this table manually.*

- SYS_STG_JOIN_MASTER

  The solution installer populates this table for the seeded dimensions.

Column Name	Data Type	Column Description
MAP_REF_NUM	NUMBER(3) NOT NULL	The Mapping Reference Number for this unique mapping of a Source to a Dimension Table.
COL_NM	VARCHAR2(30) NOT NULL	Name of the column in the Dimension Table.
COL_TYP	VARCHAR2(30) NOT NULL	Type of column. The possible values are given in the following section.
STG_COL_NM	VARCHAR2(60) NULL	Name of the column in the Staging Table.
SCD_TYP_ID	NUMBER(3) NULL	SCD type for the column.
PRTY_LOOKUP_REQD_FLG	CHAR(1) NULL	Column to determine whether Lookup is required for Priority of Source against the Source Key Column or not.
COL_DATATYPE	VARCHAR2(15) NULL	The list of possible values are VARCHAR, DATE, and NUMBER, based on the underlying column datatype.
COL_FORMAT	VARCHAR2(15) NULL	

The possible values for column type (the COL_TYPE column) in SYS_STG_JOIN_MASTER table are:

1. PK - Primary Dimension Value (can be the multiple of the given *Mapping Reference Number*)

2. SK - Surrogate Key

3. DA - Dimensional Attribute (may be multiple for a given "Mapping Reference Number")
4. SD - Start Date
5. ED - End Date
6. LRI - Latest Record Indicator (Current Flag)
7. CSK - Current Surrogate Key
8. PSK - Previous Surrogate Key
9. SS - Source Key
10. LUD - Last Updated Date/Time
11. LUB - Last Updated By
12. NN - Not Null

Example:

This is the row inserted by the solution installer for the product dimension.

MAP_REF_NUM	128
COL_NM	V_PRODUCT_NAME
COL_TYP	DA
STG_COL_NM	V_PRODUCT_NAME
SCD_TYP_ID	2
PRTY_LOOKUP_REQD_FLG	N
COL_DATATYPE	VARCHAR
COL_FORMAT	

> **Note:** *For any newly defined dimension, the column details will have to be inserted to this table manually.*

- DIM_< dimension name >_V - The database view which SCD uses as the source.

  Example
  DIM_PRODUCTS_V

  These views come as part of install for the dimensions seeded with the application.

  > **Note:** *For any newly defined dimension, a view will have to be created, which is similar to that of DIM_PRODUCTS_V.*

A sequence should be created for every user-defined dimension, using the below query:

Example

```
create sequence SEQ_< DIMENSION > minvalue 1
maxvalue 9999999999999999999999999999
increment by 1
```

Executing the SCD Component

You can execute the SCD component from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI.

The SCD component for OFSEFPA 6.1 has been seeded with the Batch ID **<INFODOM>_SCD**, which can be executed from *Batch Execution* section of OFSAAI.

You can also define a new Batch and an underlying Task definition from the *Batch Maintenance* window of OFSAAI. For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

To define a new task for a Batch definition:

- Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

- Click **Add** (+) button from the Task Details grid.

  The *Task Definition* window is displayed.

- Enter the **Task ID** and **Description**.

- Select **Run Executable** component from the drop down list.

- Select the following from the **Dynamic Parameters** list:

  - **Datastore Type** - Select the appropriate datastore type from the list

  - **Datastore Name** - Select the appropriate datastore name from the list

  - **Executable** - Enter **scd,<map ref num>**

    For example, **scd,2**

- **Wait** - Click **Yes** if you want to wait till the execution is complete or click **No** to proceed with the next task.

  > **Important:** Select **Yes** if you want the ICC component to wait for the process to complete the execution and update the status as either **Success** or **Failure**.
  >
  > If you select **No**, the component will trigger the processes and update the status as **Success**.

- **Batch Parameter** - Click **Yes** in Batch Parameter field if you want to pass the batch parameters to the executable and click **No** otherwise.

  > **Important:** Always select **Yes** in Batch Parameter.

- Click **Save**.

  The Task definition is saved for the selected Batch.

- Click **Parameters.** Select the following from the Dynamic Parameters List and then click **Save**:

  The map ref number values available for the **Executable** parameter are:

  - **-1**, if you want to process all the dimensions. The *Executable* parameter mentioned earlier is:

    scd,-1.

  - If you want to process for a single dimension, query the database table SYS_TBL_MASTER and give the number in the MAP_REF_NUM column for the dimension you want to process. These are the ones which come seeded with the install. If you want to process for Product dimension, the *Executable* parameter mentioned earlier is:

    scd,6.

MAP_REF_NUM	TBL_NM
126	DIM_ORG_UNIT
127	DIM_GL_ACCOUNT
128	DIM_PRODUCT

MAP_REF_NUM	TBL_NM
129	DIM_COMMON_COA

- You can execute a Batch definition from the *Batch Execution* section of *OFSAAI Operations* module.

Checking the Execution Status

The Batch execution status can be monitored through *Batch Monitor* section of *OFSAAI Operations* module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S – Success

The execution log can also be accessed on the application server in the directory *$FIC_DB_HOME/log/ficgen*, where file name will have the Batch Execution ID.

The detailed SCD component log can be accessed on the application server in the directory $FIC_HOME by accessing the following path */ftpshare/<infodom name>/logs*.

> **Note:** Check the **.profile** file in the installation home if you are unable to find this path.

The *Event Log* window in *Batch Monitor* section provides execution logs, in which the top row is the most recent. Any errors during the Batch execution are listed in the logs.

Legal Entity

It is manual upload. Optionally it can be done by using <INFODOM> _DIM_Org_Structure

Batch <INFODOM>DIM_Org_Structure triggers the DT and loads the DIM_Org_Structure Table

Parameter List: NULL

DT uses data from below tables:

1. STG_LEGAL_ENTITY_B_INTF
2. STG_LEGAL_ENTITY_TL_INTF

3. STG_LEGAL_ENTITY_ATTR_INTF

4. STG_LEGAL_ENTITY_HIER_INTF

Parent Child Hierarchy

OBIEE can handle Hierarchies in Parent - Child format as well. For this, the Level based Hierarchy of AMHM has to be converted to a Parent - Child Hierarchy to support Hierarchies in reports. Parent Child hierarchy is applicable for:

- Reporting Line (DIM_REP_LINE)
- General Ledger (DIM_GL_ACCOUNT)
- Organization Structure (DIM_ORG_STRUCTURE)

Once the above dimension tables are loaded, their respective Parent-Child Relation tables also have to be populated. The following are the dimension tables with their respective Parent-Child Relation tables:

- DIM_REP_LINE - REP_LINE_PARENT_CHILD_RELATION
- DIM_GL_ACCOUNT - GL_PARENT_CHILD_RELATION
- DIM_ORG_STRUCTURE - ORG_STR_PARENT_CHILD_RELATION

The following DTs populate the Parent-Child Relation tables:

1. FN_REP_LINE_PARENT_CHILD

   The database components used to populate REP_LINE_PARENT_CHILD_RELATION are:

   - Database function FN_REP_LINE_PARENT_CHILDWR
   - Database function FN_REP_LINE_PARENT_CHILD, which is called by the function FN_REP_LINE_PARENT_CHILDWR mentioned above.

   Batch **<INFODOM>_REPLINE_PARENT_CHILD_UPD** triggers the DT and loads the REP_LINE_PARENT_CHILD_RELATION table.

   Parameter List: Hierarchy Name and Folder Name

   Example: 'Repline Hierarchy','EPM61SEG'

   Below query retrieves the Hierarchy Name and the Folder Name

```
select distinct FODTL.SHORT_DESC, FODB.FOLDER_NAME
  from fsi_m_object_definition_tl FODTL, fsi_m_object_definition_b
FODB
 where FODTL.object_definition_id in
       (select object_definition_id
          from fsi_m_object_definition_b
         where table_name = 'DIM_REPORTING_LINE_HIER')
         and FOLDER_NAME = '<OFSAA segment name>';
```

There is a foreign key (FK) reference from table FCT_MGMT_REPORTING (N_REP_LINE_CD column) to table DIM_REP_LINE (N_REP_LINE_CD column). This foreign key, FK_FCT_MGMT_REPORTING_3 should be disabled if the Data Transformation (DT) FN_REP_LINE_PARENT_CHILD has to run again, as this DT deletes the DIM_REP_LINE table and reloads again. You can enable the FK after the successful execution of the DT.

2. FN_GL_PARENT_CHILD

   The database components used to populate GL_PARENT_CHILD_RELATION are:

   - Database function FN_GL_PARENT_CHILDWR

   - Database function FN_GL_PARENT_CHILD, which is called by the function FN_GL_PARENT_CHILDWR mentioned above.

   Batch **<INFODOM>_ GL_Parent_Child_UPD** triggers the DT and loads the GL_PARENT_CHILD_RELATION table.

   Parameter List: NULL

3. FN_ORG_PARENT_CHILD

   The database components used to populate ORG_STR_PARENT_CHILD_RELATION are:

   - Database function FN_ORG_PARENT_CHILDWR

   - Database function FN_ORG_PARENT_CHILD, which is called by the function FN_ORG_PARENT_CHILDWR mentioned above.

   Batch **<INFODOM>_ ORG_Parent_Child_UPD** triggers the DT and loads the ORG_STR_PARENT_CHILD_RELATION table.

   Parameter List: NULL

   > **Note:** Whenever there are changes to any of the three dimensions mentioned above, the corresponding Parent- Child relation DT should be executed. The OBIEE reports and the Parent- Child relation tables require the Parent ID to be null for the root node of the respective Hierarchy in the corresponding dimension table.

Multiple Hierarchies

The current flow of hierarchy data movement from AMHM tables to Reporting dimension tables considers the following objects:

- 4 AMHM tables (<Dimension>_B/TL/ATTR/HIER)
- REV_HIER_FLATTENED table
- SETUP_MASTER table
- Dimension specific view
- SCD to load data from View to Reporting Dimension table

This out-of-the-box product has been configured to support data movement of not only one hierarchy but also to support more than one hierarchy. The information mentioned specifies the points at high level that one has to consider in order to support more than one hierarchy.

The following steps describe the data movement in two hierarchies (Product dimension is considered as an example):

1. Create two hierarchies for Product dimension in AMHM (For example: 200183, 301741)

2. Run Hierarchy Flattening transformation to load flattened structure of both in REV_HIER_FLATTENED.

3. Load SETUP_MASTER table with two entries identified by V_COMPONENT_DESC values PRODUCT_HIER1 and PRODUCT_HIER2.

V_COMPONENT_CODE	V_COMPONENT_DESC	V_COMPONENT_VALUE
PRODUCT_HIER	PRODUCT_HIER1	301741
PRODUCT_HIER2	PRODUCT_HIER2	200183

4. Modify the view definition DIM_PRODUCT_V to include both hierarchies. Find below both the OOTB and the modified view definitions.

DIM_PRODUCT_V

CREATE OR REPLACE VIEW DIM_PRODUCT_V AS

SELECT C.LEVEL_01_CODE n_product_id_level01,

```
C.LEVEL_01_NAME v_product_name_level01,
 C.LEVEL_02_CODE n_product_id_level02,
C.LEVEL_02_NAME v_product_name_level02,
 C.LEVEL_03_CODE n_product_id_level03,
C.LEVEL_03_NAME v_product_name_level03,
C.LEVEL_04_CODE n_product_id_level04,
C.LEVEL_04_NAME v_product_name_level04,
C.LEVEL_05_CODE n_product_id_level05,
C.LEVEL_05_NAME v_product_name_level05,
C.LEVEL_06_CODE n_product_id_level06,
C.LEVEL_06_NAME v_product_name_level06,
C.LEVEL_07_CODE n_product_id_level07,
C.LEVEL_07_NAME v_product_name_level07,
C.LEVEL_08_CODE n_product_id_level08,
C.LEVEL_08_NAME v_product_name_level08,
C.LEVEL_09_CODE n_product_id_level09,
C.LEVEL_09_NAME v_product_name_level09,
C.LEVEL_10_CODE n_product_id_level10,
C.LEVEL_10_NAME v_product_name_level10,
C.LEVEL_11_CODE n_product_id_level11,
C.LEVEL_11_NAME v_product_name_level11,
C.LEVEL_12_CODE n_product_id_level12,
C.LEVEL_12_NAME v_product_name_level12,
C.LEVEL_13_CODE n_product_id_level13,
C.LEVEL_13_NAME v_product_name_level13,
C.LEVEL_14_CODE n_product_id_level14,
C.LEVEL_14_NAME v_product_name_level14,
C.LEVEL_15_CODE n_product_id_level15,
C.LEVEL_15_NAME v_product_name_level15,
C.LEVEL_16_CODE n_product_id_level16,
C.LEVEL_16_NAME v_product_name_level16,
C.LEVEL_17_CODE n_product_id_level17,
C.LEVEL_17_NAME v_product_name_level17,
C.LEVEL_18_CODE n_product_id_level18,
C.LEVEL_18_NAME v_product_name_level18,
C.LEVEL_19_CODE n_product_id_level19,
C.LEVEL_19_NAME v_product_name_level19,
NVL(C.LEVEL_20_CODE, A.PRODUCT_ID) n_product_id_level20,
NVL(C.LEVEL_20_NAME, D.PRODUCT_NAME) v_product_name_level20,
C.SEQ_01 N_SEQ_01,
C.SEQ_02 N_SEQ_02,
C.SEQ_03 N_SEQ_03,
C.SEQ_04 N_SEQ_04,
C.SEQ_05 N_SEQ_05,
C.SEQ_06 N_SEQ_06,
C.SEQ_07 N_SEQ_07,
C.SEQ_08 N_SEQ_08,
C.SEQ_09 N_SEQ_09,
C.SEQ_10 N_SEQ_10,
C.SEQ_11 N_SEQ_11,
C.SEQ_12 N_SEQ_12,
C.SEQ_13 N_SEQ_13,
C.SEQ_14 N_SEQ_14,
C.SEQ_15 N_SEQ_15,
C.SEQ_16 N_SEQ_16,
C.SEQ_17 N_SEQ_17,
C.SEQ_18 N_SEQ_18,
C.SEQ_19 N_SEQ_19,
C.SEQ_20 N_SEQ_20,
NVL((SELECT ACCOUNT_TYPE_CD
FROM FSI_ACCOUNT_TYPE_CD Z
WHERE ACCOUNT_TYPE_CD =
 (SELECT BB.DIM_ATTRIBUTE_NUMERIC_MEMBER
FROM DIM_PRODUCTS_ATTR AA, DIM_COMMON_COA_ATTR BB
WHERE A.PRODUCT_ID = AA.PRODUCT_ID
 AND AA.ATTRIBUTE_ID = '5011'
 AND AA.DIM_ATTRIBUTE_NUMERIC_MEMBER =
BB.COMMON_COA_ID
AND BB.ATTRIBUTE_ID = '5040'))
, 0) N_ACCOUNT_TYPE,
A.PRODUCT_ID N_PRODUCT_ID,
NVL(A.PRODUCT_CODE, A.PRODUCT_ID) V_PROD_CODE,
D.PRODUCT_NAME V_PRODUCT_NAME,
TO_DATE(TO_CHAR(A.CREATION_DATE, 'YYYYMMDD'), 'YYYYMMDD')
D_CREATED_DATE,
A.CREATED_BY V_CREATED_BY,
A.LAST_MODIFIED_BY V_LAST_MODIFIED_BY,
TO_DATE(TO_CHAR(A.LAST_MODIFIED_DATE, 'YYYYMMDD'), 'YYYYMMDD')
D_LAST_MODIFIED_DATE,
(SELECT BB.VARCHAR_ASSIGN_VALUE
FROM DIM_PRODUCTS_ATTR BB
WHERE A.PRODUCT_ID = BB.PRODUCT_ID
AND BB.ATTRIBUTE_ID IN (SELECT T.ATTRIBUTE_ID FROM
REV_DIM_ATTRIBUTES_TL T WHERE
 T.DIMENSION_ID=4 AND T.ATTRIBUTE_NAME ='PRODUCT TYPE DESC'
AND T.LANGUAGE IN (SELECT DECODE(installed_flg,1,mls_cd,'US') FROM fsi_mls
WHERE
mls_cd=USERENV('LANG')))) V_PROD_TYPE_DESC
FROM DIM_PRODUCTS_B A
LEFT OUTER JOIN (Select *
from REV_HIER_FLATTENED
where dimension_id = 4
AND HIERARCHY_ID =
(SELECT V_COMPONENT_VALUE
 FROM SETUP_MASTER
WHERE V_COMPONENT_DESC = 'PRODUCT_HIER1')) C on A.PRODUCT_ID =
C.LEAF_NODE
```

```
INNER JOIN DIM_PRODUCTS_TL D ON D.PRODUCT_ID = A.PRODUCT_ID
WHERE A.LEAF_ONLY_FLAG = 'Y'
```

DIM_PRODUCT_MULT_HIER_V

```
CREATE OR REPLACE VIEW DIM_PRODUCT_MULT_HIER_V AS
SELECT C.L1_CODE n_product_id_level01,
 C.L1_NAME v_product_name_level01,
 C.L2_CODE n_product_id_level02,
 C.L2_NAME v_product_name_level02,
 C.L3_CODE n_product_id_level03,
 C.L3_NAME v_product_name_level03,
 C.L4_CODE n_product_id_level04,
 C.L4_NAME v_product_name_level04,
 C.L5_CODE n_product_id_level05,
 C.L5_NAME v_product_name_level05,
 C.L6_CODE n_product_id_level06,
 C.L6_NAME v_product_name_level06,
 C.L7_CODE n_product_id_level07,
 C.L7_NAME v_product_name_level07,
 C.L8_CODE n_product_id_level08,
 C.L8_NAME v_product_name_level08,
 C.L9_CODE n_product_id_level09,
 C.L9_NAME v_product_name_level09,
 C.L10_CODE n_product_id_level10,
 C.L10_NAME v_product_name_level10,
 C.L11_CODE n_product_id_level11,
 C.L11_NAME v_product_name_level11,
 C.L12_CODE n_product_id_level12,
 C.L12_NAME v_product_name_level12,
 C.L13_CODE n_product_id_level13,
 C.L13_NAME v_product_name_level13,
 C.L14_CODE n_product_id_level14,
 C.L14_NAME v_product_name_level14,
C.L15_CODE n_product_id_level15,
 C.L15_NAME v_product_name_level15,
 C.L16_CODE n_product_id_level16,
 C.L16_NAME v_product_name_level16,
 C.L17_CODE n_product_id_level17,
 C.L17_NAME v_product_name_level17,
 C.L18_CODE n_product_id_level18,
 C.L18_NAME v_product_name_level18,
 C.L19_CODE n_product_id_level19,
 C.L19_NAME v_product_name_level19,
 C.L20_CODE n_product_id_level20,
 C.L20_NAME v_product_name_level20,
 C.SEQ_01 N_SEQ_01,
 C.SEQ_02 N_SEQ_02,
 C.SEQ_03 N_SEQ_03,
 C.SEQ_04 N_SEQ_04,
 C.SEQ_05 N_SEQ_05,
 C.SEQ_06 N_SEQ_06,
 C.SEQ_07 N_SEQ_07,
 C.SEQ_08 N_SEQ_08,
 C.SEQ_09 N_SEQ_09,
 C.SEQ_10 N_SEQ_10,
 C.SEQ_11 N_SEQ_11,
 C.SEQ_12 N_SEQ_12,
 C.SEQ_13 N_SEQ_13,
 C.SEQ_14 N_SEQ_14,
 C.SEQ_15 N_SEQ_15,
 C.SEQ_16 N_SEQ_16,
 C.SEQ_17 N_SEQ_17,
 C.SEQ_18 N_SEQ_18,
 C.SEQ_19 N_SEQ_19,
 C.SEQ_20 N_SEQ_20,
NVL((SELECT ACCOUNT_TYPE_CD
 FROM FSI_ACCOUNT_TYPE_CD Z
 WHERE ACCOUNT_TYPE_CD =
 (SELECT BB.DIM_ATTRIBUTE_NUMERIC_MEMBER
 FROM DIM_PRODUCTS_ATTR AA, DIM_COMMON_COA_ATTR BB
 WHERE A.PRODUCT_ID = AA.PRODUCT_ID
 AND AA.ATTRIBUTE_ID = '5011'
 AND AA.DIM_ATTRIBUTE_NUMERIC_MEMBER = BB.COMMON_COA_ID
 AND BB.ATTRIBUTE_ID = '5040')),
 0) N_ACCOUNT_TYPE,
 A.PRODUCT_ID N_PRODUCT_ID,
 NVL(A.PRODUCT_CODE, A.PRODUCT_ID) V_PROD_CODE,
 D.PRODUCT_NAME V_PRODUCT_NAME,
 TO_DATE(TO_CHAR(A.CREATION_DATE, 'YYYYMMDD'), 'YYYYMMDD')
D_CREATED_DATE,
 A.CREATED_BY V_CREATED_BY,
 A.LAST_MODIFIED_BY V_LAST_MODIFIED_BY,
 TO_DATE(TO_CHAR(A.LAST_MODIFIED_DATE, 'YYYYMMDD'), 'YYYYMMDD')
 D_LAST_MODIFIED_DATE,
 (SELECT BB.VARCHAR_ASSIGN_VALUE
 FROM DIM_PRODUCTS_ATTR BB
 WHERE A.PRODUCT_ID = BB.PRODUCT_ID
 AND BB.ATTRIBUTE_ID IN
 (SELECT T.ATTRIBUTE_ID
 FROM REV_DIM_ATTRIBUTES_TL T
 WHERE T.DIMENSION_ID = 4
 AND T.ATTRIBUTE_NAME = 'PRODUCT TYPE DESC'
 AND T.LANGUAGE IN
 (SELECT DECODE(installed_flg, 1, mls_cd, 'US')
 FROM fsi_mls
 WHERE mls_cd = USERENV('LANG')))) V_PROD_TYPE_DESC
 FROM DIM_PRODUCTS_B A
```

```
LEFT OUTER JOIN (select HIER1.LEAF_NODE,
 HIER1.LEVEL_11_CODE L1_CODE,
 HIER1.LEVEL_11_NAME L1_NAME,
 HIER1.LEVEL_12_CODE L2_CODE,
 HIER1.LEVEL_12_NAME L2_NAME,
 HIER1.LEVEL_13_CODE L3_CODE,
 HIER1.LEVEL_13_NAME L3_NAME,
 HIER1.LEVEL_14_CODE L4_CODE,
 HIER1.LEVEL_14_NAME L4_NAME,
 HIER1.LEVEL_15_CODE L5_CODE,
 HIER1.LEVEL_15_NAME L5_NAME,
 HIER1.LEVEL_16_CODE L6_CODE,
 HIER1.LEVEL_16_NAME L6_NAME,
 HIER1.LEVEL_17_CODE L7_CODE,
 HIER1.LEVEL_17_NAME L7_NAME,
 HIER1.LEVEL_18_CODE L8_CODE,
 HIER1.LEVEL_18_NAME L8_NAME,
 HIER1.LEVEL_19_CODE L9_CODE,
 HIER1.LEVEL_19_NAME L9_NAME,
 HIER1.LEVEL_20_CODE L10_CODE,
 HIER1.LEVEL_20_NAME L10_NAME,
 HIER2.LEVEL_11_CODE L11_CODE,
 HIER2.LEVEL_11_NAME L11_NAME,
 HIER2.LEVEL_12_CODE L12_CODE,
 HIER2.LEVEL_12_NAME L12_NAME,
 HIER2.LEVEL_13_CODE L13_CODE,
 HIER2.LEVEL_13_NAME L13_NAME,
 HIER2.LEVEL_14_CODE L14_CODE,
 HIER2.LEVEL_14_NAME L14_NAME,
 HIER2.LEVEL_15_CODE L15_CODE,
 HIER2.LEVEL_15_NAME L15_NAME,
 HIER2.LEVEL_16_CODE L16_CODE,
HIER2.LEVEL_16_NAME L16_NAME,
 HIER2.LEVEL_17_CODE L17_CODE,
 HIER2.LEVEL_17_NAME L17_NAME,
 HIER2.LEVEL_18_CODE L18_CODE,
 HIER2.LEVEL_18_NAME L18_NAME,
 HIER2.LEVEL_19_CODE L19_CODE,
 HIER2.LEVEL_19_NAME L19_NAME,
 HIER2.LEVEL_20_CODE L20_CODE,
 HIER2.LEVEL_20_NAME L20_NAME,
 HIER1.SEQ_01,
 HIER1.SEQ_02,
 HIER1.SEQ_03,
 HIER1.SEQ_04,
 HIER1.SEQ_05,
 HIER1.SEQ_06,
 HIER1.SEQ_07,
 HIER1.SEQ_08,
 HIER1.SEQ_09,
 HIER1.SEQ_10,
 HIER1.SEQ_11,
 HIER1.SEQ_12,
 HIER1.SEQ_13,
 HIER1.SEQ_14,
 HIER1.SEQ_15,
 HIER1.SEQ_16,
 HIER1.SEQ_17,
 HIER1.SEQ_18,
 HIER1.SEQ_19,
 HIER1.SEQ_20
 from (select *
 from (Select *
 from rev_hier_flattened
where dimension_id = 4
 AND HIERARCHY_ID =
 (SELECT V_COMPONENT_VALUE
 FROM SETUP_MASTER
 WHERE V_COMPONENT_DESC =
 'PRODUCT_HIER1'))) HIER1
 FULL OUTER JOIN (select *
 from (Select *
 from rev_hier_flattened
 where dimension_id = 4
 AND HIERARCHY_ID =
 (SELECT V_COMPONENT_VALUE
 FROM SETUP_MASTER
 WHERE V_COMPONENT_DESC =
 'PRODUCT_HIER2'))) HIER2 on HIER1.LEAF_NODE =
 HIER2.LEAF_NODE) C on A.PRODUCT_ID =
 C.LEAF_NODE
 INNER JOIN DIM_PRODUCTS_TL D ON D.PRODUCT_ID = A.PRODUCT_ID
 WHERE A.LEAF_ONLY_FLAG = 'Y'
```

In both the hierarchies, level 10 to level 1 represent the first hierarchy and level 20 to level 11 represent the second hierarchy. Both the hierarchies share leaf nodes.

Use the DIM_PRODUCT_MULT_HIER_V.sql, which is a sample view on product dimension. It can be used in the SCD process to move multiple alternate hierarchies defined in the AMHM to the flattened tables used for BI analytics.

This sample view considers two alternate hierarchies. You can extend this for additional hierarchies that may be required. Also, the sample is based on the Product dimension, but can be modified for other dimensions by replacing the relevant tables.

4

Time Dimension Population

Business data commonly represents information as of a point in time (for example, a balance as of a point in time) or as of a particular span of time (for example, income for the month of March). The rollup of a particular balance depending on their nature could be a simple additive rollup wherein the child member balances are added up to arrive at the parent node balance (for example, Ending Balance) or non additive rollups wherein a node formula is used to specify how to rollup the child member balances (for example, 3 month rolling average).

This chapter covers the following topics:

- Overview of Time Dimension Population
- Prerequisites
- Tables Used by the Time Dimension Population Transformation
- Executing the Time Dimension Population Transformation
- Checking the Execution Status

Overview of Time Dimension Population

The twelve month columns in LEDGER_STAT table of OFSPM are replaced by a single N_AS_OF_DATE_SKEY column in OFSEFPA's FCT_LEDGER_STAT table, with each month value stored in N_VALUE column. Similarly, the YTD column value is stored in N_VALUE_YTD. This is done to make reporting easier considering Time is a dimension for most of the OFSEFPA reports. Time dimension population transformation is used to populate the DIM_DATES table with values between two dates specified by the user.

The database components, used by the transformations are:

1. Database function FN_DIM_DATES
2. Database procedure PROC_DIM_DATES_POPULATION that is called by the function FN_DIM_DATES mentioned earlier.

Note: OFSEFPA 6.1 does not use D_FISCAL_YEAR_START_DATE, D_FISCAL_YEAR_END_DATE, or N_DAY_OF_WEEK columns.

Prerequisites

1. All the post install steps mentioned in the *Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration guide* and the solution installation manual of *Enterprise Financial Performance Analytics* have to be completed successfully.

2. Application User must be mapped to a role that has seeded batch execution function (BATPRO).

3. Before executing a Batch, check if the following services are running on the application server:

    1. Iccserver

    2. Router

    3. AM Server

    4. Messageserver

    5. Olapdataserver

    For more information on how to check if the services are up and on and how to start the services if you find them not running, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

4. Batches will have to be created for executing the function. For more details, refer to section How to Define a Batch, page D-1.

Tables Used by the Time Dimension Population Transformation

For more details on viewing the structure of earlier tables, refer to *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *OFSEFPA Erwin Data Model.*

Executing the Time Dimension Population Transformation

You can execute the function from the *Operations* (formerly Information Command Center (ICC) framework) module of OFSAAI.

This component for OFSEFPA 6.1 has been seeded with the Batch ID

<INFODOM>_Dim_Dates_Population, which can be executed from Batch Execution section of OFSAAI. In the Parameter List, enter the Start Date and End Date. For example, '19940101','19941231'.

You can also define a new Batch and an underlying Task definition from the *Batch Maintenance* window of OFSAAI. For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

To define a new task for a Batch definition:

1. Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

2. Click **Add** (+) button from the Task Details grid.

   The *Task Definition* window is displayed.

3. Enter the **Task ID** and **Description**.

4. Select **Transform Data** component from the drop down list.

5. Select the following from the **Dynamic Parameters** list:

   - **Datastore Type** - Select the appropriate datastore type from the list.

   - **Datastore Name** - Select the appropriate datastore name from the list.

   - **IP address** - Select the IP address from the list.

   - **Rule Name** - Select **Dim_Dates_Population** from the drop down list of available transformations. (This is a seeded Data Transformation which is installed as part of the OFSEFPA solution installer. If you don't see this in the list, contact Oracle support)

   - **Parameter List –** Enter the Start Date and End Date.

     - **Start Date –** This is the starting date, from which the Transformation will populate DIM_DATES table. This date should be specified in 'YYYYMMDD' format.

       For example, '**20081131**'.

     - **End Date** - This is the end date, to which the Transformation will populate DIM_DATES table. This date should also be specified in 'YYYYMMDD' format.

       For example, '**20091231**'.

6. Click **Save**.

   The Task definition is saved for the selected Batch.

7. Execute the batch.

   You can execute a Batch definition from the *Batch Execution* section of *OFSAAI Operations* module.

The function can also be executed directly on the database through SQLPLUS

Details are:

Function Name: FN_DIM_DATES

Parameters: P_BATCH_RUN_ID, P_AS_OF_DATE, P_ST_DT, and P_ED_DT

Sample Parameter Values: 'Batch1', '20091231', '20081131', and '20091231'

> **Note:** This DT should be executed for each year for which data is present in the source table.

Checking the Execution Status

The Batch execution status can be monitored through *Batch Monitor* section of *OFSAAI Operations* module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S – Success

The execution log can also be accessed on the application server in the directory *$FIC_DB_HOME/log/date*, where file name will have the Batch Execution ID.

You can access the database level operations log by querying the FSI_MESSAGE_LOG table. Filter the Batch Run ID column for identifying the relevant log.

> **Note:** Check the **.profile** file in the installation home if you are unable to find the above mentioned path.

5

Modification and Mapping of Reporting Lines

Overview

Reporting lines are configured as hierarchies within the Business Metadata of Oracle Financial Services Analytical Applications Infrastructure (OFSAAI). Reporting Line hierarchy originates in Reporting layer so it is enabled in AMHM. User needs to click on the Application Preferences screen to be able to create a new hierarchy of Reporting Line.

- **Reporting Lines for Financial Reporting**
  - Financial Reporting Line is based on General Ledger dimension. (DIM_GL_ACCOUNT table).
  - This reporting line hierarchy is a REGULAR parent-child hierarchy, based on the members of GL dimension.
- **Reporting Lines for Management Reporting**
  - Management Reporting Line is based on Reporting Line dimension (DIM_REP_LINE table).
  - This reporting line hierarchy is a REGULAR parent-child hierarchy based on the members of Reporting line dimension.

Summary of Configuration Required for Financial Reporting and Management Reporting

Financial Reporting (FR)

Reporting Line is based on General Ledger dimension. Therefore, all the GL codes for this subject area should functionally relate to Income Statement line items. The idea of this report is to show the un-allocated data coming in from the source system. The Income Statement General Ledger accounts in the source system need to be understood and be categorized as either income or expense GLs. The tag of income/expense can be achieved by putting in the appropriate Financial Element Code (FE) (for example, 420 for Interest, 455 for Non Interest Income, and 457 for Non Interest Expense). Each leaf level GL will be categorized as either Income or Expense with the appropriate FE mapping. Financial Reporting Income Statement hierarchy can then be constructed by using the leaf level GLs and rolling them up to higher level nodes. Assign the rollup signage (+/-) to each leaf/node while constructing the hierarchy. Data is expected to be present only for the leaf level GL codes. Once the FR Income Statement hierarchy (can also be called as Income Statement from source ledger data) is constructed same can be made visible in Financial Reporting Dashboard by running the hierarchy transformation and SCD of GL dimension.

Management Reporting (MR)

Income Statement (MR-IS)

Management Reporting Income Statement (MR-IS) is seeded in the system and is available in the system using the reporting line codes present in the DIM_REP_LINE table. MR-IS line items are much more granular and data is created for them using the allocation engine (for example, OFSPM). The allocated data is expected to be present in the LEDGER_STAT table. Line items of MR-IS are constructed based on Reporting line codes, but the data availability is expected into the seeded Financial Element Codes. EFPA seeds 9000 series FEs and allocation should be done to this series. Once the allocation engine provides data for these Financial Elements, the MR-IS will show results in the OBIEE dashboards. There is a mapping table DIM_REPORTING_LINE_ATTR that ties up Reporting Line codes and Financial Elements. Seeded Financial Element list is provided in the install guide of OFSEFPA. The mapping between Reporting Lines and Financial Elements is 1:1 and one FE can only be mapped to a single leaf level Reporting Line code of MR Income Statement hierarchy.

Balance Sheet (MR-BS)

Management Reporting Balance Sheet (MR-BS) is seeded in the system and is available in the system using the reporting line codes present in the DIM_REP_LINE table.

Mapping of Balance Sheet General Ledger codes to Reporting Lines is explained in section Mapping of Balance Sheet Leaf Nodes to General Ledger, page 5-5.

Addition and Modification of Reporting Lines Using AMHM

Modifying Financial Reporting Line

Financial Reporting line hierarchy can be modified to include new members and modify existing members. As a prerequisite, if the member to be added is a leaf, GL dimension table (DIM_GL_ACCOUNT) needs to have these leaf members.

GL member

When creating a new GL member in AMHM, the signage attribute should be properly mapped to the GL. This signage will be used as rollup signage in the GL hierarchy for node level calculation.

GL Hierarchy

General ledger Hierarchy is used for Financial Reporting. The hierarchy should build with proper parent and child level mapping from the topmost root to leaf level. The node level GL entries should not be present in the source table, that is LEDGER_STAT. Only leaf level GLs are allowed to have data.

New members can be created from the *Financial Services Applications* > *Dimension Management* screen of OFSAAI

Modifying Management Reporting Line

Management Reporting line hierarchy can be modified to include new members and modify existing members. As a prerequisite, if the member to be added is a leaf, Reporting Line dimension table (DIM_REP_LINE) needs to have these leaf members.

The table DIM_REP_LINE table is populated using AMHM tables, that is DIM_REPORTING_LINE_B, DIM_REPORTING_LINE_TL, DIM_REPORTING_LINE_ATTR, and DIM_REPORTING_LINE_HIER. The hierarchy has to be modified from the *Financial Services Applications* > *Dimension Management* section of OFSAAI, so that relevant changes are captured in the AMHM tables mentioned above. Once the hierarchy is saved, the DT has to be executed to populate DIM_REP_LINE and REP_LINE_PARENT_CHILD_RELATION tables.

Additional reporting line hierarchy can also be created using existing nodes using the AMHM framework of OFSAAI. If there are reporting line codes created, then you should create all such custom reporting lines using a sequence of numbers that is different from the numbers used in seeded reporting line hierarchy. It is recommended that you should create custom reporting lines with a six digit number starting with 500000, 500001, and so on.

Mapping of Reporting Line Items

The reporting lines are seeded in the application and are used during hierarchy rollups in OBIEE reports. These reporting lines can be classified broadly into two types, that is Income Statement (IS) reporting lines and Balance Sheet (BS) reporting lines. All reporting lines are part of either IS hierarchy or BS hierarchy.

All reporting line codes of DIM_REPORTING_LINE_ATTR or REP_LINE_GL_MAP should be at leaf level reporting lines. When creating a new Reporting line item in AMHM, the signage attribute should be properly assigned to the Reporting line. This signage will be used as rollup signage in the Reporting Line hierarchy for node level calculation.

When a new financial element needs to be seeded into the application, you should manually seed the data into the below tables with unique FE IDs:

DIM_FINANCIAL_ELEMENTS_B

DIM_FINANCIAL_ELEMENTS_TL

DIM_FINANCIAL_ELEMENTS_ATTR

Once the above tables are loaded, DIM_FINANCIAL_ELEMENT has to be populated. The database components used to populate DIM_FINANCIAL_ELEMENT are:

- Database function FN_DIM_FINANCIAL_ELEM_UPDATEWR
- Database function FN_DIM_FINANCIAL_ELEM_UPDATE, which is called by the function FN_DIM_FINANCIAL_ELEM_UPDATEWR mentioned above.

The seeded Batch **<INFODOM>_Financial_Element_UPD** triggers the DT FN_DIM_FINANCIAL_ELEMENT that loads the DIM_FINANCIAL_ELEMENT table.

Parameter List: OFSAA User

Example: 'OFSAAUSER'

Mapping of Income Statement Leaf Nodes to Financial Elements

IS reporting lines are seeded into the application and a default mapping is provided between Financial Element (FE) and Reporting Lines. The mapping of FE and IS reporting lines is present in DIM_REPORTING_LINE_ATTR table. This mapping can be enhanced based on the requirements and the data need to be created as tabulated below:

Table Name: DIM_REPORTING_LINE_ATTR

Columns: REPORTING_LINE_ID, ATTRIBUTE_ID, and DIM_ATTRIBUTE_NUMERIC_MEMBER

Column Name	Description	Remarks
REPORTING_LINE_ID	Column to stores Reporting Line Identifiers	
ATTRIBUTE_ID	Column to store the type of attribute used for Mapping	1-Financial Element 2-Signage
DIM_ATTRIBUTE_NUMERIC_MEMBER	Column to store the actual attribute i.e. Financial Element or Signage	a) Actual Financial Element Code is used in this column in case the ATTRIBUTE_ID = 1 b) Signage value 1 or 2 is used in this column to signify positive signage or negative signage. Signage is applicable only when ATTRIBUTE_ID=2

- It is not required to process the mapping of financial elements 140 and 100 to reporting lines (Average Bal and End Bal).

- A single reporting line should be mapped only to one financial element.

- IS reporting lines are applicable for allocated data from OFSPM.

- Management Reporting - Income Statement: The Identity Codes available in LEDGER_STAT table, having source_type as 100 are considered for populating Income Statement reporting lines in FCT_MGMT_REPORTING table.

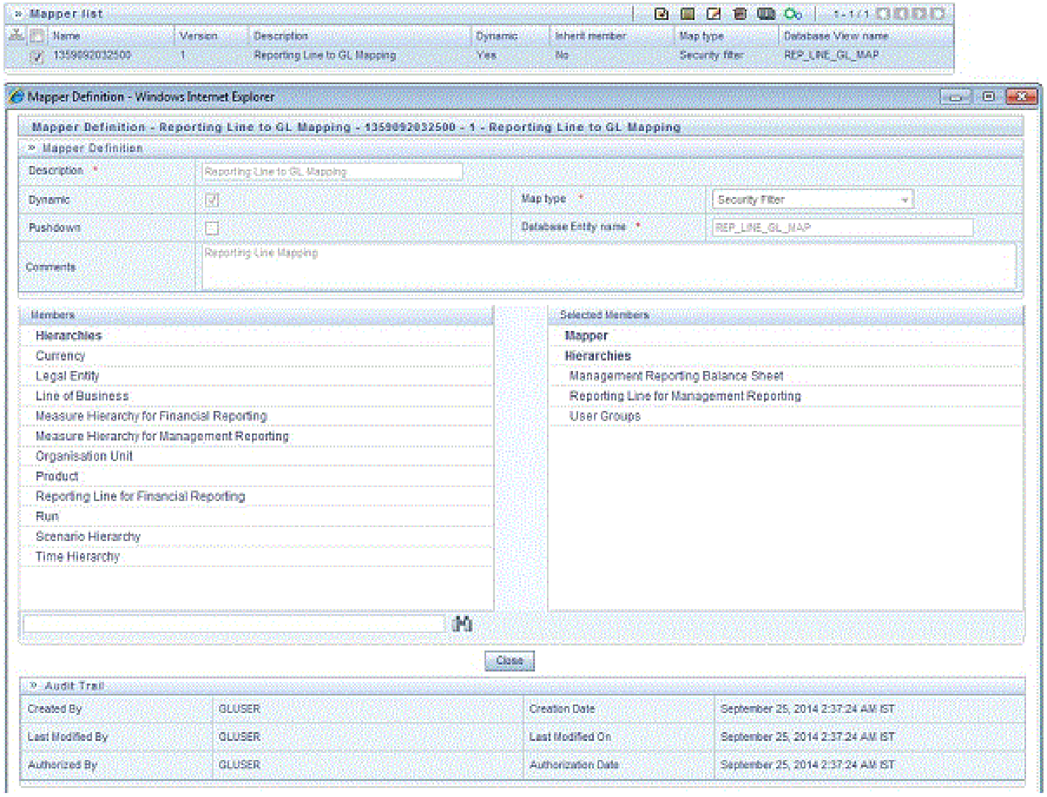
```
 Select identity_code from FSI_DATA_IDENTITY where
source_type = 100;
```

Mapping of Balance Sheet Leaf Nodes to General Ledger

BS reporting lines are seeded in OFSEFPA application. The mapping of BS reporting lines are based on General Ledger codes, as they are available to OFSEFPA from Profitability Management. The data considered for BS hierarchy is un-allocated data, that is the data available in the OFSPM in un-allocated form (for all those IDENTITY_CODE where SOURCE_TYPE is '0' in the FSI_DATA_IDENTITY table). However, default mapping cannot be provided as the general ledger codes can differ at the customer site. The mapping of BS reporting lines and general ledger codes are done from the *Map Maintenance* section of OFSAAI. Map definition has been seeded in the OFSEFPA application for users to create the mapping of reporting lines (leaf nodes) of Reporting Line hierarchy with the general ledger hierarchy (leaf nodes). The general ledger hierarchy is evolved from DIM_GL_ACCOUNT table through HEPMGLBS. In the SETUP_MASTER table configuration (key-value pair) has been seeded to allow the mapping.

V_COMPONENT_CODE	V_COMPONENT_DESC	V_COMPONENT_VALUE
REP_LINE_GL_MAPPER	REP_LINE_GL_MAPPER	REP_LINE_GL_MAP

A new map definition can be created from the in *Map Maintenance* section of OFSAAI and the new table name against the key provided (REP_LINE_GL_MAPPER) in SETUP_MASTER table can be used.



Mapper list

1 - 1 / 1

	Name	Version	Description	Dynamic	Inherit member	Map type	Database View name
☑	1359092032500	1	Reporting Line to GL Mapping	Yes	No	Security filter	REP_LINE_GL_MAP

Mapper Definition - Windows Internet Explorer

Mapper Definition - Reporting Line to GL Mapping - 1359092032500 - 1 - Reporting Line to GL Mapping

Mapper Definition

Description *	Reporting Line to GL Mapping		
Dynamic	☑	Map type *	Security Filter
Pushdown	☐	Database Entity name *	REP_LINE_GL_MAP
Comments	Reporting Line Mapping		

Members	Selected Members
Hierarchies	**Mapper**
Currency	**Hierarchies**
Legal Entity	Management Reporting Balance Sheet
Line of Business	Reporting Line for Management Reporting
Measure Hierarchy for Financial Reporting	User Groups
Measure Hierarchy for Management Reporting	
Organisation Unit	
Product	
Reporting Line for Financial Reporting	
Run	
Scenario Hierarchy	
Time Hierarchy	

Audit Trail

Created By	GLUSER	Creation Date	September 25, 2014 2:37:24 AM IST
Last Modified By	GLUSER	Last Modified On	September 25, 2014 2:37:24 AM IST
Authorized By	GLUSER	Authorization Date	September 25, 2014 2:37:24 AM IST

- Mapping of more than one general ledger leaf node can be done to a single reporting line.

- V_COMPONENT_VALUE is the column that should be changed in case a new mapper is created on OFSAAI platform.

You can map the Balance Sheet leaf level GLs to leaf level Balance sheet Reporting Lines from the *Map Maintenance* screen of OFSAAI. However, this mapping can also be done directly in the database table that stores the mapping. Use the following table details to map leaf level codes in the map table:

- **Table Name**: REP_LINE_GL_MAP
- **GL ID Column**: V_MEMBER_1

  Leaf level General Ledger ID that needs to be mapped to the leaf level Node of seeded Balance Sheet Reporting Line.
- **Balance Sheet Reporting Line Code**: V_MEMBER_2

  Leaf level Balance Sheet Reporting Line Code. The Balance Sheet Reporting line hierarchy is seeded in the application and it starts with node 107003.
- **Map ID**:V_Map_ID

  1359092032500 (It is the seeded ID of the mapper that is present after installation).
- **User Group Name**: V_Member_3

  User Group- It is the user group to which the logged in user belongs. You can refer to the below query to find it (in oonfig schema).

  SELECT * FROM cssms_usr_group_map where v_usr_id = <>

Example

Insert into REP_LINE_GL_MAP (V_MAP_ID, N_MAP_ID, N_INHERIT_MAP_ID, V_MEMBER_1, V_MEMBER_2, V_MEMBER_3, V_MEMBER_4, V_MEMBER_5, V_MEMBER_6, V_MEMBER_7, V_MEMBER_8, V_MEMBER_9)

Values ('1359092032500', null, null, '12345700000149', '50100', 'TESTGRP', null, null, null, null, null, null)

12345700000149 - n_gl_account_id in DIM_GL_ACCOUNT table of dev env.

50100 - Leaf level node from Balance Sheet hierarchy (Refer to the attached excel).

TESTGRP – test user group.

The Financial Elements (FE) supported for the MR- Balance Sheet are 100 and 140. Therefore, the GL IDs for which FE is either 100 or 140 will be picked up to populate Balance Sheet reporting line ID in FCT_MGMT_REPORTING table.

> **Note:** The Identity Codes available in LEDGER_STAT table, having source_type as 0 are considered for populating Balance Sheet reporting lines in FCT_MGMT_REPORTING table.
>
> ```
>  Select identity_code from FSI_DATA_IDENTITY where
> source_type = 0;
> ```

6

Multi-Currency Reporting

Introduction

A bank may desire to represent the reports in different currencies depending on geographic presence, regulatory requirements, and so on. The functionality of multi-currency reporting enables the reports to be displayed in multiple currencies.

Overview of Multi-Currency Reporting

There are three different currency types that are supported and they are as follows:

- Local Currency
- Regional Currency
- Reporting Currency

Each currency type needs to be mapped to a currency and the reports can then be viewed in the corresponding currency. Using the exchange rate prevailing at the time the data is loaded, the conversion from the reporting currency (which is the default currency type) to another currency is done. The selection of the currency in which reports are to be viewed is done through dashboard prompts.

7

Fact Ledger Population

Fact Ledger population involves populating the FCT_LEDGER_STAT table from the LEDGER_STAT table.

This chapter covers the following topics:

- Overview of Fact Ledger Population Transformation
- Prerequisites
- Tables Used by the Fact Ledger Population Transformation
- Map New Dimensions
- Executing the Fact Ledger Population Transformation
- Checking the Execution Status
- Optional Load from Ledger Class Entities for Fact Ledger Population
- Optional Load from Ledger Class Entities for Fact Ledger Population
- Identity Codes Used in Financial Reporting

Overview of Fact Ledger Population Transformation

The LEDGER_STAT table is optimized for processing purposes, but is not a convenient structure for reporting purposes. In generating FACT_LEDGER_STAT, time from LEDGER_STAT is transformed into an explicit dimension in FACT_LEDGER_STAT.

Fact Ledger Population transformation is used to populate the FCT_LEDGER_STAT table from the Profitability LEDGER_STAT table. The horizontally structured MONTH and YTD columns in Ledger/Stat are transposed to a vertical structure. The twelve Month Columns in LEDGER_STAT are replaced by a single N_AS_OF_DATE_SKEY column in FCT_LEDGER_STAT with each month value stored in N_VALUE column. Similarly, the YTD column value is stored in N_VALUE_YTD. This is done to make reporting easier, considering Time is a dimension in most of the reports.

The database components, used by the Fact Ledger Population transformations are:

1. Database function FSI_LEDGER_STAT_TRM

2. Database function LEDGER_STAT_TRM which is called by the function FSI_LEDGER_STAT_TRM as mentioned earlier.

Prerequisites

1. All the post install steps mentioned in the *Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration guide* and the solution installation manuals of *Profitability Management* (only if OFSPM is installed) and *Enterprise Financial Performance Analytics* have to be completed successfully.

2. Application User must be mapped to a role that has seeded batch execution function (BATPRO).

3. Ensure that your FISCAL year information is configured properly. It has the following two columns.

   1. FISCAL_PERIOD: This gives the number of months in the given FISCAL period

   2. START_MONTH: This indicates which month of the calendar year is the FISCAL starting month. For example, a value '1' for this column means FISCAL year starts from January and value of '4' indicates that the FISCAL year starts from April.

4. Before executing a batch, check if the following services are running on the application server.

   1. Iccserver

   2. Router

   3. AM Server

   4. Messageserver

   5. Olapdataserver

   For more information on how to check if the services are up and how to start the services, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

5. Batches will have to be created for executing the function. For more information, refer to section How to Define a Batch, page D-1.

Tables Used by the Fact Ledger Population Transformation

- FSI_FISCAL_YEAR_INFO - This table has the FISCAL year info. The entries required in this table are mentioned in the Prerequisites, page 7-2 section.

- FSI_BI_SETUP_TABLE - This table has the setup information used by the Transformation. They are :

  - TARGET_TABLE_NAME: This is the destination table name for transformation.

  - TARGET_COLUMN_NAME: This is the Destination column name in FCT_LEDGER_STAT table.

  - MEMBER_COL_NAME: This is the column Name in LEDGER_STAT table.

  - SOURCE_DIM_TABLE_NAME: This is the dimension table to which the Ledger data has to be joined to get the surrogate key value.

  - SOURCE_COLUMN_NAME: This is the column in the dimension table to which the LEDGER STAT ID column is joined.

  - SKEY_COLUMN_NAME: This is the column in the dimension table which has the surrogate key value.

  - JOIN_REQUIRED: This provides the information, whether the column to be moved to FCT_LEDGER_STAT is directly available in LEDGER_STAT or a join has to be taken with dimension table to get the skey.

Sample data for this table is plotted below:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_LEDGER_STAT	N_BALANCE_TYPE_CD	BALANCE_TYPE_CD	LEDGER_STAT	BALANCE_TYPE_CD	BALANCE_TYPE_CD	N	
FCT_LEDGER_STAT	N_COMMON_COA_SKEY	COMMON_COA_ID	DIM_COMMON_COA	N_COMMON_COA_ID	N_COMMON_COA_SKEY	Y	

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_LEDGER_STAT	N_CONSOLIDATION_CD	CONSOLIDATION_CD	LEDGER_STAT	CONSOLIDATION_CD	CONSOLIDATION_CD	N	
FCT_LEDGER_STAT	N_ENTITY_SKEY	V_ENTITY_SKEY	DIM_ORG_STRUCTURE	-1	-1	N	
FCT_LEDGER_STAT	N_FINANCIAL_ELEM_ID	FINANCIAL_ELEM_ID	LEDGER_STAT	FINANCIAL_ELEM_ID	FINANCIAL_ELEM_ID	N	
FCT_LEDGER_STAT	N_GL_ACCOUNT_SKEY	GL_ACCOUNT_ID	DIM_GL_ACCOUNT	N_GL_ACCOUNT_ID	N_GL_ACCOUNT_SKEY	Y	
FCT_LEDGER_STAT	N_IDENTITY_CODE	IDENTITY_CODE	LEDGER_STAT	IDENTITY_CODE	IDENTITY_CODE	N	
FCT_LEDGER_STAT	N_LOB_SKEY	V_LOB_SKEY	DIM_LOB	-1	-1	N	
FCT_LEDGER_STAT	N_ORG_UNIT_SKEY	ORG_UNIT_ID	DIM_ORG_UNIT	N_ORG_UNIT_ID	N_ORG_UNIT_SKEY	Y	
FCT_LEDGER_STAT	N_PROD_SKEY	PRODUCT_ID	DIM_PRODUCT	N_PRODUCT_ID	N_PROD_SKEY	Y	

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_LEDGER_STAT	V_ACCUMULATION_TYPE_CD	ACCUM_TYPE_CD	LEDGER_STAT	ACCUM_TYPE_CD	ACCUM_TYPE_CD	N	
FCT_LEDGER_STAT	V_ISO_CURRENCY_CD	ISO_CURRENCY_CD	LEDGER_STAT	ISO_CURRENCY_CD	ISO_CURRENCY_CD	N	

- LEDGER_STAT - This table is the source for the transformation.
- DIM_<dimension Name> - The flattened dimension tables used in Business Intelligence (BI) reporting are accessed to obtain the surrogate key to be populated to FCT_LEDGER_STAT dimension columns.

  For example, DIM_ORG_UNIT, DIM_PRODUCT, and so on.
- FCT_LEDGER_STAT - This is the output table for the transformation.

For more details on viewing the structure of these tables, refer to *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *OFSEFPA Erwin Data Model*.

Map New Dimensions

The Custom Dimensions can be added to target table, FCT_LEDGER_STAT by updating the configuration table FSI_BI_SETUP_TABLE. DIM_LOB and Dim_Org_Structure can be loaded manually or by using the AMHM tables and batch.

> **Note:** All nodes in FCT_LEDGER_STAT table should be part of the respective hierarchies for which the SCDs run. Otherwise, the cube will fail.

Hierarchy Configuration for LOB and Legal Entity

DIM_LOB

The hierarchy to be built for LOB (Line of Business) is a level based hierarchy. All leaf nodes of this hierarchy are mapped to a single root. The branch navigation of the hierarchy starts from root node which is the Level 1 column of the DIM_LOB table. The Leaf node has to be placed at the appropriate column between Level 1 and Level 16 of the DIM_LOB table. This step is required only if the reports are working out of ESSBASE cubes.

DIM_ORG_STRUCTURE

The hierarchy to be built for DIM_ORG_STRUCTURE is parent-child hierarchy. The parent node of the root has to be kept blank in the V_PARENT_CODE column. The hierarchy can then be built appropriately by putting a child node - parent node combination in the V_ENTITY_CODE and V_PARENT_CODE columns of the DIM_ORG_STRUCTURE table. This metadata setup supports only one hierarchy for Legal Entity.

Follow the below steps manually to populate data into DIM_LOB and DIM_ORG_STRUCTURE custom dimensions for the fact table:

Procedure

1. Add the columns N_LOB_ID and N_ENTITY_ID as part of Unique Key (Ledger_stat) in FCT_LEDGER_STAT table.

2. Update the columns SOURCE_COLUMN_NAME and SKEY_COLUMN_NAME for the target table FCT_LEDGER_STAT in FSI_BI_SETUP_TABLE, with the actual source and skey column values, that is N_LOB_ID and N_ENTITY_ID.

3. Set the value to 'Y' in JOIN_REQUIRED column of FCT_LEDGER_STAT table.

Seeded entry of FSI_BI_SETUP_TABLE:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_LEDGER_STAT	N_ENTITY_SKEY	V_ENTITY_SKEY	DIM_ORG_STRUCTURE	-1	-1	N	
FCT_LEDGER_STAT	N_LOB_SKEY	V_LOB_SKEY	DIM_LOB	-1	-1	N	

Updated Entry of FSI_BI_SETUP_TABLE:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_LEDGER_STAT	N_ENTITY_SKEY	N_ENTITY_ID	DIM_ORG_STRUCTURE	N_ENTITY_ID	N_ENTITY_SKEY	Y	
FCT_LEDGER_STAT	N_LOB_SKEY	N_LOB_ID	DIM_LOB	N_LOB_ID	N_LOB_SKEY	Y	

- The new data transformation FN_ORG_PARENT_CHILD has to be re executed with respect to any change in DIM_ORG_STRUCTURE table. This process loads the ORG_STR_PARENT_CHILD_RELATION table data, which is sourced from DIM_ORG_STRUCTURE table

- The columns N_ENTITY_ID and N_LOB_ID are only required to be present in LEDGER_STAT if user decides to move data of LOB and ENTITY dimensions to EFPA reporting area. The seeded entries provided during installation doesn't mandate that the both these columns need to present before the transformation is executed.

Executing the Fact Ledger Population Transformation

You can execute the function from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI.

This component for OFSEFPA 6.1 has been seeded with the following Batch ID **<INFODOM>_Fact_Table_Transformation - Task1**, which can be executed from *Batch Execution* section of OFSAAI. A single Batch triggers the transformation for both FCT_LEDGER_STAT and FCT_MGMT_REPORTING as separate tasks. You can execute these task individually, by excluding the other. In the Parameter List, include pstart_month, pend_month, pyears, pidentity_code, psource_type, pre_run_flg, and prcy. For example, 1,12,1994,'','','Y','USD'.

You can also define a new Batch and an underlying Task definition from the *Batch Maintenance* window of OFSAAI. For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

To define a new task for a Batch definition:

1. Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

2. Click **Add** (+) button from the Task Details grid.

   The *Task Definition* window is displayed.

3. Enter the **Task ID** and **Description**.

4. Select **Run Executable** component from the drop down list.

5. Select the following from the **Dynamic Parameters** list:

   - **Datastore Type** - Select the appropriate datastore type from the list.

   - **Datastore Name** - Select the appropriate datastore name from the list.

   - **IP address** - Select the IP address from the list.

   - **Rule Name** - Select **FSI_LEDGER_STAT_TRM** from the drop down list of available transformations. (This is a seeded Data Transformation which is installed as part of the OFSEFPA solution installer. If you don't see this in the list, contact Oracle support).

- **Parameter List** - Enter pStart_Month , pEnd_Month , pYears , pIdentity_Code , pSource_Type , pRe_Run_Flg, and pRCY.
  - **pStart_Month** - This parameter indicates the starting Month.
  - **pEnd_Month** - This parameter indicates the ending Month.
  - **pYears** - This is a mandatory parameter that indicates the Year value.
  - **pIdentity_Code** - This is an optional parameter that indicates the Identity Code.

    This is the identity code in OFSPM LEDGER_STAT table. The value '0' in this field indicates, only the rows in LEDGER_STAT with source code '0' should get processed. Source code '0' indicates rows in LEDGER_STAT loaded by the ledger load program. This results in movement of rows loaded by ledger load program to FCT_LEDGER_STAT in OFSEFPA solution. Similarly, any particular allocation output values can be moved by filtering on the identity code.
  - **pSource_Type** - This is an optional parameter that indicates the Source Type.

    Source Type indicate which process populated a row in LEDGER_STAT of profitability solution. For example, '0' indicates it was loaded by the Ledger Load program and '100' indicates the allocation rule populated it, and so on.
  - **pRe_Run_Flg** - This is an optional parameter that indicates Re-run Flag. If value is 'Y', the existing data in the fact table will be removed and reloaded.
  - **pRCY** - This indicates the reporting currency with Default Value '**USD**'.

6. Click **Save**.

   The Task definition is saved for the selected Batch.

7. Execute the Batch.

   You can execute a Batch definition from the *Batch Execution* section of *OFSAAI Operations* module.

Ledger Stat Transformation can also be directly executed on the database through SQLPLUS.

Details are:

Function Name: FSI_LEDGER_STAT_TRM

Parameters : pBatch_Id, pAs_of_date, pStart_Month , pEnd_Month, pYears , pIdentity_Code, pSource_Type, pRe_Run_Flg, and prcy.

Sample parameter values are 'Batch1', '20091231', 1, 8, 2009, 0, 0, 'Y', and 'GBP' respectively.

Checking the Execution Status

The Batch execution status can be monitored through *Batch Monitor* section of *OFSAAI Operations* module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S - Success

The execution log can also be accessed on the application server in the directory *$FIC_DB_HOME/log/date*, where file name will have the Batch Execution ID.

You can access the database level operations log by querying the FSI_MESSAGE_LOG table. Filter the Batch Run ID column for identifying the relevant log.

> **Note:** Check the **.profile** file in the installation home if you are unable to find this path.

Optional Load from Ledger Class Entities for Fact Ledger Population

Batch Name: <Infodom>_MANAGEMENT_LEDGER

Task1: Loads data from Fsi_D_Management_Ledger to Fct_Ledger_stat

Parameter to be configured: [DRCY]=To_Currency_Code

Example
[DRCY]=USD

Data will be loaded for the data selected while running the batch.

Optional Load from Ledger Class Entities for Fact Ledger Population

Batch Name: <Infodom>_MANAGEMENT_LEDGER

Task1: Loads data from Fsi_D_Management_Ledger to Fct_Ledger_stat

Parameter to be configured: [DRCY]=To_Currency_Code

Example
[DRCY]=USD

Data will be loaded for the data selected while running the batch.

Identity Codes Used in Financial Reporting

Financial Reporting reports are used to show the custom built GL hierarchy. This GL hierarchy is treated as a Reporting LIne dimension in Financial Reporting (FR) Dashboard pages. The idea of reports in FR is to show the unallocated data that has been downloaded from source system and rule out the allocated data records. In order to filter out the unallocated data a SETUP_MASTER configuration table entry is required with V_COMPONENT_DESC column value as 'IDENTITY_CODE_PFTBI_FR_UNALLOCATED_DATA'. The other two columns of SETUP_MASTER can be popuated as: V_COMPONENT_CODE- Identity Code corresponding to unallocated data. V_COMPONENT_VALUE- Identity Code corresponding to unallocated data. Refer to FSI_DATA_IDENTITY table to find out the IDENTITY CODES. Generally, the source_type = 0 highlights the records that are unallocated.

8

Fact Management Reporting Population

Fact Management Reporting Population involves populating the FCT_MGMT_REPORTING table from the LEDGER_STAT table.

This chapter covers the following topics:

- Overview of Fact Management Reporting Transformation
- Prerequisites
- Tables Used by the Fact Management Reporting Transformation
- Map New Dimensions
- Executing the Fact Management Reporting Transformation
- Checking the Execution Status
- Optional Load from Ledger Class Entities for Fact Management Reporting Population
- Rollup Signage and Operational Signage

Overview of Fact Management Reporting Transformation

The LEDGER_STAT table is optimized for processing purposes, but is not a convenient structure for reporting purposes. In generating FCT_MGMT_REPORTING table, time from LEDGER_STAT table is transformed into an explicit dimension in FCT_MGMT_REPORTING table. Fact Management Reporting transformation is used to populate the FCT_MGMT_REPORTING table from the Profitability LEDGER_STAT table.

During FCT_MGMT_REPORTING table population, the horizontally structured MONTH column in LEDGER_STAT is transposed to a vertical structure. The twelve Month Columns in LEDGER_STAT table are replaced by a single N_DATE_SKEY column in FCT_MGMT_REPORTING table. The fact table is populated with reporting line codes from DIM_REP_LINE table.

The database components used by the Fact Management Reporting transformation are:

- Database function FN_MGMT_REPORTING_TRM
- Database function MGMT_REPORTING_TRM, which is called by the function FN_MGMT_REPORTING_TRM mentioned above.

Prerequisites

1. All the post install steps mentioned in the *Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration guide* and the solution installation manuals of *Profitability Management* (only if OFSPM is installed) and *Enterprise Financial Performance Analytics* have to be completed successfully.

2. Application User must be mapped to a role that has seeded batch execution function (BATPRO).

3. Ensure that your FISCAL year information is configured properly. It has the following two columns:

    1. FISCAL_PERIOD: This gives the number of months in the given FISCAL period.

    2. START_MONTH: This indicates which month of the calendar year is the FISCAL starting month. For example, a value '1' for this column means FISCAL year starts from January and value of '4' indicates that the FISCAL year starts from April.

4. Before executing a batch, check if the following services are running on the application server:

    1. Iccserver

    2. Router

    3. AM Server

    4. Messageserver

    5. Olapdataserver

    For more information on how to check if the services are up and how to start the services, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

5. The following seeded Batch should be available for execution: **<INFODOM>_Fact_Table_Transformation – Task2**.

Tables Used by the Fact Management Reporting Transformation

- FSI_FISCAL_YEAR_INFO - This table has the FISCAL year info. The entries required in this table are mentioned in the Prerequisites, page 8-2 section.

- FSI_BI_SETUP_TABLE - This table has the setup information used by the Fact Management Reporting Transformation.

  They are:

  - TARGET_TABLE_NAME - This is the destination table name for transformation.

  - TARGET_COLUMN_NAME - This is the Destination column name in FCT_MGMT_REPORTING table.

  - MEMBER_COL_NAME - This is the column Name in source tables.

  - SOURCE_DIM_TABLE_NAME - This is the table to which the Ledger data has to be joined to get the surrogate key value.

  - SOURCE_COLUMN_NAME - This is the column in the dimension table to which the LEDGER_STAT_ID column is joined.

  - SKEY_COLUMN_NAME - This is the column in the dimension table which has the surrogate key value.

  - JOIN_REQUIRED - This provides the information, whether the column to be moved to FCT_MGMT_REPORTING table is directly available in LEDGER_STAT table or a join has to be taken with dimension table to get the skey.

  - GROUP_BY_REQUIRED - The group by is used to find the unique records of LEDGER_STAT for which the key column should exist as part of target table's primary key.

    Sample data for this table is plotted below:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_MGMT_REPORTING	N_ENTITY_SKEY	V_ENTITY_SKEY	DIM_ORG_STRUCTURE	-1	-1	N	Y
FCT_MGMT_REPORTING	N_LOB_SKEY	V_LOB_SKEY	DIM_LOB	-1	-1	N	Y
FCT_MGMT_REPORTING	N_IDENTITY_CODE	IDENTITY_CODE	LEDGER_STAT	IDENTITY_CODE	IDENTITY_CODE	N	N
FCT_MGMT_REPORTING	N_ORG_UNIT_SKEY	ORG_UNIT_ID	DIM_ORG_UNIT	N_ORG_UNIT_ID	N_ORG_UNIT_SKEY	Y	Y
FCT_MGMT_REPORTING	N_GL_ACCOUNT_SKEY	GL_ACCOUNT_ID	DIM_GL_ACCOUNT	N_GL_ACCOUNT_ID	N_GL_ACCOUNT_SKEY	Y	N
FCT_MGMT_REPORTING	N_PROD_SKEY	PRODUCT_ID	DIM_PRODUCT	N_PRODUCT_ID	N_PROD_SKEY	Y	Y
FCT_MGMT_REPORTING	N_COMMON_COA_SKEY	COMMON_COA_ID	DIM_COMMON_COA	N_COMMON_COA_ID	N_COMMON_COA_SKEY	Y	N

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_MGMT_REPORTING	V_ISO_CURRENCY_CD	ISO_CURRENCY_CD	LEDGER_STAT	ISO_CURRENCY_CD	ISO_CURRENCY_CD	N	Y
FCT_MGMT_REPORTING	N_SCENARIO_CD	CONSOLIDATION_CD	LEDGER_STAT	CONSOLIDATION_CD	CONSOLIDATION_CD	N	Y
FCT_MGMT_REPORTING	V_ACCUMULATION_TYPE_CD	ACCUM_TYPE_CD	LEDGER_STAT	ACCUM_TYPE_CD	ACCUM_TYPE_CD	N	N
FCT_MGMT_REPORTING	N_BALANCE_TYPE_CD	BALANCE_TYPE_CD	LEDGER_STAT	BALANCE_TYPE_CD	BALANCE_TYPE_CD	N	N
FCT_MGMT_REPORTING	N_FINANCIAL_ELEM_ID	FINANCIAL_ELEM_ID	LEDGER_STAT	FINANCIAL_ELEM_ID	FINANCIAL_ELEM_ID	N	Y

- LEDGER_STAT - This table is the source for the transformation.
- DIM_<dimension Name> - The flattened dimension tables used in Business Intelligence (BI) reporting are accessed to obtain the surrogate key to be populated to FCT_MGMT_REPORTING dimension columns.

  For example, DIM_ORG_UNIT, DIM_PRODUCT, and so on.
- FCT_MGMT_REPORTING - This is the output table for the transformation.

For more details on viewing the structure of these tables, refer to *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *Erwin Data Model*.

Map New Dimensions

The dimensions which you can customize during OFSEFPA implementation are known as Custom Dimensions. Custom Dimensions can be added to target FCT_MGMT_REPORTING table by updating the configuration table FSI_BI_SETUP_TABLE. DIM_LOB and DIM_ORG_STRUCTURE are two Custom Dimension tables, for which the dimension data has to be entered manually during implementation.

> **Note:** All nodes in FCT_MGMT_REPORTING table should be part of the respective hierarchies for which the SCDs run. Otherwise, the cube will fail.

Hierarchy Configuration for LOB and Legal Entity

DIM_LOB

The hierarchy to be built for LOB (Line of Business) is a level based hierarchy. All leaf nodes of this hierarchy are mapped to a single root. The branch navigation of the hierarchy starts from root node which is the Level 1 column of the DIM_LOB table. The Leaf node has to be placed at the appropriate column between Level 1 and Level 16 of the DIM_LOB table. This step is required only if the reports are working out of ESSBASE cubes.

DIM_ORG_STRUCTURE

The hierarchy to be built for DIM_ORG_STRUCTURE is parent-child hierarchy. The parent node of the root has to be kept blank in the V_PARENT_CODE column. The hierarchy can then be built appropriately by putting a child node - parent node combination in the V_ENTITY_CODE and V_PARENT_CODE columns of the DIM_ORG_STRUCTURE table. This metadata setup supports only one hierarchy for Legal Entity.

V_LCY_CODE column needs to be entered for each legal entity. This currency column is used to calculate the amount in local currency. FSI_EXCHANGE_RATE_HIST table will then guide the conversion from base currency to local currency columns in fact table.

Follow the below steps manually to populate data into DIM_LOB and DIM_ORG_STRUCTURE custom dimensions for the fact table:

Procedure

1. Add the columns N_LOB_ID and N_ENTITY_ID as part of Unique Key (Ledger_stat) in FCT_LEDGER_STAT table.

2. Update the columns SOURCE_COLUMN_NAME and SKEY_COLUMN_NAME for the target table FCT_MGMT_REPORTING in FSI_BI_SETUP_TABLE, with the actual source and skey column values, that is N_LOB_ID and N_ENTITY_ID.

3. Set the value to 'Y' in JOIN_REQUIRED and GROUP_BY_REQUIRED columns of FCT_MGMT_REPORTING table.

Seeded entry of FSI_BI_SETUP_TABLE:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_MGMT_REPORTING	N_ENTITY_SKEY	V_ENTITY_SKEY	DIM_ORG_STRUCTURE	-1	-1	N	Y
FCT_MGMT_REPORTING	N_LOB_SKEY	V_LOB_SKEY	DIM_LOB	-1	-1	N	Y

Updated entry of FSI_BI_SETUP_TABLE:

TARGET_TABLE_NAME	TARGET_COL_NAME	MEMBER_COL_NAME	SOURCE_DIM_TABLE_NAME	SOURCE_COLUMN_NAME	SKEY_COLUMN_NAME	JOIN_REQUIRED	GROUP_BY_REQUIRED
FCT_MGMT_REPORTING	N_ENTITY_SKEY	N_ENTITY_ID	DIM_ORG_STRUCTURE	N_ENTITY_ID	N_ENTITY_SKEY	Y	Y
FCT_MGMT_REPORTING	N_LOB_SKEY	N_LOB_ID	DIM_LOB	N_LOB_ID	N_LOB_SKEY	Y	Y

Note: The columns N_ENTITY_ID and N_LOB_ID are only required to be present in LEDGER_STAT if user decides to move data of LOB and ENTITY dimensions to EFPA reporting area. The seeded entries provided during installation doesn't mandate that the both these columns need to present before the transformation is executed.

Executing the Fact Management Reporting Transformation

You can execute the function from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI.

This component for OFSEFPA 6.1 has been seeded with the Batch ID **<INFODOM>_Fact_Table_Transformation**, which can be executed from *Batch Execution* section of OFSAAI. A single batch triggers the transformations for both FCT_LEDGER_STAT and FCT_MGMT_REPORTING as separate tasks. You can execute these task individually, by excluding the other. In the Parameter List, include pstart_month, pend_month, pyears, prcy, and pre_run_flg. For example, 1,12,1994,'USD','Y'.

You can also define a new Batch and an underlying Task definition from the *Batch Maintenance* window of OFSAAI.

For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

A seeded batch **<INFODOM>_Fact_Table_Transformation – Task2** has to be executed. Parameter list has to be provided in the *Batch Maintenance* window after selecting Batch and the Task2. You can use Edit mode to provide the parameters associated with the Batch. Sample list of parameters is mentioned below.

To define a new task for a Batch definition:

1. Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

2. Click **Add** (+) button from the Task Details grid.

   The *Task Definition* window is displayed.

3. Enter the **Task ID** and **Description**.

4. Select **Run Executable** component from the drop down list.

5. Select the following from the **Dynamic Parameters** list:

   - **Datastore Type** - Select the appropriate datastore type from the list.

   - **Datastore Name** - Select the appropriate datastore name from the list.

- **IP address** - Select the IP address from the list.
- **Rule Name** - Select **FCT_MGMT_TRANSFORMATION** from the drop down list of available transformations. (This is a seeded Data Transformation which is installed as part of the OFSEFPA solution installer. If you don't see this in the list, contact Oracle support).
- **Parameter List** - Enter pstart_month, pend_month, pyears, prcy, and pre_run_flg.
    - **pStart_Month** - This parameter indicates the starting Month.
    - **pEnd_Month** - This parameter indicates the ending Month.
    - **pYears** - This is a mandatory parameter that indicates the Year value.
    - **prcy** - This indicates the reporting currency with default value '**USD**'.
    - **pRe_Run_Flg** - This is an optional parameter that indicates Re-run Flag. If value is 'Y', the existing data in the fact table will be removed and reloaded.

6. Click **Save**.

   The Task definition is saved for the selected Batch.

7. Execute the Batch.

   You can execute a Batch definition from the *Batch Execution* section of *OFSAAI Operations* module.

Fact Management Reporting Transformation can also be directly executed on the database through SQLPLUS.

Details are:

Function Name: FN_MGMT_REPORTING_TRM

Parameters: pBatch_Id, pAs_of_date, pStart_Month, pEnd_Month, pYears, pRCY, and pRe_Run_Flg

Sample Parameter Values are 'Batch1', '20091231', 1, 8, 2009, 'USD', and 'Y' respectively.

Checking the Execution Status

The Batch execution status can be monitored through *Batch Monitor* section of *OFSAAI Operations* module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S - Success

The execution log can also be accessed on the application server in the directory *$FIC_DB_HOME/log/date*, where file name will have the Batch Execution ID.

You can access the database level operations log by querying the FSI_MESSAGE_LOG table. Filter the Batch Run ID column for identifying the relevant log.

Optional Load from Ledger Class Entities for Fact Management Reporting Population

Batch Name: <Infodom>_MANAGEMENT_LEDGER

Task2: Loads data from Fsi_D_Management_Ledger to Fct_Mgmt_Reporting

Parameter to be configured:
(pStart_Month,pEnd_Month,pYears,pRCY,pRe_Run_Flg,pTable_Name)

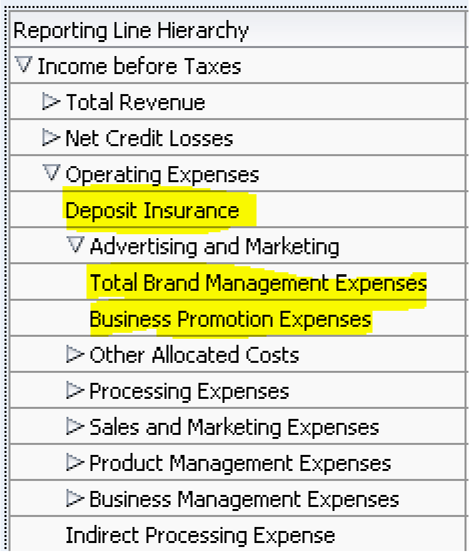
Sample Parameters Configured:
(1,12,2012,'USD','Y','FSI_D_MANAGEMENT_LEDGER')

Add the column mapping between "Ledger Class" Entity, Dimension Column and Target Column in FSI_BI_SETUP_TABLE.

Rollup Signage and Operational Signage

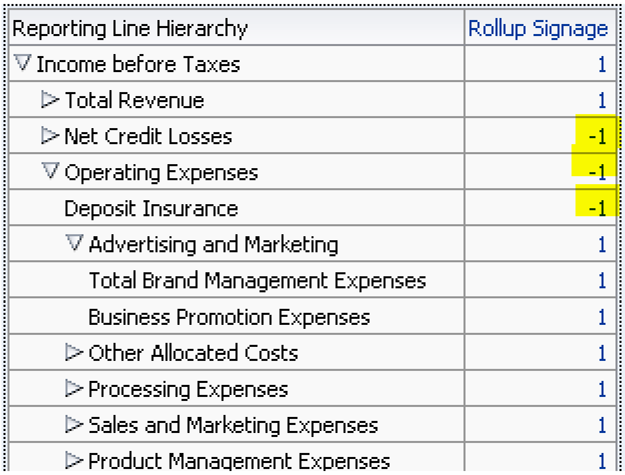
In the context of Reporting Lines, the significance of Signage is that it indicates whether the Reporting Line Value in question will be an addition or subtraction to the corresponding parent reporting line. It is important to note at this point that the reporting line values that are loaded on the Fact tables like FCT_ACCOUNT_PROFITABILITY or FCT_MGMT_REPORTING are leaf level reporting lines.

For example, consider the following hierarchy:



Reporting Line Hierarchy
▽ Income before Taxes
▷ Total Revenue
▷ Net Credit Losses
▽ Operating Expenses
Deposit Insurance
▽ Advertising and Marketing
Total Brand Management Expenses
Business Promotion Expenses
▷ Other Allocated Costs
▷ Processing Expenses
▷ Sales and Marketing Expenses
▷ Product Management Expenses
▷ Business Management Expenses
Indirect Processing Expense

The Fact table will not contain values for **Advertising and Marketing**, as that value is expected to be calculated based on the rollup of the underlying leaf level values – **Total Brand Management Expenses and Business Promotion Expenses**. However, all the underlying values will not be added together. Some of the values expected are positive and some negative. For example, the following are the leaf level values:



Reporting Line Hierarchy	Rollup Signage
▽ Income before Taxes	1
▷ Total Revenue	1
▷ Net Credit Losses	-1
▽ Operating Expenses	-1
Deposit Insurance	-1
▽ Advertising and Marketing	1
Total Brand Management Expenses	1
Business Promotion Expenses	1
▷ Other Allocated Costs	1
▷ Processing Expenses	1
▷ Sales and Marketing Expenses	1
▷ Product Management Expenses	1

Hence, when **Deposit Insurance** rollup into **Operating Expenses**, it is considered a subtraction. This type of rollup into the immediate parent is called **Rollup Signage**.

However, on rolling up further (**Income before Taxes** in the following example), the signage of **Deposit Insurance** is dependent on the rollup signage of **Operating Expenses**.

Operating Expenses = (-1) x Deposit Insurance

Income before Taxes = (-1) x Operating Expenses

Hence, when the leaf value **Deposit Insurance** rollup into **Income before Taxes**,

Income before Taxes = (-1) x (-1) x Deposit Insurance = (+1) x Deposit Insurance.

Hence, the rollup signage of **Deposit Insurance** is -1 (or negative).

However, in relation to **Income before Taxes**, the Operational Signage of **Deposit Insurance** is +1 (or positive).

The effective signage of the leaf reporting line with respect to a parent reporting line is called **Operational Signage**.

It is important to note that the Operational Signage of a reporting line is defined in relation to a parent reporting line. However, the Rollup Signage is always in relation to the immediate parent reporting line.

9

What-if Analysis

Introduction

What-if analysis functionality enables you to account for the change in profitability owing to any probable changes in the projected components of profitability. The probable change can be defined and termed as **Variation**. You can define the parameters on which the variation is applied and also the magnitude of variation. The net effect on profitability as a result of these variations can be applied.

Overview of What-if Analysis

The effect of variations on profitability can be analyzed at differing levels of granularity such as enterprise, LOB and Product. This selection is enabled through the dashboard prompt selections. The projected data of income statement is created based on the historical data available from the General Ledger. Based on the desired level of granularity at which variations are applied, aggregations are done.

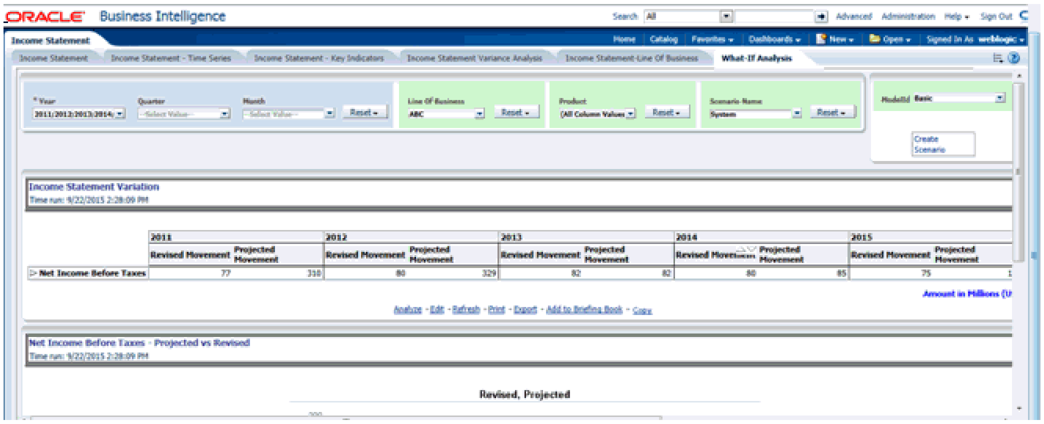
The variations can be defined through the UI. On imposing the variations on the income statement, you derive the net income. The income statement resulting after applying the variations is termed as a **Scenario**. The projections are by default created for a period of 5 years, but the change in projection may not necessarily be applied for the entire 5 years. The tenure, for which the specified variation is applicable, can also be defined while specifying the variation. You can specify the magnitude of variation in **percentage** or **absolute**. On specifying variation as percentage, the value of the component to which the variation is applied changes according to the corresponding percentage value for the specified time. Similarly, on specifying variation as absolute, the value of the component to which variation is applied changes according to the corresponding absolute value for the specified time.

Depending on the user authority, you can save a scenario, which can later be accessed for reference. The variations can be applied on the income statement in the following two methods:

- Basic: The variations applied get simply aggregated with the modified values of the components to derive the net income. The Basic version supports the variations to be applied to multiple parameters at the same time.

- Advanced: The variations applied also affect the other components it is correlated to and the modified values of all such parameters get aggregated to derive the net income. The Advanced version supports the variation to be applied to only a single component at a time.

The scenarios that are created are used to analyze the outcome on profitability of any probable change in future. Depending on the user authority, you can save a scenario, which can later be accessed for reference. It is also possible to create a scenario on an existing scenario by applying variations to the components of income statement.


ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Income Statement
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Income Statement
Income Statement - Time Series
Income Statement - Key Indicators
Income Statement Variance Analysis
Income Statement-Line Of Business
What-If Analysis
Reset
Create Scenario
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

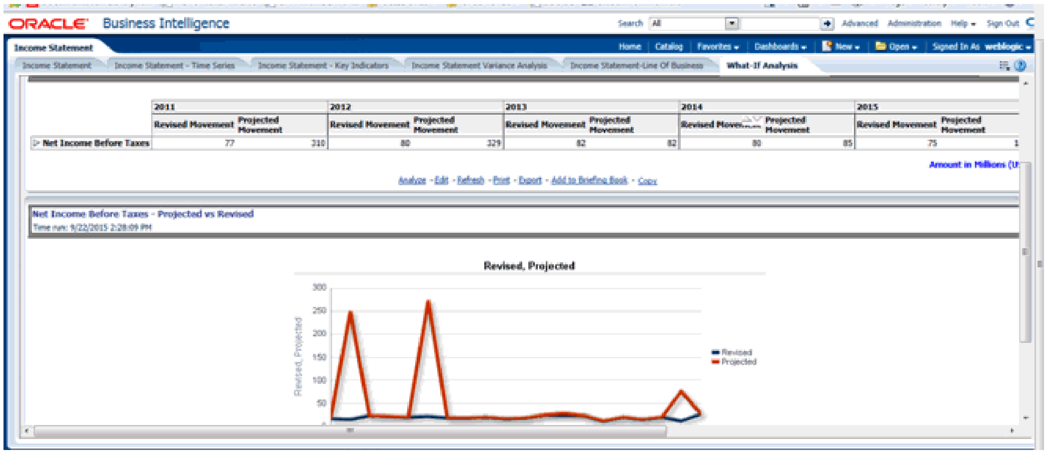
* Year: 2011;2012;2013;2014;
Quarter: --Select Value--
Month: --Select Value--
Line Of Business: ABC
Product: (All Column Values)
Scenario Name: System
ModelId: Basic

Income Statement Variation
Time run: 9/22/2015 2:28:09 PM

	2011		2012		2013		2014		2015	
	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement
▷ Net Income Before Taxes	77	310	80	329	82	82	80	85	75	1

Amount in Millions (U

Net Income Before Taxes - Projected vs Revised
Time run: 9/22/2015 2:28:09 PM

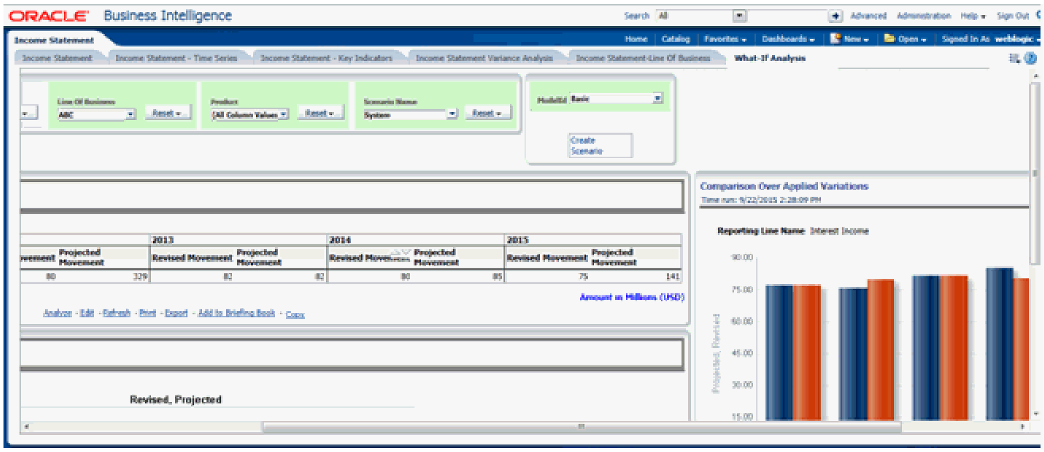

ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Income Statement
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Income Statement
Income Statement - Time Series
Income Statement - Key Indicators
Income Statement Variance Analysis
Income Statement-Line Of Business
What-If Analysis
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

	2011		2012		2013		2014		2015	
	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement
▷ Net Income Before Taxes	77	310	80	329	82	82	80	85	75	1

Amount in Millions (U

Net Income Before Taxes - Projected vs Revised
Time run: 9/22/2015 2:28:09 PM

Revised, Projected
300
250
200
150
100
50
Revised, Projected
Revised
Projected


ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Income Statement
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Income Statement
Income Statement - Time Series
Income Statement - Key Indicators
Income Statement Variance Analysis
Income Statement-Line Of Business
What-If Analysis
Reset
Create Scenario
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
Revised, Projected
Comparison Over Applied Variations
Time run: 9/22/2015 2:28:09 PM
90.00
75.00
60.00
45.00
30.00
15.00
Projected, Revised

Line Of Business: ABC

Product: All Column Values

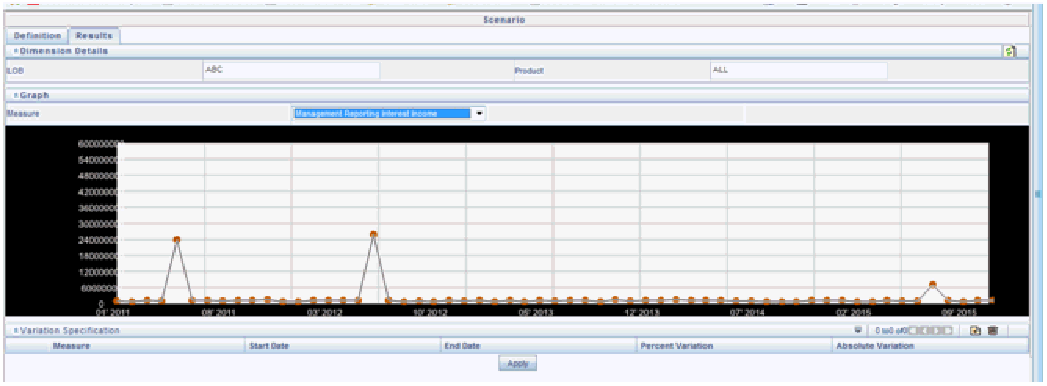
Scenario Name: System

ModelId: Basic

		2013		2014		2015	
vement	Projected Movement	Revised Movement	Projected Movement	Revised Movem...	Projected Movement	Revised Movement	Projected Movement
80	329	82	82	80	85	75	141

Amount in Millions (USD)

Reporting Line Name: Interest Income



LOB: ABC

Product: ALL

Measure: Management Reporting Interest Income

Configuration for What-If Analysis

The following list of configurations is to be performed for working on What-If Analysis

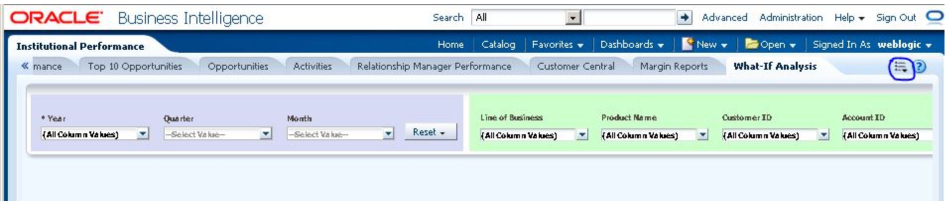
1. Configure connection to the What-If Variation application page in OFSAAI in the Create Scenario Analysis. OFSAAI is Oracle Financial Services proprietary tool, which uses Java to enable users to apply variations on the projected data.

   For example: Assume that the OFSAA hostname is 10.184.150.107 and the OBIEE analytics port is 7001, then the OBIEE analytics access url would be: *http://bank_host:8080/PFT801*.

   To configure the What-If analysis framework, perform the following steps:

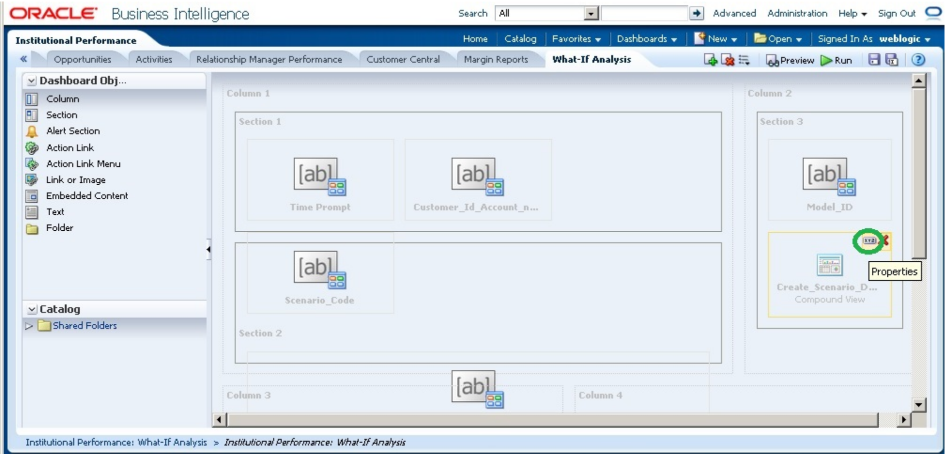
   - Navigate to What-If Analysis Dashboard Page in the Dashboard – Institutional Performance for OFSIPA, Retail Performance for OFSRPA, and Income Statement for EFPA.

   - Edit Dashboard Page.

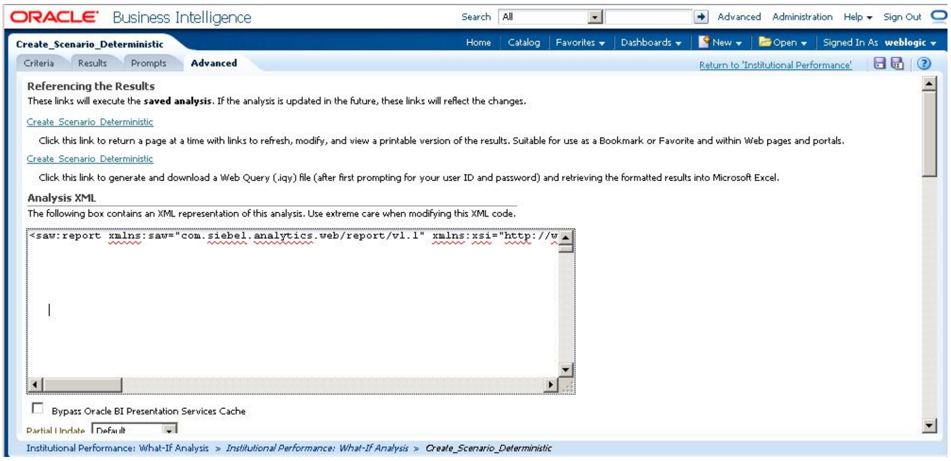

ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Institutional Performance
Home Catalog Favorites Dashboards New Open Signed In As weblogic
mance Top 10 Opportunities Opportunities Activities Relationship Manager Performance Customer Central Margin Reports What-If Analysis
* Year (All Column Values)
Quarter --Select Value--
Month --Select Value--
Reset
Line of Business (All Column Values)
Product Name (All Column Values)
Customer ID (All Column Values)
Account ID (All Column Values)

- Edit the analysis Create Scenario.

> **Note:** The OBIEE users mapped to the role of BI Author or higher will be able to configure the What-If Analysis framework.


ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Institutional Performance
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Opportunities Activities Relationship Manager Performance Customer Central Margin Reports What-If Analysis
Preview Run
Dashboard Obj...
Column
Section
Alert Section
Action Link
Action Link Menu
Link or Image
Embedded Content
Text
Folder
Catalog
Shared Folders
Column 1
Section 1
Time Prompt
Customer_Id_Account_n...
Scenario_Code
Section 2
Column 2
Section 3
Model_ID
Properties
Create_Scenario_D...
Compound View
Column 3
Column 4
Institutional Performance: What-If Analysis > Institutional Performance: What-If Analysis

- Navigate to the Advanced XML section and edit the contents of the Analysis XML



ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Create_Scenario_Deterministic
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Criteria Results Prompts Advanced
Return to 'Institutional Performance'

Referencing the Results

These links will execute the **saved analysis**. If the analysis is updated in the future, these links will reflect the changes.

Create_Scenario_Deterministic

Click this link to return a page at a time with links to refresh, modify, and view a printable version of the results. Suitable for use as a Bookmark or Favorite and within Web pages and portals.

Create_Scenario_Deterministic

Click this link to generate and download a Web Query (.iqy) file (after first prompting for your user ID and password) and retrieving the formatted results into Microsoft Excel.

Analysis XML

The following box contains an XML representation of this analysis. Use extreme care when modifying this XML code.

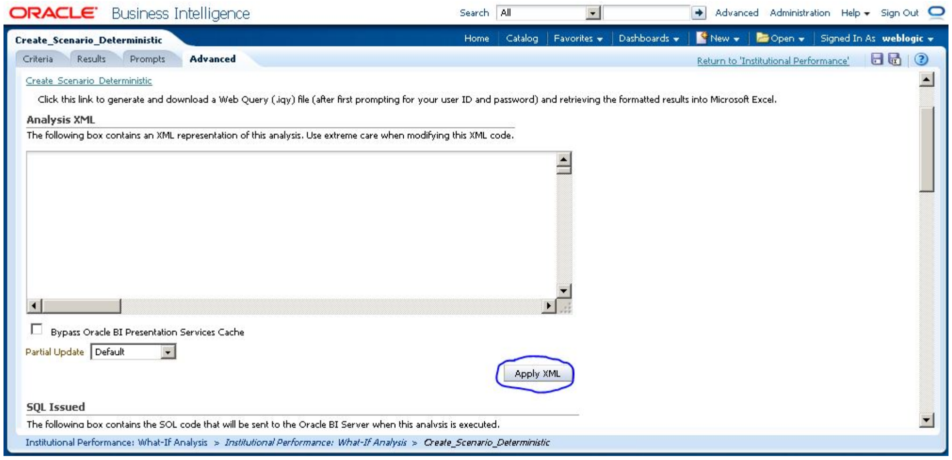
```
<saw:report xmlns:saw="com.siebel.analytics.web/report/v1.1" xmlns:xsi="http://w
```

□ Bypass Oracle BI Presentation Services Cache

Partial Update: Default

Institutional Performance: What-If Analysis > Institutional Performance: What-If Analysis > Create_Scenario_Deterministic

- Replace all occurrences as follows:
  - ##ofsaa_hostname## with the OFSAAI user hostname (For example: bank_host)
  - ##ofsaa_port## with the OFSAAI servlet port (For example: 8080)
  - ##ofsaa_context## with the context of the OFSAAI instance (For example: PFT801).
- Click Apply XML. Save the analysis after the occurrences of placeholders have been replaced and the XML contents have been pasted.



ORACLE Business Intelligence
Search All
Advanced Administration Help Sign Out
Create_Scenario_Deterministic
Home Catalog Favorites Dashboards New Open Signed In As weblogic
Criteria Results Prompts Advanced
Return to 'Institutional Performance'

Create_Scenario_Deterministic

Click this link to generate and download a Web Query (.iqy) file (after first prompting for your user ID and password) and retrieving the formatted results into Microsoft Excel.

Analysis XML

The following box contains an XML representation of this analysis. Use extreme care when modifying this XML code.

□ Bypass Oracle BI Presentation Services Cache

Partial Update: Default

SQL Issued

The following box contains the SQL code that will be sent to the Oracle BI Server when this analysis is executed.

Institutional Performance: What-If Analysis > Institutional Performance: What-If Analysis > Create_Scenario_Deterministic

2. Configure the OBIEE URL in the What-If Model Definition setup tables to navigate between the OFSAAI and OBIEE screens.

   Example : Assume that the user hostname is 10.184.150.107 and the OBIEE analytics port is 7001, then the OBIEE analytics access url would be *http://10.184.150.107:7001/analytics*.

To configure the details on the What-If analysis framework, the user needs to execute the following update on the atomic schema:

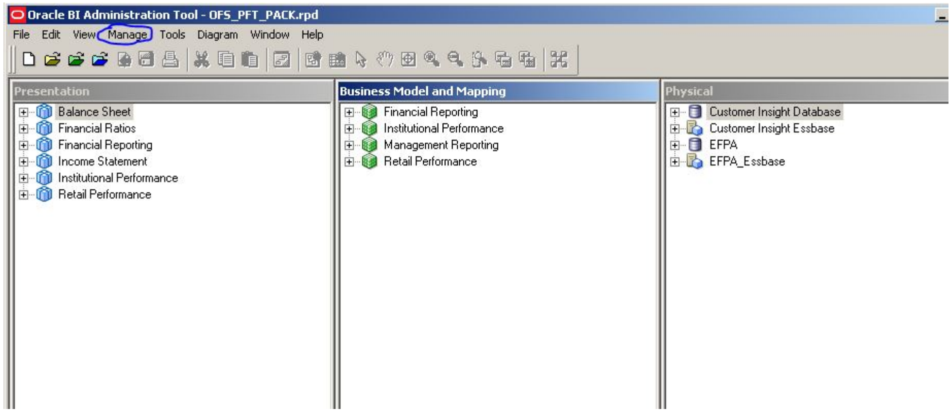
```
update fsi_m_wif_model_defn set output_page =

replace(replace(output_page,'##hostname##', '10.184.150.107' ), '##port##','7001')

/

Commit

/
```

Where the hostname and port replaced would be the user's corresponding hostname and port instead of the examples mentioned.

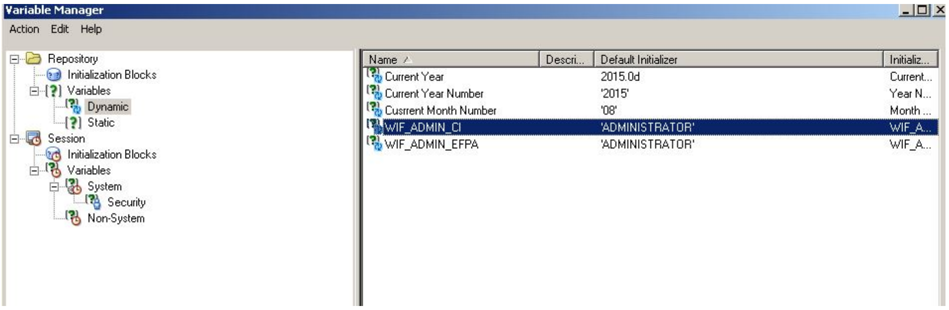
3. Configure What-If Admin Role in RPD to configure security roles to restrict Scenario Creation.

   If a user is not mapped to the role of a What-If Analysis administrator, the user will only have access to Display Results. The results of this operation will not be persisted beyond one session per user.

   - Open the OFS_PFT_PACK RPD and navigate to the Variable Definition Screen.



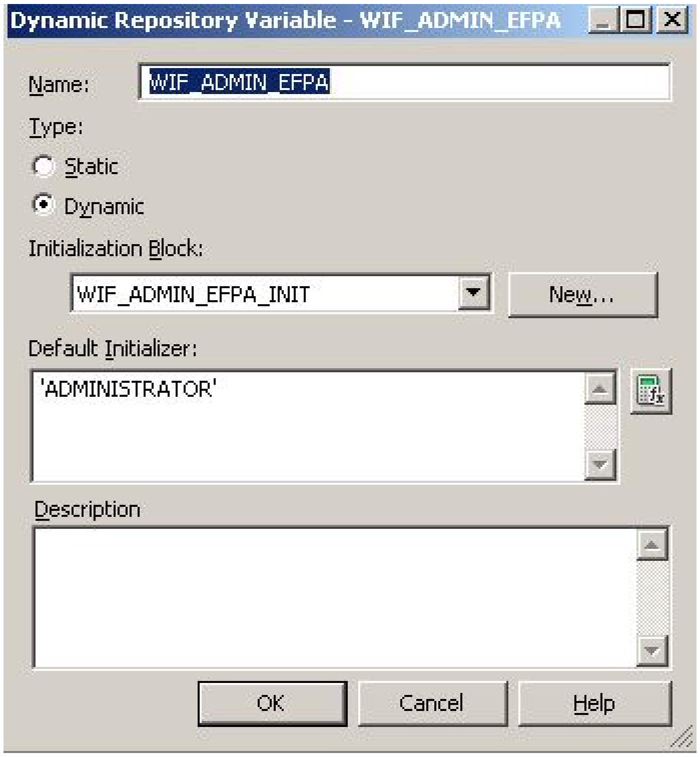
   - Modify the WIF_ADMIN_CI Repository Dynamic Variable.



Name	Descri...	Default Initializer	Initializ...
Current Year		2015.0d	Current...
Current Year Number		'2015'	Year N...
Cusrrent Month Number		'08'	Month ...
WIF_ADMIN_CI		'ADMINISTRATOR'	WIF_A...
WIF_ADMIN_EFPA		'ADMINISTRATOR'	WIF_A...

- Edit the default initializer to enter the desired What-If Administrator role.

  Note: The user mapped to the role of Administrator has the privilege to create and save a scenario. Users who are not mapped to the role of Administrator will only be able to create a scenario, but not save it.



Name: WIF_ADMIN_EFPA

Type:

○ Static

● Dynamic

Initialization Block: WIF_ADMIN_EFPA_INIT

Default Initializer: 'ADMINISTRATOR'

Description

If the web server is Tomcat of version >= 8.0.18, following additional configuration needs to be done to avoid performance issues while performing What-If Analysis.

Add the following tags in the server.xml file under tomcat_folder/conf/:

1. Insert the following tag in the Context tag as the first nested tag:

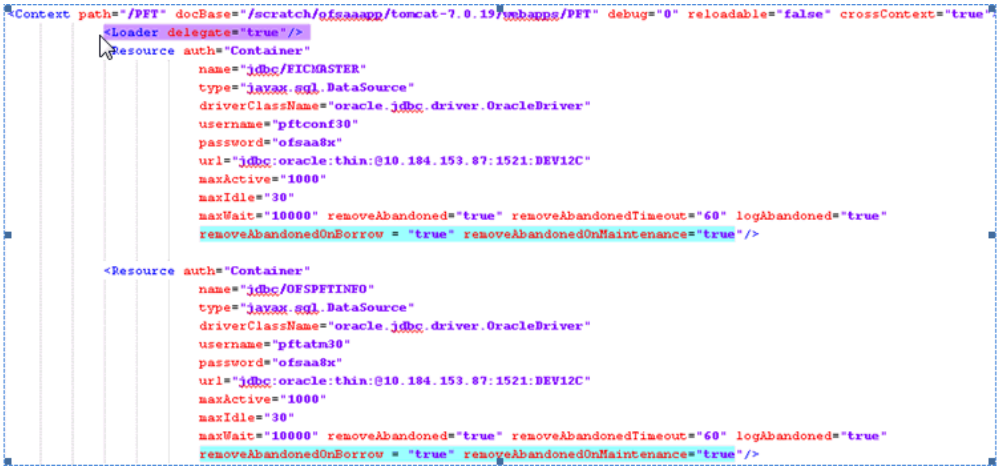
<Loader delegate="true"/>

2. Insert the following attributes for all the Resource tags in the Context tag:

   removeAbandonedOnBorrow = "true"

   removeAbandonedOnMaintenance = "true"

   Example:



```
<Context path="/PFT" docBase="/scratch/ofsaaapp/tomcat-7.0.19/webapps/PFT" debug="0" reloadable="false" crossContext="true"
            <Loader delegate="true"/>
            <Resource auth="Container"
                  name="jdbc/FICMASTER"
                  type="javax.sql.DataSource"
                  driverClassName="oracle.jdbc.driver.OracleDriver"
                  username="pftconf30"
                  password="ofsaa8x"
                  url="jdbc:oracle:thin:@10.184.153.87:1521:DEV12C"
                  maxActive="1000"
                  maxIdle="30"
                  maxWait="10000" removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
                  removeAbandonedOnBorrow = "true" removeAbandonedOnMaintenance="true"/>

            <Resource auth="Container"
                  name="jdbc/OFSPFTINFO"
                  type="javax.sql.DataSource"
                  driverClassName="oracle.jdbc.driver.OracleDriver"
                  username="pftatm30"
                  password="ofsaa8x"
                  url="jdbc:oracle:thin:@10.184.153.87:1521:DEV12C"
                  maxActive="1000"
                  maxIdle="30"
                  maxWait="10000" removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
                  removeAbandonedOnBorrow = "true" removeAbandonedOnMaintenance="true"/>
```

10

Cube Build Process

Introduction

OFSEFPA reports can be configured to work on Relational database or Essbase cubes. Source of data for the reports is determined by the priority set for each Logical Table Source (LTS) in OBIEE RPD. Multi-dimensional databases store aggregated data for better performance and provide mechanisms for performing non-additive rollup within a hierarchy and defining complex derived measures using cross-dimensional operations. OFSAA Infrastructure is used for defining metadata for cube building process. Cubes are optional source of data in OFSEFPA 8.0.0.0.0

Overview of Cubes

OFSEFPA application has the following seeded cubes:

- Cube for Financial Reporting - ADPAFNRE
    - **Purpose**: The purpose of this cube is to store data for reports belonging to Financial Reporting Dashboard.
    - **Dataset**: DSPAFR

      This cube is based on the FCT_LEDGER_STAT fact table.
- Cube for Management Reporting - ADPAMNRE
    - **Purpose**: The purpose of this cube is to store data for reports belonging to Management Reporting Dashboard.
    - **Dataset**: DSPAMR

      This cube is based on the FCT_MGMT_REPORTING fact table.

Creating Configuration Files

The metadata tables of the config schema hold the details of the cube. Also, the Measure Hierarchy details are seeded in the REV_BIHIER and the REV_LOCALE_HIER tables in the atomic schema.

Other Hierarchy data with their Measures are loaded into these REV tables, once following steps are performed:

- Individual re-save of each Hierarchy
- Individual re-save of each Dimension
- Metadata Authorization (If any Forms are to be authorized)
- Save Metadata

Each cube has a configuration file that contains the details of Dimensions and Measures which are part of the cube. Essbase outline is created using the configuration file. The configuration file is created during saving of the cube definition.

Follow the below steps:

Procedure

1. In OFSAAI, navigate to **Home > Unified Metadata Manager > Business Metadata Management > Cubes**.
2. Click **Search**.

   All the available cubes are displayed in the *Cube* popup.
3. Select the Cube name that needs to be built, and click **OK** to return to the **Cube Definition** window.
4. Click **Save** to save the cube.

   A pop up appears with a message 'Operation Successful'.

   > **Note:** Cube definition is saved only when the UI component detects any change event. In order to trigger the change event, type a blank space in 'Long Description' text-box and remove the same. Or a dimension can be removed from selected list, again the same dimension re-selected, variation applied for the dimension, and saved.

Building Of Cubes

OFSAAI Cube definition process is as follows:

- Generate an aggregate DATA file containing the measure values for each dimension leaf that are part of the Cube definition. This is performed by the AGGREGATE DATA component task within the Batch definition.
- Create the Cube outline on Essbase server. This is performed by the CREATE CUBE component task within the Batch definition.
- Load the data to the cube. This is performed by the CREATE CUBE task within the Batch definition. Data load has to happen in sequential order for the months for which data is available in fact tables.

Prerequisites

The following are prerequisites for creating a Cube:

- All the post install steps mentioned in the *Oracle Financial Services Analytical Applications Infrastructure Installation Guide* and Solution installation manual have to be completed successfully.
- Parentage files need to be created for BI Hierarchies after dimension data is loaded. 'Resave Metadata' process is used to create the parentage files.
- OFSAAI user must have the necessary function roles mapped to Resave Metadata from the **Home> Unified Metadata Manager > Business Metadata Management** screen and execute a Batch from *Batch Operations* screen.
- You can save Metadata as mentioned below:
  - Navigate to **Home > Administration > Save Metadata** section.
  - Select the available metadata under a Hierarchy and move it to the RHS pane using '>>' button.
  - Click **Save**.
  - You can view the log by clicking **Show Details** button.

  For more details, refer to *System Configuration* and *Administration* chapters in *Oracle Financial Services Analytical Applications Infrastructure User Manual*.

  Saving metadata creates all the parentage files required for building Cubes.
- Ensure that the following services are running on the application server before

doing a cube build:

  - Iccserver
  - Router
  - Messageserver
  - Olapdataserver

- Batches need to be created for executing. For more information, refer to section How to Define a Batch, page D-1.

- All the required Dimensions and Fact tables should be populated before executing the cube.

- The dataset for the cube should return some rows in the database for the cube build to happen.

  To check the same, perform the following steps:

  - Navigate to **Home > Unified Metadata Manager > Business Metadata Management > Data Sets**.
  - Click **Search**.
  - Select the dataset in the pop up and click **OK** to return to the data set screen.
  - Click the button on right of ANSI Join text box. Enter the required expression or click the **Browse** button to define an expression using the *Expression* screen.
  - Click **OK** to return to the data set screen.

  For more information, refer to *Create Expression* section in *Oracle Financial Services Analytical Applications Infrastructure User Manual*.

- Perform the same for Join/Filter Condition and Date filter.

- Frame an SQL query like this:

  ```
  SELECT COUNT(1) FROM <ENTER THE PART YOU OBTAINED FROM ANSI JOIN
  PART ABOVE>WHERE<ENTER THE PART YOU OBTAINED FROM JOIN/FILTER
  CONDITION & DATE FILTER PARTS>
  ```

  This query should show record count greater than zero when you fire this from SQL prompt in the database.

Tables Used by the Cube Build Component

Tables that are part of the dataset need to be populated before executing the cube build component. In addition, REV_BIHIER table in atomic database schema stores the hierarchy data for Business Intelligence-enabled hierarchies for cube build. This table gets populated when a hierarchy is saved using *Save Metadata* screen.

Executing the Cube Build Task

You can execute the function from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI, as mentioned below:

Define a new Batch and an underlying Task definition from the Batch Maintenance window of OFSAAI. For more information on defining a new Batch, refer to section How to Define a Batch, page D-1.

To define a new task for a Batch definition:

Procedure

Aggregate Data Component:

1. Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

2. Click **Add** (+) button from the *Task Details* grid.

   The *Task Definition* window is displayed.

3. Enter the **Task ID** and **Description**.

4. Select **Aggregate Data** component from the drop down list.

5. Select the following from the **Dynamic Parameters** list:

   - **Datastore Type** - Select the appropriate datastore type from the list.

   - **Datastore Name** - Select the appropriate datastore name from the list.

   - **IP address** - Select the IP address from the list.

   - **Cube Parameter** - Choose the cube code to be built from the drop down list.

   - **Operation** - Choose **All** from the drop down list.

6. Click Save.

   The Task definition is saved for the selected Batch.

Create Cube Component:

1. Select the check box adjacent to the newly created Batch Name in the *Batch Maintenance* window.

2. Click **Add** (+) button from the *Task Details* grid.

   The *Task Definition* window is displayed.

3. Enter the **Task ID** and **Description**.

4. Select **Create Cube** component from the drop down list.

5. Select the following from the **Dynamic Parameters** list:

   - **Datastore Type** - Select the appropriate datastore type from the list.
   - **Datastore Name** - Select the appropriate datastore name from the list.
   - **IP address** - Select the IP address from the list.
   - **Cube Parameter** - Choose the cube code to be built from the drop down list.
   - **Operation** - Choose **All** from the drop down list.

6. Click Save.

   The Task definition is saved for the selected Batch.

7. Execute the Batch.

   You can execute a Batch definition from the *Batch Execution* section of *OFSAAI Operations* module.

   > **Note:** A common issue in the Aggregate task is, Data Set not having records. You can check this as mentioned in the Prerequisites or by executing the SQL query in the Data Cruncher log file available in the path *$FIC_DB_HOME/log/dc*.
   >
   > In the Create Cube task one common error is the hierarchy member being the same for two different dimensions which are part of the same cube (Error message: 'Duplicate Alias' in the Create Cube log file). In this case, you can append a string to the Hierarchy member code to make it unique across the Cube or change the hierarchy data to make the node unique across the cube.

Points to be considered before executing the batches for the cubes **<INFODOM>_Financial_Reporting_Cube** and **<INFODOM>_Management_Reporting_Cube**:

- Populate dummy values against NULL (wherever applicable) for Level IDs and Level Names in the underlying DIM tables for the Hierarchies Product, Line of Business, and Org Unit.

For the successful execution of the cube, all the levels (Level IDs and Level Names used in the Hierarchy) having null in the underlying Dimension tables along with the level that holds the leaf should be loaded with some dummy values (with proper hierarchical structure followed) for these three Hierarchies. A new Level, LEVEL00 has been introduced which holds the leaf nodes for these hierarchies. The Hierarchy should always be balanced. Members across dimensions should be distinct in Code and Names.

For instance, if the leaf is present at LEVEL18 for a product, then LEVEL06 to LEVEL17 will have null values in Level IDs and Level Names after SCD. These null values should be updated to some dummy values, with a proper hierarchical structure. The Level ID and Level Name of LEVEL18 should also be updated along with this. The leaf node will be depicted by LEVEL00. A particular level value should not be repeated across Level IDs, Level Names, v_prod_code, v_product_name, and n_product_id in a row.

DIM_PRODUCT has to be updated in this manner for only those product skeys that are part of the Fact table for which the cube is executed. This method should be applied to DIM_ORG_UNIT and DIM_LOB as well.

DIM_PRODUCT and DIM_ORG_UNIT have a similar Hierarchical Level structure with root being LEVEL20. But for DIM_LOB, the root is LEVEL1. This difference should be kept in mind while populating the dummy values.

- Levels that are provided for DIM_PRODUCT, DIM_ORG_UNIT and DIM_LOB are as follows:
  - DIM_PRODUCT: Levels will be defined from Level_20 to Level_6 and Level_0 will be the leaf level. In CUBE, Dimension and Hierarchy levels get added by default. All together constituting 16 levels.
  - DIM_ORG_UNIT: Levels will be defined from Level_20 to Level_6 and Level_0 will be the leaf level. In CUBE, Dimension and Hierarchy levels get added by default. All together constituting 16 levels.
  - DIM_LOB: Levels will be defined from Level_1 to Level_16 and Level_0 will be the leaf level. In CUBE, Dimension and Hierarchy levels get added by default. All together constituting 17 levels.
- Month data of cubes should be populated in proper order, else the AGO function in OBIEE will not give correct results.
  - Preference of reports is RDBMS and not cubes.

For more information on Cubes, refer to *Cubes* section under *Unified Metadata Manager* chapter in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Seeded batches are provided along with the Enterprise Financial Performance

Analytics application installer. The below are the OFSEFPA seeded batches:

- Cube for Financial Reporting

  Seeded batch <INFODOM>_Financial_Reporting_Cube is provided with the installer. Execute the batch for the required MIS Date.

- Cube for Management Reporting

  Seeded batch <INFODOM>_Management_Reporting_Cube is provided with the installer. Execute the batch for the required MIS Date.

  > **Note:** "What-if Analysis" report of Management Reporting Income Statement will work out of RDBMS Data source.

a. Bug 16504664 - CUBES: NOT ABLE TO SAVE HEPMGLFR (REPORTING LINE FOR FR) HIERARCHY. Workaround – Execute the scripts available in the integrated SQL file below post SCD run.

DIM_GL_ACCOUNT_update_EFPA.sql

Seeded batches are provided along with the Enterprise Financial Performance Analytics application installer. The below are the OFSEFPA seeded batches:

- Cube for Financial Reporting

  Seeded batch **<INFODOM> _Financial_Reporting_Cube** is provided with the installer. Execute the batch for the required MIS Date.

- Cube for Management Reporting

  Seeded batch **<INFODOM> _Management_Reporting_Cube** is provided with the installer. Execute the batch for the required MIS Date.

Checking the Execution Status

The Batch execution status can be monitored through Batch Monitor section of OFSAAI Operations module.

The status messages in batch monitor are:

N - Not Started

O - On Going

F - Failure

S – Success

The execution log can be accessed on the application server in the directory *$FIC_DB_HOME/log/dc* for the Task 1 above (Aggregate Data). The file name will have the Batch Execution ID.

The execution log can be accessed on the application server by going to the following directory *$FIC_DB_HOME/log/olap* for the Task 2 above (Create Cube). The file name will have the Batch Execution ID.

> **Note:** Refer to section How to Create a New Cube, page D-1 on how to add a New cube or modifying existing ones. For any new cube added using the OFSAAI Cube screen, the tasks for execution are the same as mentioned in this section.

11

Predictive Modelling

Introduction to Predictive Modelling

What-If analysis reports use the reporting line forecast values that are generated using the Arima Algorithm in the R code seeded with the application. R has a base package called 'stats' which provides the function called as 'arima()'. This function enables the usage of ARIMA technique for time series forecasting.

Guidelines

The guidelines associated with respect to the execution R code are mentioned below:

- Data should be generated for at least one group for more than 12 continuous mis dates. 12 is the parameter nConsider where we are setting how many records is significant to be considered for prediction.

- Assumption is that the data is chronological for consecutive end of month dates. By default prediction will be done for 60 months starting with the immediate month after the last available MIS Date.

- If the data provided is not for chronological end of month dates, results generated will not be accurate.

- ARIMA is a statistical technique used for time series predictions. It accepts a host of parameters of which the basic parameters are p,d and q. p is the order of Autoregressive Process, q is the parameter for Moving Average process and d is the number of differences after which the data can be considered stationary with a desired confidence level. It has more parameters that can be customized.

- Detailed documentation of the technique can be found at : http://www.dms.umontreal.ca/~duchesne/chap7.pdf

- Documentation of implementation of ARIMA in R can be found at :

http://stat.ethz.ch/R-manual/R-patched/library/stats/html/arima.html

Logging for the individual groups' arima model summaries will happen in a file named "out.log". This log file will be generated in $FIC_DB_HOME/bin folder. If a particular group has unstable data and prediction fails, corresponding error will also be documented in the out.log file against that particular group.

Files Used

There are two files that are required to R script execution. Both the files are present at $FIC_DB_HOME/bin folder. Provide execute permissions to the following files:

1. RExec executable
2. EFPA_ARIMA_MR_IS_PV.r

Errors

Following are the errors:

1. Logging for the individual groups' arima model summaries will happen in a file named "out.log". This log file will be generated in $FIC_DB_HOME/bin folder. If a particular group has unstable data and prediction fails, corresponding error will also be documented in the out.log file against that particular group.

2. Error : Error in if ((lv > nr) || (lv == 0L && nr > 0L) || (nr%%lv != 0L)) stop(gettextf("replacement data has %s rows, data has %s", :

   Missing value where TRUE/FALSE needed means an if condition is receiving null and is unable to evaluate true or false

12

Overview of OFSEFPA Reports

Introduction to Dashboards

OFSEFPA offers the following dashboards that organize different kinds of reports by subject area.

- Financial Reporting
  - Financial Reporting
- Management Reporting
  - Balance Sheet
  - Income Statement
  - Key Trends
  - Performance Measures

Tabular Consolidated List of OFSEFPA Reports

Data Source	Business Intent	Dashboard Name	Page Name	Report Name
Initial loads to the management ledger (i.e. exclusive of any data generated by allocation rules and/or transfer pricing rules). Reporting line is General Ledger Account.	Represent Income Statement reporting lines across Legal Entity and Time dimension	Financial Reporting	Income Statement	Income Statement
		Financial Reporting	Income Statement-Time Series	Income Statement-Time Series
		Financial Reporting	Income Statement Variance Analysis	Income Statement Variance Analysis
Data from Fact Management table. Reporting line is Reporting Line Dimension.	Represent balance sheet and show balances across dimensions	Balance Sheet	Balance Sheet	Balance Sheet
		Balance Sheet	Balance Sheet-Time Series	Balance Sheet-Time Series
		Balance Sheet	Balance Sheet - Key Indicators	Total Assets
		Balance Sheet	Balance Sheet - Key Indicators	Total Liabilities

Data Source	Business Intent	Dashboard Name	Page Name	Report Name
		Balance Sheet	Balance Sheet - Key Indicators	Capital
		Balance Sheet	Balance Sheet - Key Indicators	Total Customer Assets
		Balance Sheet	Balance Sheet - Key Indicators	Total Customer Liabilities
Data from Fact Management table. Reporting line is Reporting Line Dimension.	Represent Income Statement reporting lines across dimensions	Income Statement	Income Statement	Income Statement
		Income Statement	Income Statement - Time Series	Income Statement - Time Series
		Income Statement	Income Statement - Key Indicators	Net Interest Income
		Income Statement	Income Statement - Key Indicators	Commission
		Income Statement	Income Statement - Key Indicators	Fees
		Income Statement	Income Statement - Key Indicators	Net Credit Loss

Data Source	Business Intent	Dashboard Name	Page Name	Report Name
		Income Statement	Income Statement Variance Analysis	Income Statement Variance Analysis
		Income Statement	Income Statement-Line Of Business	Income Statement across Line Of Business
		Income Statement	Income Statement-Line Of Business	Key Statistics from Balance Sheet
		Income Statement	Income Statement-Line Of Business	No. of Customers
		Income Statement	Income Statement-Line Of Business	Other Key Indicators
Data from Fact Management table. Reporting line is Reporting Line Dimension.	Show trends for balance and movement measures for key line items relevant to the business.	Key Trends	Key Trends	Break-up of Total Assets
		Key Trends	Key Trends	Break-up of Total Liabilities
		Key Trends	Key Trends	Break-up of Total Assets by Line of Business

Data Source	Business Intent	Dashboard Name	Page Name	Report Name
		Key Trends	Key Trends	Break-up of Total Liabilities by Line of Business
		Key Trends	Key Trends	NIM Speedometer
		Key Trends	Key Trends	Balance Sheet
		Key Trends	Key Trends	Income Statement
		Key Trends	Net Loss Analysis	Net Loss Rate Time Series
		Key Trends	Net Loss Analysis	Net Loss Across Line of Business
		Key Trends	Net Loss Analysis	Net Loss Across Products
		Key Trends	Net Loss Analysis	Net Loss Rate Across Scenarios
		Key Trends	NII Impact Analysis	NII at a Glance
		Key Trends	NII Impact Analysis	Impact of FX Fluctuation
		Key Trends	NII Impact Analysis	Impact of Volume Movement

Data Source	Business Intent	Dashboard Name	Page Name	Report Name
		Key Trends	NII Impact Analysis	Impact of Yield & Margin Rates
		Key Trends	Balance Trends	Details By Product
		Key Trends	Balance Trends	Top 10 Products - Balance
		Key Trends	Balance Trends	Top 10 Products - MOM Variance
Data from Fact Management table. Reporting line is Reporting Line Dimension.	Show key performance measures relevant to the business.	Performance Measures	Performance Measures	Return on Total Assets
		Performance Measures	Performance Measures	Return on Equity
		Performance Measures	Performance Measures	RAROC
		Performance Measures	Performance Measures	Profitability by Product over Reporting Period
		Performance Measures	Capital Analysis	Capital Analysis

Structure of Reporting Line Items

Reporting line items describe the financial meaning of the report.

The following spreadsheet contains the Management versus Financial Reporting Line Item Definitions.

Hard-Coded Dimension Members

The node level reporting lines (including higher level nodes) of the reporting line hierarchy that are used in the OBIEE components (Web Catalog and RPD) are mentioned below for reference.

These reporting line codes should not be modified.

Reporting Line Dimensions

Catalog/RPD	Repline Code	Repline Name
catalog	107001	Management Reporting
catalog	107002	Income Statement
catalog	107003	Balance Sheet
catalog	98000	Net Income Before Taxes
catalog	51000	Cash
catalog	54000	Trading Assets
catalog	61000	Investments
catalog	63000	Fixed Assets
catalog	64000	Other Assets
catalog	60000	Loans '&' Advances to Customers
catalog	72000	Total Liabilities

Catalog/RPD	Repline Code	Repline Name
catalog	74900	Allocated Capital
catalog	77000	Shareholders Equity
catalog	72000	Total Liabilities
catalog	75000	Share Capital
catalog	76000	Reserves
catalog	60000	Loans '&' Advances to Customers
catalog	68000	Customer Liabilities
catalog	85000	Net Interest Revenue
catalog	85100	Commission
catalog	85200	Fees
catalog	91910	Credit Losses
catalog	91920	Recoveries of amounts previously written-off
RPD	95000	Operating Income before Taxes
RPD	99500	Provisions for Credit Losses
RPD	91910	Credit Losses

Catalog/RPD	Repline Code	Repline Name
RPD	107100	Number of Customers
RPD	52000	Interest-bearing Assets
RPD	74900	Allocated Capital
RPD	88000	Non-Interest Revenue
RPD	107120	Number of Closed Customers
RPD	83000	Credit for Float
RPD	82100	Interest Income
RPD	91920	Recoveries of amounts previously written-off
RPD	107300	Attrition Rate
RPD	60000	Loans '&' Advances to Customers
RPD	85000	Net Interest Revenue
RPD	83200	Interest Expenses

Other Dimensions

- Run Dimension is hard-coded with a value zero.
- Legal Entity and Line of Business are hard-coded to -1 in the initial load. When the custom dimensions are added into LEDGER_STAT and FACT tables, the corresponding values are used in the reports.

Reporting Currency Calculation

The base currency values are converted into reporting currency and are populated to RCY columns of Fact table using FSI_EXCHANGE_RATE_HIST table. In FSI_EXCHANGE_RATE_HIST table data should exist for the available currency combinations for the particular FIC_MIS_DATE column, for which user is running the fact transformations, that is FCT_LEDGER_STAT and FCT_MGMT_REPORTING tables.

Currency Type: OFSEFPA BI reports work for Local, Base and Reporting currencies.

OBIEE: Catalogs Related Configuration

This section explains the necessary configurations for OBIEE with respect to catalogs.

Currency Label Modification

The currency label in the catalogs has to be configured in such a way that the amounts are displayed in a single currency unit. For Example: Amount in $ millions.

Follow the below steps to configure the currency label:

Procedure

1. Open RPD in online mode.

2. In the **Manage** menu, click **Variables** option.

3. Change the value of **RV_Reporting_Currency** variable as required. For example, " **Amount in Millions (USD)**".

4. Set RV_Base_Currency repository variable to the appropriate Base currency.

5. Update the connection details of **OFSEFPA Connection Pool** and **OFSEFPA Init Block** in Physical Layer of RPD, to view the Local Currency label.

6. Save and check the global consistency of the RPD.

Amount Unit Modification

Follow the below steps to configure the amount unit:

Procedure

1. Open the RPD in online mode.

2. In the **Manage** menu, click **Variables** option.

3. Change the value of **ABBREVIATE_VALUE** variable as required. For example: 1000000

4. Save and check the global consistency of the RPD.

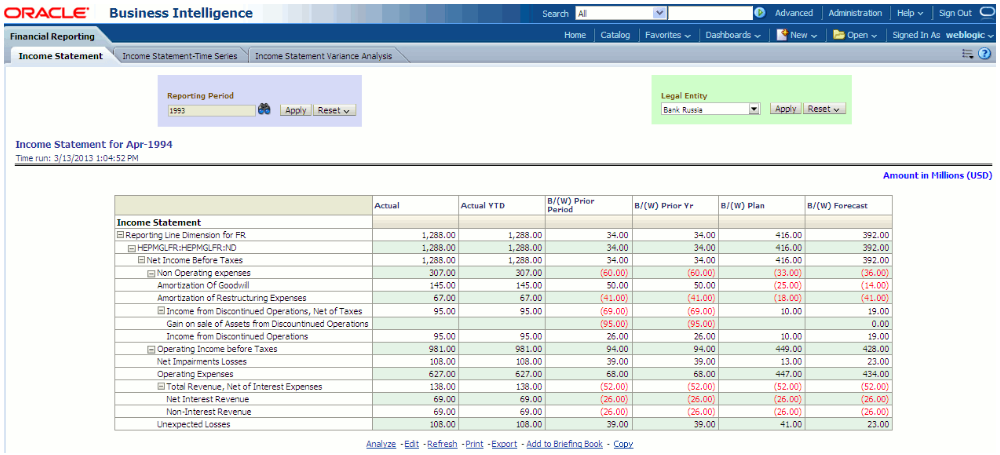
Dashboards

The following sections provide the dashboard screens that demonstrate the OFSEFPA reports.

Financial Reporting

The business intent of the Financial Reporting dashboard is to analyze the General Ledger Account based Income Statement and Balance Sheet reporting. "Currency Type" prompt allows data to be viewed for 'Reporting Currency' and 'Base Currency' for Financial Reporting reports.

- Financial Reporting - Income Statement


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Financial Reporting
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Income Statement
Income Statement-Time Series
Income Statement Variance Analysis
Apply
Reset
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: 1993

Legal Entity: Bank Russia

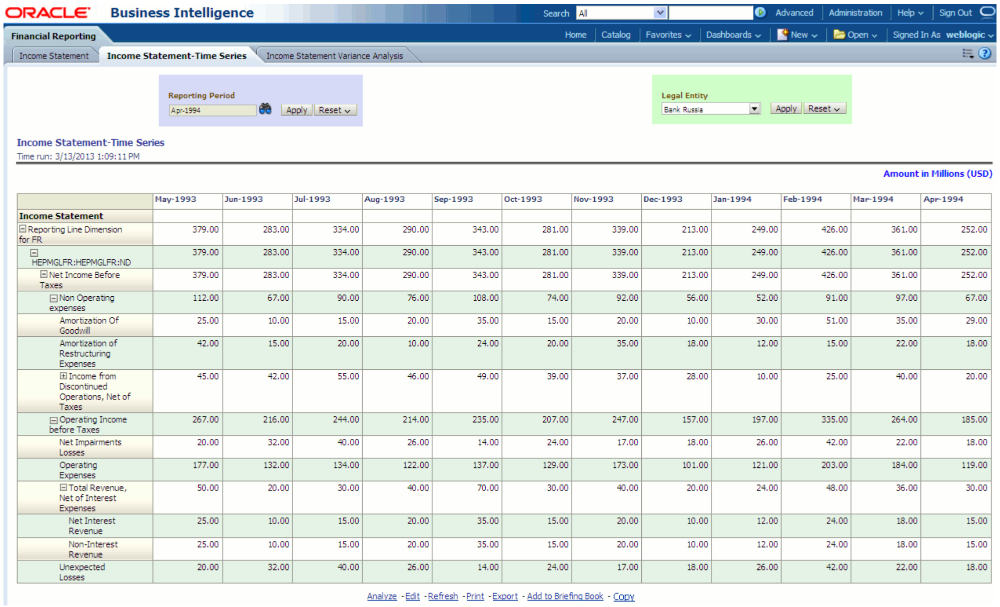
Income Statement for Apr-1994

Time run: 3/13/2013 1:04:52 PM

Amount in Millions (USD)

	Actual	Actual YTD	B/(W) Prior Period	B/(W) Prior Yr	B/(W) Plan	B/(W) Forecast
Income Statement						
⊟ Reporting Line Dimension for FR	1,288.00	1,288.00	34.00	34.00	416.00	392.00
⊟ HEPMGLFR:HEPMGLFR:ND	1,288.00	1,288.00	34.00	34.00	416.00	392.00
⊟ Net Income Before Taxes	1,288.00	1,288.00	34.00	34.00	416.00	392.00
⊟ Non Operating expenses	307.00	307.00	(60.00)	(60.00)	(33.00)	(36.00)
Amortization Of Goodwill	145.00	145.00	50.00	50.00	(25.00)	(14.00)
Amortization of Restructuring Expenses	67.00	67.00	(41.00)	(41.00)	(18.00)	(41.00)
⊟ Income from Discontinued Operations, Net of Taxes	95.00	95.00	(69.00)	(69.00)	10.00	19.00
Gain on sale of Assets from Discountinued Operations			(95.00)	(95.00)		0.00
Income from Discontinued Operations	95.00	95.00	26.00	26.00	10.00	19.00
⊟ Operating Income before Taxes	981.00	981.00	94.00	94.00	449.00	428.00
Net Impairments Losses	108.00	108.00	39.00	39.00	13.00	23.00
Operating Expenses	627.00	627.00	68.00	68.00	447.00	434.00
⊟ Total Revenue, Net of Interest Expenses	138.00	138.00	(52.00)	(52.00)	(52.00)	(52.00)
Net Interest Revenue	69.00	69.00	(26.00)	(26.00)	(26.00)	(26.00)
Non-Interest Revenue	69.00	69.00	(26.00)	(26.00)	(26.00)	(26.00)
Unexpected Losses	108.00	108.00	39.00	39.00	41.00	23.00

- Financial Reporting - Income Statement - Time Series


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Financial Reporting
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Income Statement
Income Statement-Time Series
Income Statement Variance Analysis
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: Apr-1994

Legal Entity: Bank Russia

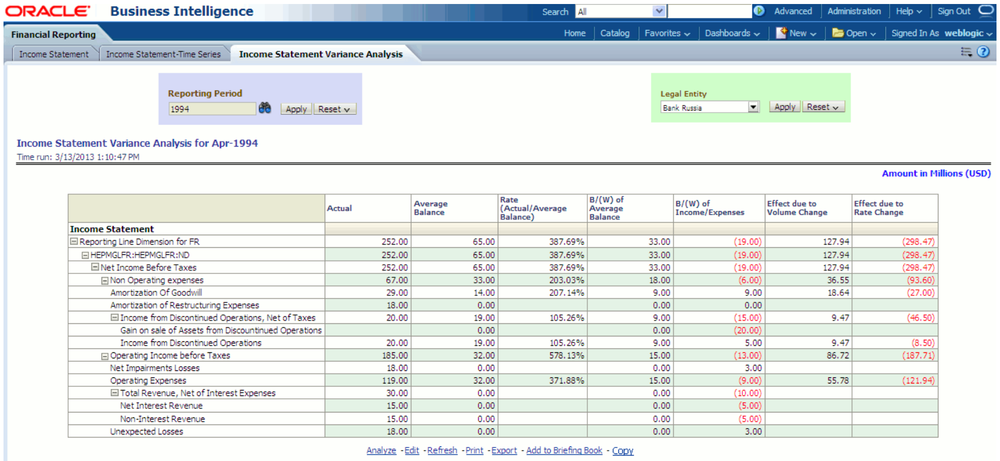
Income Statement-Time Series

Time run: 3/13/2013 1:09:11 PM

Amount in Millions (USD)

Income Statement	May-1993	Jun-1993	Jul-1993	Aug-1993	Sep-1993	Oct-1993	Nov-1993	Dec-1993	Jan-1994	Feb-1994	Mar-1994	Apr-1994
⊟ Reporting Line Dimension for FR	379.00	283.00	334.00	290.00	343.00	281.00	339.00	213.00	249.00	426.00	361.00	252.00
⊟ HEPMGLFR:HEPMGLFR:ND	379.00	283.00	334.00	290.00	343.00	281.00	339.00	213.00	249.00	426.00	361.00	252.00
⊟ Net Income Before Taxes	379.00	283.00	334.00	290.00	343.00	281.00	339.00	213.00	249.00	426.00	361.00	252.00
⊟ Non Operating expenses	112.00	67.00	90.00	76.00	108.00	74.00	92.00	56.00	52.00	91.00	97.00	67.00
Amortization Of Goodwill	25.00	10.00	15.00	20.00	35.00	15.00	20.00	10.00	30.00	51.00	35.00	29.00
Amortization of Restructuring Expenses	42.00	15.00	20.00	10.00	24.00	20.00	35.00	18.00	12.00	15.00	22.00	18.00
⊞ Income from Discontinued Operations, Net of Taxes	45.00	42.00	55.00	46.00	49.00	39.00	37.00	28.00	10.00	25.00	40.00	20.00
⊟ Operating Income before Taxes	267.00	216.00	244.00	214.00	235.00	207.00	247.00	157.00	197.00	335.00	264.00	185.00
Net Impairments Losses	20.00	32.00	40.00	26.00	14.00	24.00	17.00	18.00	26.00	42.00	22.00	18.00
Operating Expenses	177.00	132.00	134.00	122.00	137.00	129.00	173.00	101.00	121.00	203.00	184.00	119.00
⊟ Total Revenue, Net of Interest Expenses	50.00	20.00	30.00	40.00	70.00	30.00	40.00	20.00	24.00	48.00	36.00	30.00
Net Interest Revenue	25.00	10.00	15.00	20.00	35.00	15.00	20.00	10.00	12.00	24.00	18.00	15.00
Non-Interest Revenue	25.00	10.00	15.00	20.00	35.00	15.00	20.00	10.00	12.00	24.00	18.00	15.00
Unexpected Losses	20.00	32.00	40.00	26.00	14.00	24.00	17.00	18.00	26.00	42.00	22.00	18.00

- Financial Reporting - Income Statement Variance Analysis


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Financial Reporting
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Income Statement
Income Statement-Time Series
Income Statement Variance Analysis
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: 1994

Legal Entity: Bank Russia

Income Statement Variance Analysis for Apr-1994

Time run: 3/13/2013 1:10:47 PM

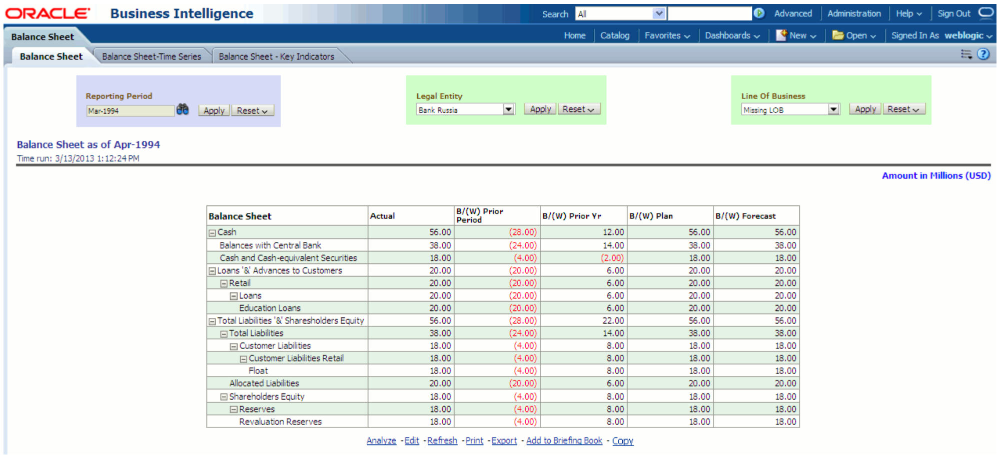
Amount in Millions (USD)

Income Statement	Actual	Average Balance	Rate (Actual/Average Balance)	B/(W) of Average Balance	B/(W) of Income/Expenses	Effect due to Volume Change	Effect due to Rate Change
⊟ Reporting Line Dimension for FR	252.00	65.00	387.69%	33.00	(19.00)	127.94	(298.47)
⊟ HEPMGLFR:HEPMGLFR:ND	252.00	65.00	387.69%	33.00	(19.00)	127.94	(298.47)
⊟ Net Income Before Taxes	252.00	65.00	387.69%	33.00	(19.00)	127.94	(298.47)
⊟ Non Operating expenses	67.00	33.00	203.03%	18.00	(6.00)	36.55	(93.60)
Amortization Of Goodwill	29.00	14.00	207.14%	9.00	9.00	18.64	(27.00)
Amortization of Restructuring Expenses	18.00	0.00		0.00	0.00		
⊟ Income from Discontinued Operations, Net of Taxes	20.00	19.00	105.26%	9.00	(15.00)	9.47	(46.50)
Gain on sale of Assets from Discountinued Operations		0.00		0.00	(20.00)		
Income from Discontinued Operations	20.00	19.00	105.26%	9.00	5.00	9.47	(8.50)
⊟ Operating Income before Taxes	185.00	32.00	578.13%	15.00	(13.00)	86.72	(187.71)
Net Impairments Losses	18.00	0.00		0.00	3.00		
Operating Expenses	119.00	32.00	371.88%	15.00	(9.00)	55.78	(121.94)
⊟ Total Revenue, Net of Interest Expenses	30.00	0.00		0.00	(10.00)		
Net Interest Revenue	15.00	0.00		0.00	(5.00)		
Non-Interest Revenue	15.00	0.00		0.00	(5.00)		
Unexpected Losses	18.00	0.00		0.00	3.00		

Management Reporting

The business intent of the Management Reporting dashboard is to analyze the Financial Element based on Income Statement and Balance Sheet reporting (management). "Currency Type" prompt allows data to be viewed for 'Reporting Currency', 'Local Currency' and 'Base Currency' for Management Reporting reports.

- Management Reporting - Balance Sheet


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Balance Sheet
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Balance Sheet
Balance Sheet-Time Series
Balance Sheet - Key Indicators
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: Mar-1994

Legal Entity: Bank Russia

Line Of Business: Missing LOB

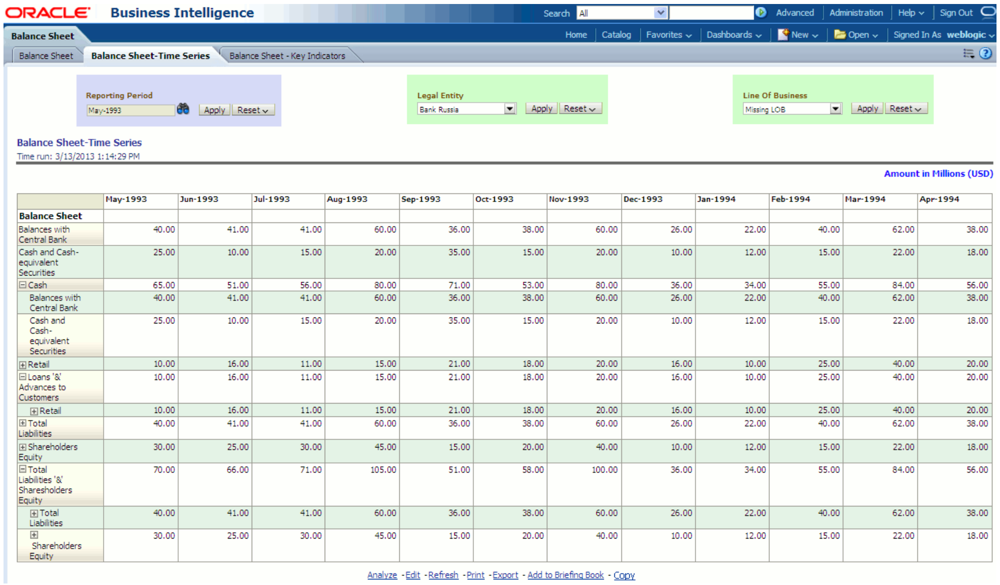
Balance Sheet as of Apr-1994

Time run: 3/13/2013 1:12:24 PM

Amount in Millions (USD)

Balance Sheet	Actual	B/(W) Prior Period	B/(W) Prior Yr	B/(W) Plan	B/(W) Forecast
⊟ Cash	56.00	(28.00)	12.00	56.00	56.00
Balances with Central Bank	38.00	(24.00)	14.00	38.00	38.00
Cash and Cash-equivalent Securities	18.00	(4.00)	(2.00)	18.00	18.00
⊟ Loans '&' Advances to Customers	20.00	(20.00)	6.00	20.00	20.00
⊟ Retail	20.00	(20.00)	6.00	20.00	20.00
⊟ Loans	20.00	(20.00)	6.00	20.00	20.00
Education Loans	20.00	(20.00)	6.00	20.00	20.00
⊟ Total Liabilities '&' Sharesholders Equity	56.00	(28.00)	22.00	56.00	56.00
⊟ Total Liabilities	38.00	(24.00)	14.00	38.00	38.00
⊟ Customer Liabilities	18.00	(4.00)	8.00	18.00	18.00
⊟ Customer Liabilities Retail	18.00	(4.00)	8.00	18.00	18.00
Float	18.00	(4.00)	8.00	18.00	18.00
Allocated Liabilities	20.00	(20.00)	6.00	20.00	20.00
⊟ Shareholders Equity	18.00	(4.00)	8.00	18.00	18.00
⊟ Reserves	18.00	(4.00)	8.00	18.00	18.00
Revaluation Reserves	18.00	(4.00)	8.00	18.00	18.00

- Management Reporting - Balance Sheet-Time Series


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Balance Sheet
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Balance Sheet
Balance Sheet-Time Series
Balance Sheet - Key Indicators
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: May-1993

Legal Entity: Bank Russia

Line Of Business: Missing LOB

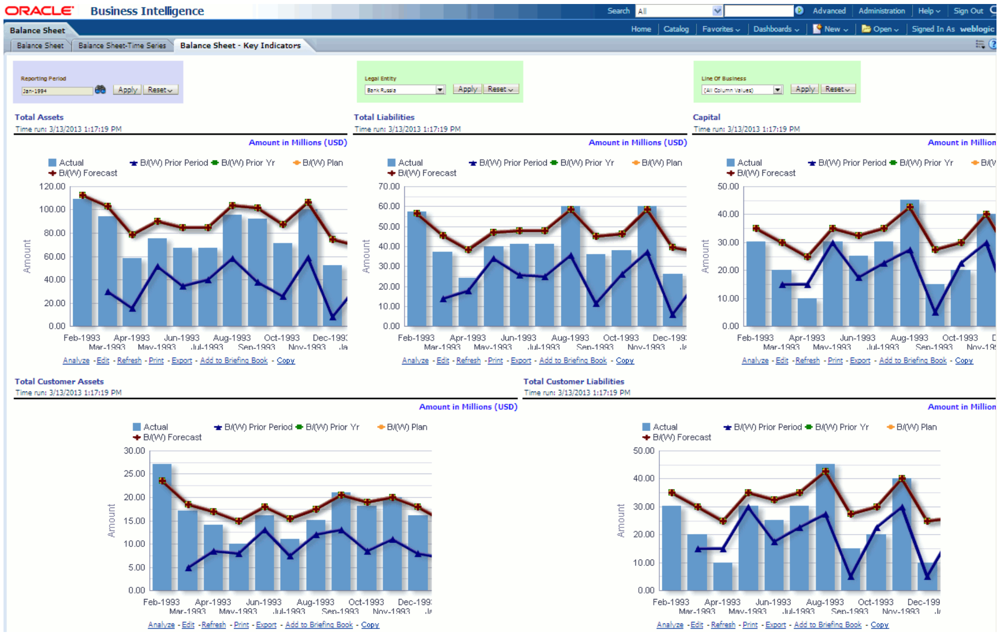
Balance Sheet-Time Series

Time run: 3/13/2013 1:14:29 PM

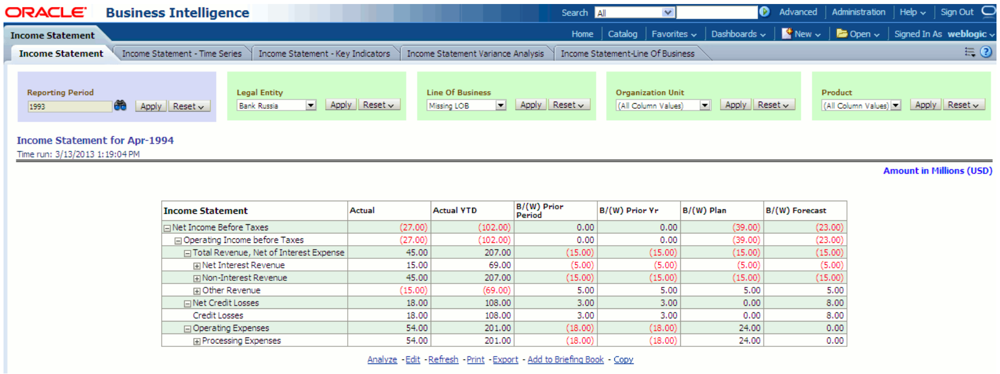
Amount in Millions (USD)

	May-1993	Jun-1993	Jul-1993	Aug-1993	Sep-1993	Oct-1993	Nov-1993	Dec-1993	Jan-1994	Feb-1994	Mar-1994	Apr-1994
Balance Sheet												
Balances with Central Bank	40.00	41.00	41.00	60.00	36.00	38.00	60.00	26.00	22.00	40.00	62.00	38.00
Cash and Cash-equivalent Securities	25.00	10.00	15.00	20.00	35.00	15.00	20.00	10.00	12.00	15.00	22.00	18.00
⊟ Cash	65.00	51.00	56.00	80.00	71.00	53.00	80.00	36.00	34.00	55.00	84.00	56.00
Balances with Central Bank	40.00	41.00	41.00	60.00	36.00	38.00	60.00	26.00	22.00	40.00	62.00	38.00
Cash and Cash-equivalent Securities	25.00	10.00	15.00	20.00	35.00	15.00	20.00	10.00	12.00	15.00	22.00	18.00
⊞ Retail	10.00	16.00	11.00	15.00	21.00	18.00	20.00	16.00	10.00	25.00	40.00	20.00
⊟ Loans '&' Advances to Customers	10.00	16.00	11.00	15.00	21.00	18.00	20.00	16.00	10.00	25.00	40.00	20.00
⊞ Retail	10.00	16.00	11.00	15.00	21.00	18.00	20.00	16.00	10.00	25.00	40.00	20.00
⊞ Total Liabilities	40.00	41.00	41.00	60.00	36.00	38.00	60.00	26.00	22.00	40.00	62.00	38.00
⊞ Shareholders Equity	30.00	25.00	30.00	45.00	15.00	20.00	40.00	10.00	12.00	15.00	22.00	18.00
⊟ Total Liabilities '&' Sharesholders Equity	70.00	66.00	71.00	105.00	51.00	58.00	100.00	36.00	34.00	55.00	84.00	56.00
⊞ Total Liabilities	40.00	41.00	41.00	60.00	36.00	38.00	60.00	26.00	22.00	40.00	62.00	38.00
⊞ Shareholders Equity	30.00	25.00	30.00	45.00	15.00	20.00	40.00	10.00	12.00	15.00	22.00	18.00

- Management Reporting - Balance Sheet - Key Indicators



- Management Reporting - Income Statement - Income Statement


ORACLE Business Intelligence
Search All
Advanced
Administration
Help
Sign Out
Income Statement
Home
Catalog
Favorites
Dashboards
New
Open
Signed In As weblogic
Income Statement
Income Statement - Time Series
Income Statement - Key Indicators
Income Statement Variance Analysis
Income Statement-Line Of Business
Apply
Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: 1993

Legal Entity: Bank Russia

Line Of Business: Missing LOB

Organization Unit: (All Column Values)

Product: (All Column Values)

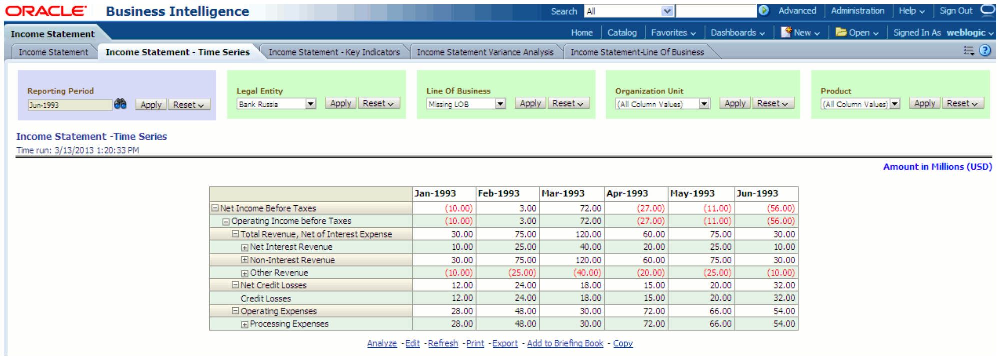
Income Statement for Apr-1994

Time run: 3/13/2013 1:19:04 PM

Amount in Millions (USD)

Income Statement	Actual	Actual YTD	B/(W) Prior Period	B/(W) Prior Yr	B/(W) Plan	B/(W) Forecast
⊟ Net Income Before Taxes	(27.00)	(102.00)	0.00	0.00	(39.00)	(23.00)
⊟ Operating Income before Taxes	(27.00)	(102.00)	0.00	0.00	(39.00)	(23.00)
⊟ Total Revenue, Net of Interest Expense	45.00	207.00	(15.00)	(15.00)	(15.00)	(15.00)
⊞ Net Interest Revenue	15.00	69.00	(5.00)	(5.00)	(5.00)	(5.00)
⊞ Non-Interest Revenue	45.00	207.00	(15.00)	(15.00)	(15.00)	(15.00)
⊞ Other Revenue	(15.00)	(69.00)	5.00	5.00	5.00	5.00
⊟ Net Credit Losses	18.00	108.00	3.00	3.00	0.00	8.00
Credit Losses	18.00	108.00	3.00	3.00	0.00	8.00
⊟ Operating Expenses	54.00	201.00	(18.00)	(18.00)	24.00	0.00
⊞ Processing Expenses	54.00	201.00	(18.00)	(18.00)	24.00	0.00

- Management Reporting - Income Statement - Income Statement - Time Series



ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Income Statement
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Income Statement | Income Statement - Time Series | Income Statement - Key Indicators | Income Statement Variance Analysis | Income Statement-Line Of Business
Apply
Reset

Reporting Period: Jun-1993

Legal Entity: Bank Russia

Line Of Business: Missing LOB

Organization Unit: (All Column Values)

Product: (All Column Values)

Income Statement -Time Series

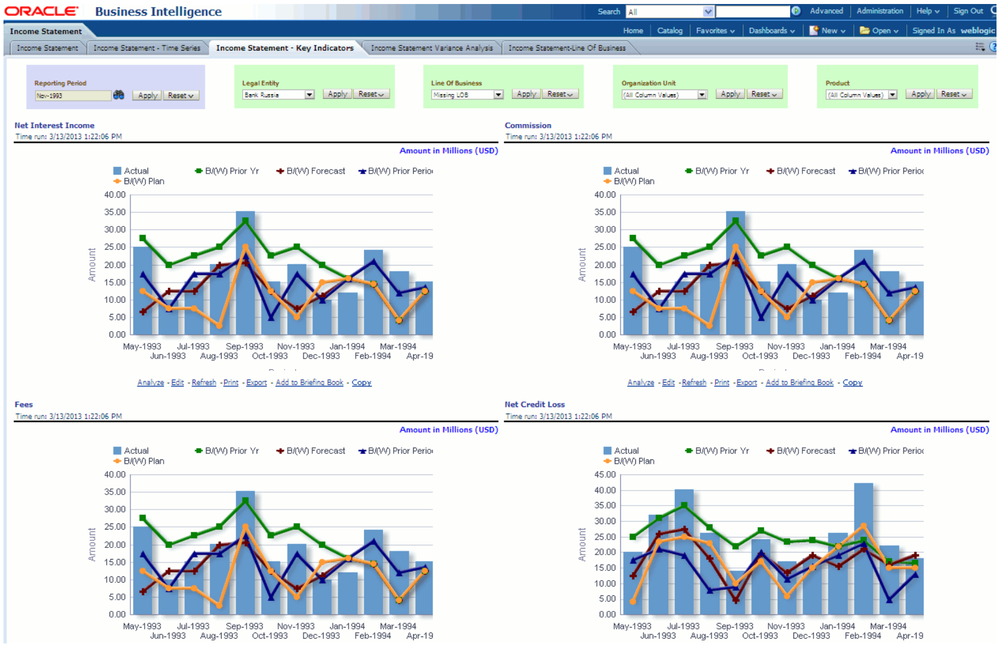
Time run: 3/13/2013 1:20:33 PM

Amount in Millions (USD)

	Jan-1993	Feb-1993	Mar-1993	Apr-1993	May-1993	Jun-1993
⊟ Net Income Before Taxes	(10.00)	3.00	72.00	(27.00)	(11.00)	(56.00)
⊟ Operating Income before Taxes	(10.00)	3.00	72.00	(27.00)	(11.00)	(56.00)
⊟ Total Revenue, Net of Interest Expense	30.00	75.00	120.00	60.00	75.00	30.00
⊞ Net Interest Revenue	10.00	25.00	40.00	20.00	25.00	10.00
⊞ Non-Interest Revenue	30.00	75.00	120.00	60.00	75.00	30.00
⊞ Other Revenue	(10.00)	(25.00)	(40.00)	(20.00)	(25.00)	(10.00)
⊟ Net Credit Losses	12.00	24.00	18.00	15.00	20.00	32.00
Credit Losses	12.00	24.00	18.00	15.00	20.00	32.00
⊟ Operating Expenses	28.00	48.00	30.00	72.00	66.00	54.00
⊞ Processing Expenses	28.00	48.00	30.00	72.00	66.00	54.00

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

- Management Reporting - Income Statement - Income Statement - Key Indicators



ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Income Statement
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Income Statement | Income Statement - Time Series | Income Statement - Key Indicators | Income Statement Variance Analysis | Income Statement-Line Of Business
Apply
Reset
Actual
B/(W) Prior Yr
B/(W) Forecast
B/(W) Prior Perio
B/(W) Plan
Amount
45.00
40.00
35.00
30.00
25.00
20.00
15.00
10.00
5.00
0.00
May-1993
Jun-1993
Jul-1993
Aug-1993
Sep-1993
Oct-1993
Nov-1993
Dec-1993
Jan-1994
Feb-1994
Mar-1994
Apr-19
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: Nov-1993

Legal Entity: Bank Russia

Line Of Business: Missing LOB

Organization Unit: (All Column Values)

Product: (All Column Values)

Net Interest Income

Time run: 3/13/2013 1:22:06 PM

Amount in Millions (USD)

Commission

Time run: 3/13/2013 1:22:06 PM

Amount in Millions (USD)

Fees

Time run: 3/13/2013 1:22:06 PM

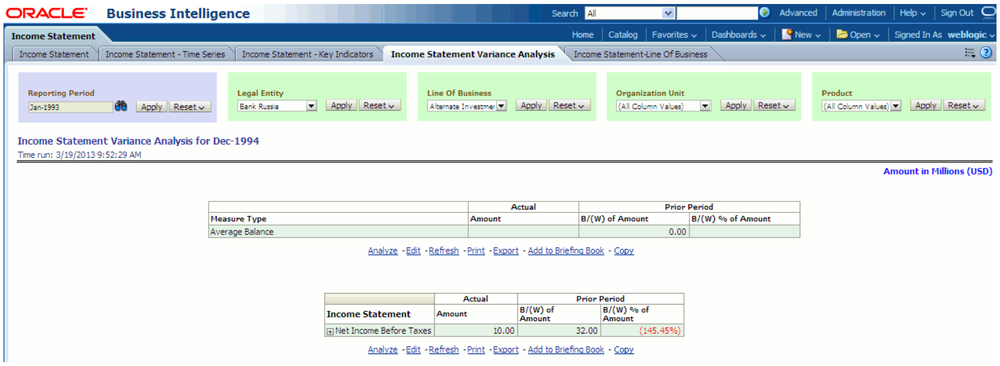
Amount in Millions (USD)

Net Credit Loss

Time run: 3/13/2013 1:22:06 PM

Amount in Millions (USD)

- Management Reporting - Income Statement - Income Statement Variance Analysis


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Income Statement
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Income Statement | Income Statement - Time Series | Income Statement - Key Indicators | Income Statement Variance Analysis | Income Statement-Line Of Business
Apply | Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: Jan-1993

Legal Entity: Bank Russia

Line Of Business: Alternate Investmei

Organization Unit: (All Column Values)

Product: (All Column Values)

Income Statement Variance Analysis for Dec-1994

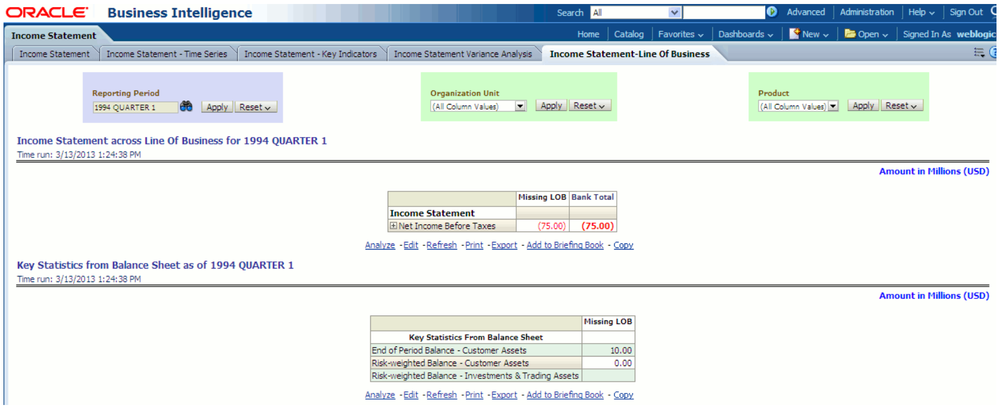
Time run: 3/19/2013 9:52:29 AM

Amount in Millions (USD)

	Actual	Prior Period	
Measure Type	Amount	B/(W) of Amount	B/(W) % of Amount
Average Balance		0.00	

	Actual	Prior Period	
Income Statement	Amount	B/(W) of Amount	B/(W) % of Amount
⊞ Net Income Before Taxes	10.00	32.00	(145.45%)

- Management Reporting - Income Statement - Income Statement-Line Of Business


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Income Statement
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Income Statement | Income Statement - Time Series | Income Statement - Key Indicators | Income Statement Variance Analysis | Income Statement-Line Of Business
Apply | Reset
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: 1994 QUARTER 1

Organization Unit: (All Column Values)

Product: (All Column Values)

Income Statement across Line Of Business for 1994 QUARTER 1

Time run: 3/13/2013 1:24:38 PM

Amount in Millions (USD)

	Missing LOB	Bank Total
Income Statement		
⊞ Net Income Before Taxes	(75.00)	(75.00)

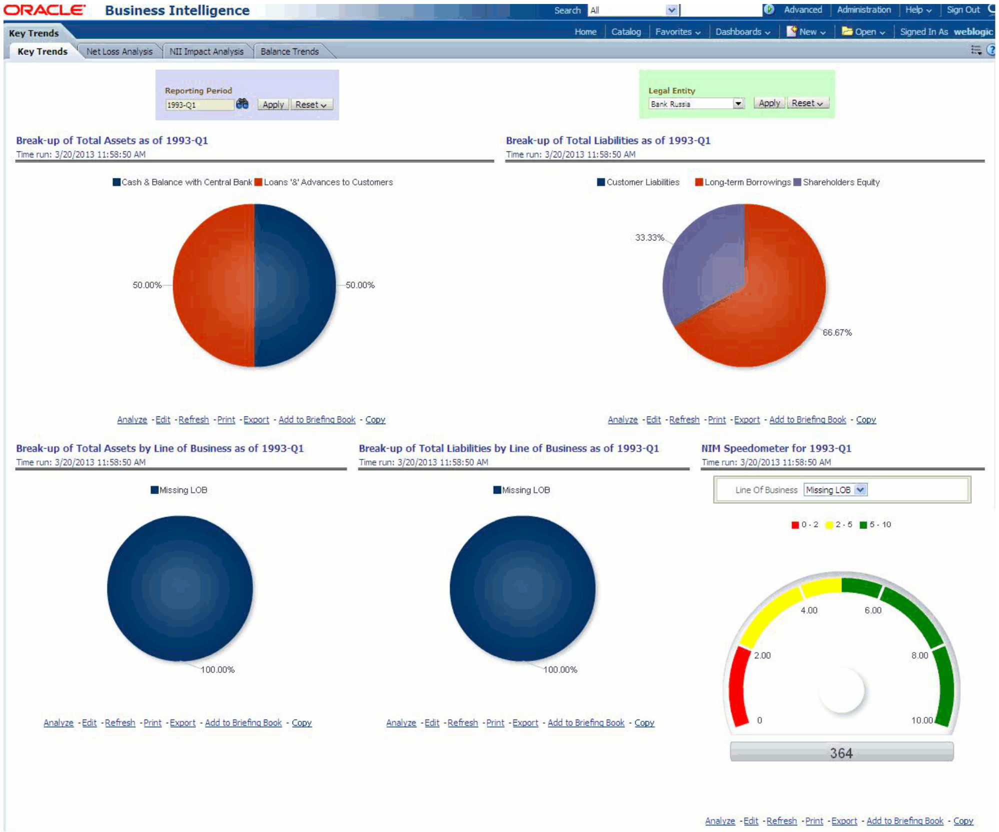
Key Statistics from Balance Sheet as of 1994 QUARTER 1

Time run: 3/13/2013 1:24:38 PM

Amount in Millions (USD)

	Missing LOB
Key Statistics From Balance Sheet	
End of Period Balance - Customer Assets	10.00
Risk-weighted Balance - Customer Assets	0.00
Risk-weighted Balance - Investments & Trading Assets	

- Management Reporting - Income Statement - What-If Analysis
- Management Reporting - Key Trends - Key Trends


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Key Trends
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Key Trends | Net Loss Analysis | NII Impact Analysis | Balance Trends
Apply
Reset
■ Cash & Balance with Central Bank ■ Loans '&' Advances to Customers
50.00%
50.00%
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
■ Customer Liabilities ■ Long-term Borrowings ■ Shareholders Equity
33.33%
66.67%
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
■ Missing LOB
100.00%
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
■ Missing LOB
100.00%
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
🟥 0 - 2 🟨 2 - 5 🟩 5 - 10
0
2.00
4.00
6.00
8.00
10.00
364
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Reporting Period: 1993-Q1

Legal Entity: Bank Russia

Break-up of Total Assets as of 1993-Q1
Time run: 3/20/2013 11:58:50 AM

Break-up of Total Liabilities as of 1993-Q1
Time run: 3/20/2013 11:58:50 AM

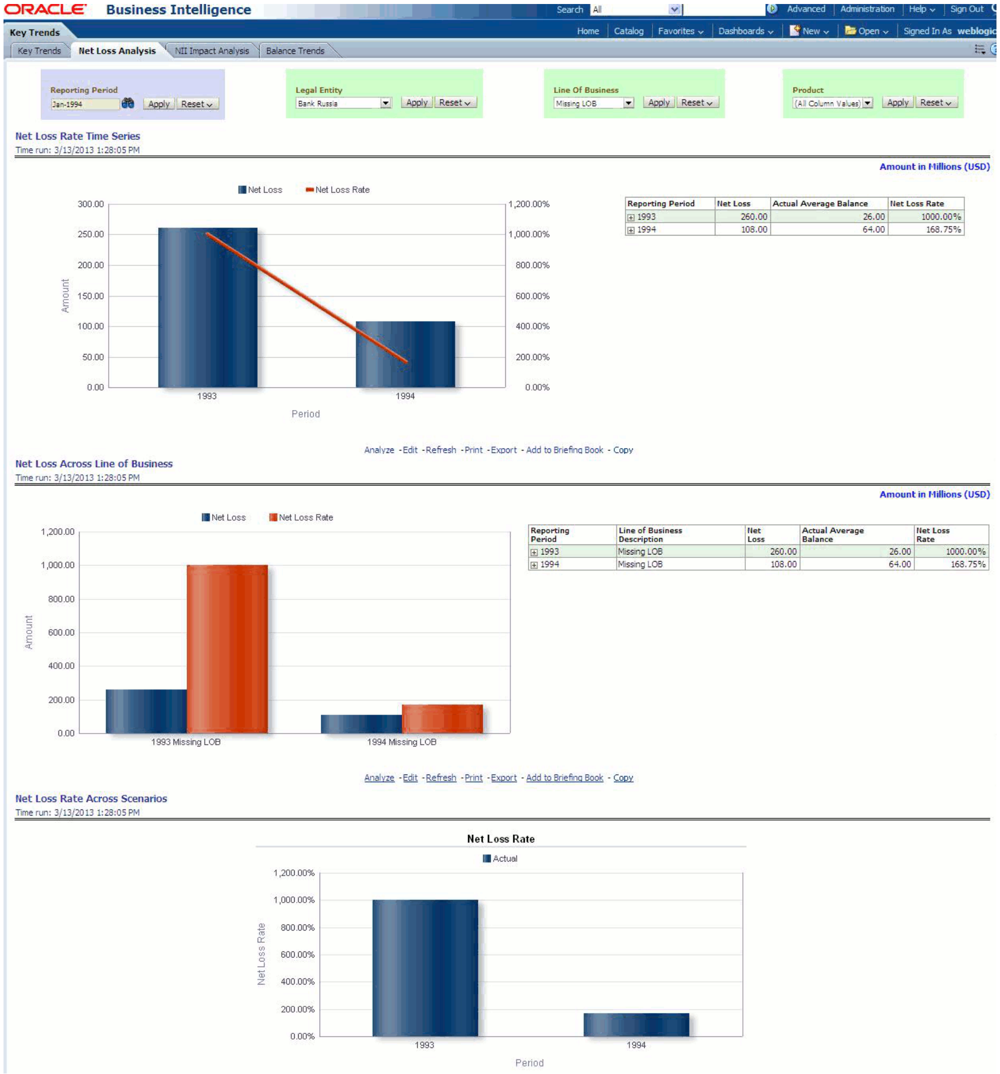
Break-up of Total Assets by Line of Business as of 1993-Q1
Time run: 3/20/2013 11:58:50 AM

Break-up of Total Liabilities by Line of Business as of 1993-Q1
Time run: 3/20/2013 11:58:50 AM

NIM Speedometer for 1993-Q1
Time run: 3/20/2013 11:58:50 AM

Line Of Business: Missing LOB

- Management Reporting - Key Trends - Net Loss Analysis



Reporting Period: Jan-1994

Legal Entity: Bank Russia

Line Of Business: Missing LOB

Product: (All Column Values)

Net Loss Rate Time Series

Time run: 3/13/2013 1:28:05 PM

Amount in Millions (USD)

Reporting Period	Net Loss	Actual Average Balance	Net Loss Rate
⊞ 1993	260.00	26.00	1000.00%
⊞ 1994	108.00	64.00	168.75%

Net Loss Across Line of Business

Time run: 3/13/2013 1:28:05 PM

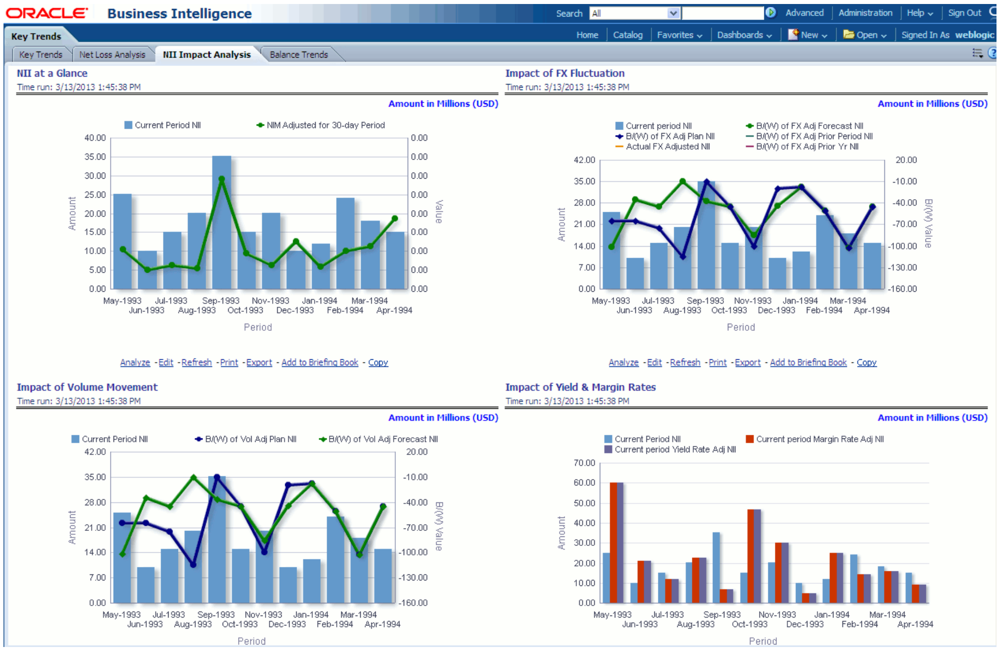
Amount in Millions (USD)

Reporting Period	Line of Business Description	Net Loss	Actual Average Balance	Net Loss Rate
⊞ 1993	Missing LOB	260.00	26.00	1000.00%
⊞ 1994	Missing LOB	108.00	64.00	168.75%

Net Loss Rate Across Scenarios

Time run: 3/13/2013 1:28:05 PM

- Management Reporting - Key Trends - NII Impact Analysis


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Key Trends
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Key Trends | Net Loss Analysis | NII Impact Analysis | Balance Trends

NII at a Glance
Time run: 3/13/2013 1:45:38 PM
Amount in Millions (USD)

Current Period NII
NIM Adjusted for 30-day Period
40.00
35.00
30.00
25.00
20.00
15.00
10.00
5.00
0.00
Amount
Value
May-1993 Jun-1993 Jul-1993 Aug-1993 Sep-1993 Oct-1993 Nov-1993 Dec-1993 Jan-1994 Feb-1994 Mar-1994 Apr-1994
Period
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Impact of FX Fluctuation
Time run: 3/13/2013 1:45:38 PM
Amount in Millions (USD)

Current period NII
B/(W) of FX Adj Plan NII
Actual FX Adjusted NII
B/(W) of FX Adj Forecast NII
B/(W) of FX Adj Prior Period NII
B/(W) of FX Adj Prior Yr NII
42.00
35.00
28.00
21.00
14.00
7.00
0.00
20.00
-10.00
-40.00
-70.00
-100.00
-130.00
-160.00
Amount
B/(W) Value
May-1993 Jun-1993 Jul-1993 Aug-1993 Sep-1993 Oct-1993 Nov-1993 Dec-1993 Jan-1994 Feb-1994 Mar-1994 Apr-1994
Period
Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

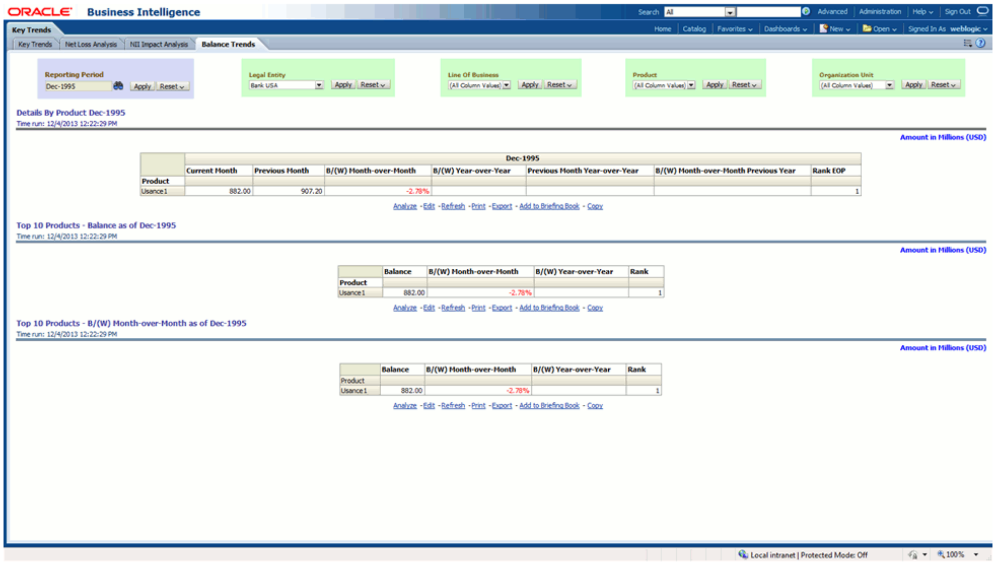
Impact of Volume Movement
Time run: 3/13/2013 1:45:38 PM
Amount in Millions (USD)

Current Period NII
B/(W) of Vol Adj Plan NII
B/(W) of Vol Adj Forecast NII
42.00
35.00
28.00
21.00
14.00
7.00
0.00
20.00
-10.00
-40.00
-70.00
-100.00
-130.00
-160.00
Amount
B/(W) Value
May-1993 Jun-1993 Jul-1993 Aug-1993 Sep-1993 Oct-1993 Nov-1993 Dec-1993 Jan-1994 Feb-1994 Mar-1994 Apr-1994
Period

Impact of Yield & Margin Rates
Time run: 3/13/2013 1:45:38 PM
Amount in Millions (USD)

Current Period NII
Current period Yield Rate Adj NII
Current period Margin Rate Adj NII
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
Amount
May-1993 Jun-1993 Jul-1993 Aug-1993 Sep-1993 Oct-1993 Nov-1993 Dec-1993 Jan-1994 Feb-1994 Mar-1994 Apr-1994
Period

- Management Reporting - Key Trends - Balance Trends


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Key Trends
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogic
Key Trends | Net Loss Analysis | NII Impact Analysis | Balance Trends

Reporting Period: Dec-1995 Apply Reset
Legal Entity: Bank USA Apply Reset
Line Of Business: (All Column Values) Apply Reset
Product: (All Column Values) Apply Reset
Organization Unit: (All Column Values) Apply Reset

Details By Product Dec-1995
Time run: 12/4/2013 12:22:29 PM

Amount in Millions (USD)

	Dec-1995						
	Current Month	Previous Month	B/(W) Month-over-Month	B/(W) Year-over-Year	Previous Month Year-over-Year	B/(W) Month-over-Month Previous Year	Rank EOP
Product							
Usance1	882.00	907.20	-2.78%				1

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Top 10 Products - Balance as of Dec-1995
Time run: 12/4/2013 12:22:29 PM

Amount in Millions (USD)

	Balance	B/(W) Month-over-Month	B/(W) Year-over-Year	Rank
Product				
Usance1	882.00	-2.78%		1

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

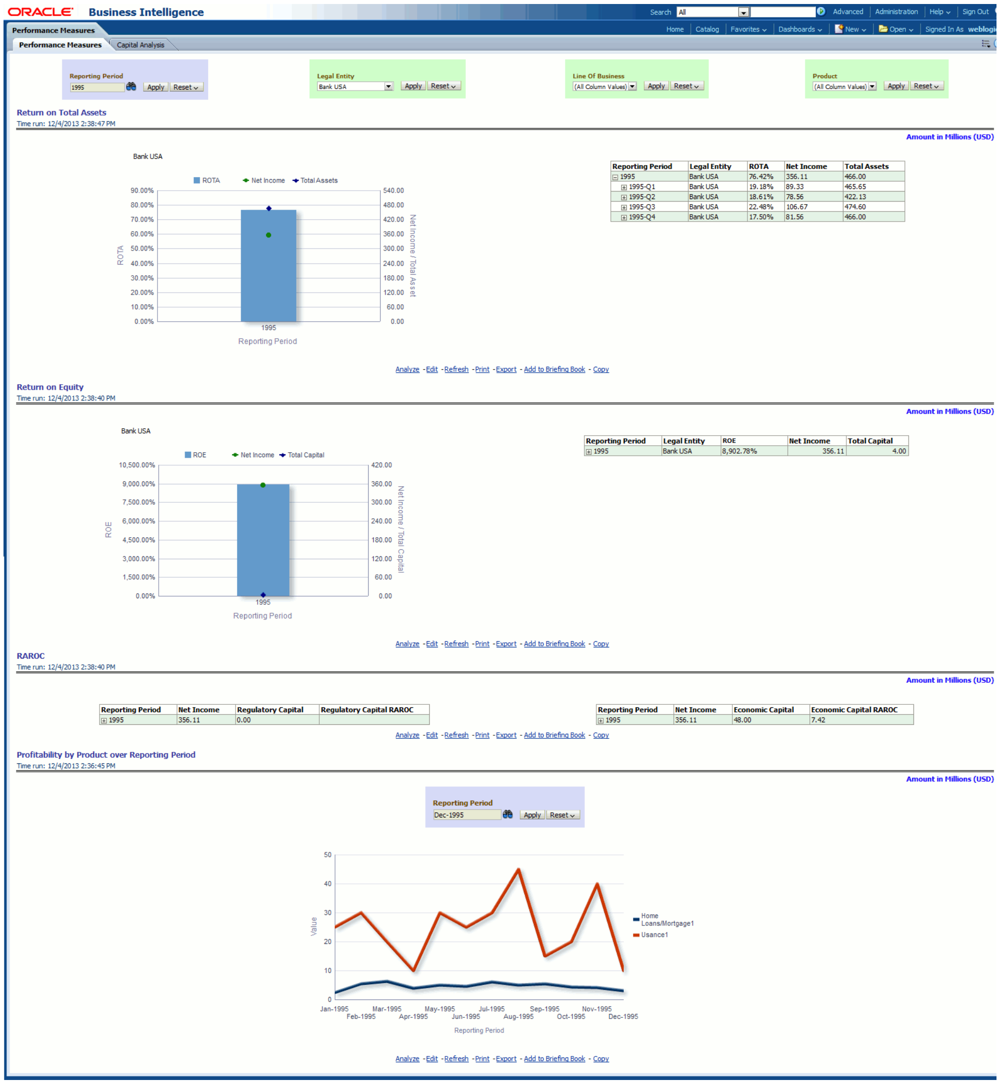
Top 10 Products - B/(W) Month-over-Month as of Dec-1995
Time run: 12/4/2013 12:22:29 PM

Amount in Millions (USD)

	Balance	B/(W) Month-over-Month	B/(W) Year-over-Year	Rank
Product				
Usance1	882.00	-2.78%		1

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy
Local intranet | Protected Mode: Off
100%

- Management Reporting - Performance Measures - Performance Measures


ORACLE Business Intelligence
Search All
Advanced | Administration | Help | Sign Out
Performance Measures
Home | Catalog | Favorites | Dashboards | New | Open | Signed In As weblogi
Performance Measures
Capital Analysis

Reporting Period 1995 Apply Reset

Legal Entity Bank USA Apply Reset

Line Of Business (All Column Values) Apply Reset

Product (All Column Values) Apply Reset

Return on Total Assets

Time run: 12/4/2013 2:38:47 PM

Amount in Millions (USD)

Reporting Period	Legal Entity	ROTA	Net Income	Total Assets
⊟ 1995	Bank USA	76.42%	356.11	466.00
⊞ 1995-Q1	Bank USA	19.18%	89.33	465.65
⊞ 1995-Q2	Bank USA	18.61%	78.56	422.13
⊞ 1995-Q3	Bank USA	22.48%	106.67	474.60
⊞ 1995-Q4	Bank USA	17.50%	81.56	466.00

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Return on Equity

Time run: 12/4/2013 2:38:40 PM

Amount in Millions (USD)

Reporting Period	Legal Entity	ROE	Net Income	Total Capital
⊞ 1995	Bank USA	8,902.78%	356.11	4.00

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

RAROC

Time run: 12/4/2013 2:38:40 PM

Amount in Millions (USD)

Reporting Period	Net Income	Regulatory Capital	Regulatory Capital RAROC
⊞ 1995	356.11	0.00	

Reporting Period	Net Income	Economic Capital	Economic Capital RAROC
⊞ 1995	356.11	48.00	7.42

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

Profitability by Product over Reporting Period

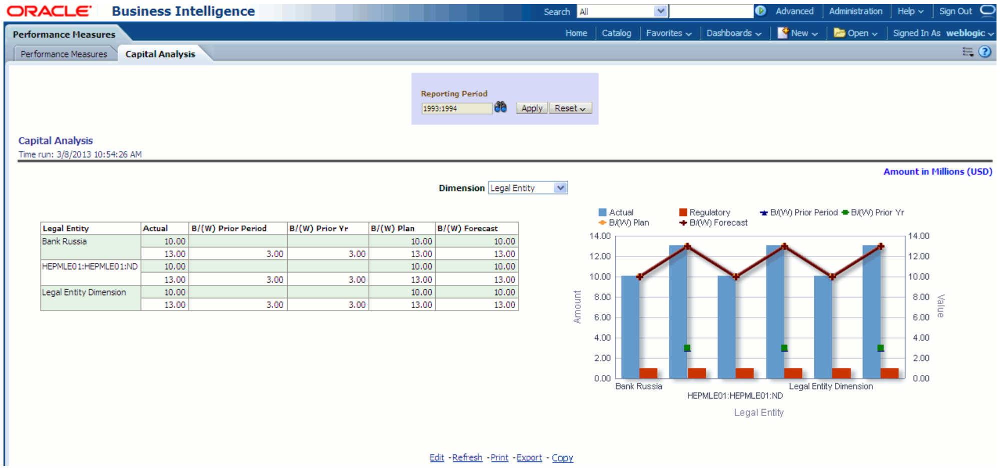
Time run: 12/4/2013 2:36:45 PM

Amount in Millions (USD)

Reporting Period Dec-1995 Apply Reset

Analyze - Edit - Refresh - Print - Export - Add to Briefing Book - Copy

- Management Reporting - Performance Measures – Capital Analysis



Capital Analysis

Time run: 3/8/2013 10:54:26 AM

Amount in Millions (USD)

Dimension Legal Entity

Legal Entity	Actual	B/(W) Prior Period	B/(W) Prior Yr	B/(W) Plan	B/(W) Forecast
Bank Russia	10.00			10.00	10.00
	13.00	3.00	3.00	13.00	13.00
HEPMLE01:HEPMLE01:ND	10.00			10.00	10.00
	13.00	3.00	3.00	13.00	13.00
Legal Entity Dimension	10.00			10.00	10.00
	13.00	3.00	3.00	13.00	13.00

A

How to Define a Dimension

Introduction - Dimension Definition

As a prerequisite, dimension tables should be added in the data model and the fact table needs to have the referential key with the dimension table. These dimension tables hold the dimension members and can be level-based or parent-child. Level based dimension tables contain columns for each level of the hierarchy, while parent-child dimension tables contain columns for storing the relationship between the parent and child members. These dimension tables can be loaded from external systems or can be maintained within the Dimension Management component of OFSAAI. If user intends to maintain the dimension within OFSAAI for adding dimension tables, refer to *Oracle Financial Services Analytical Applications Data Model Utilities User Guide*. If the dimension data is fetched from OFSPM, it reads only the nodes/leafs belonging to a single hierarchy. Therefore, before fetching dimension data from OFSPM, ensure that the AMHM tables belonging to a single dimension have data present for only one hierarchy.

Dimension Definition Process

Create Business Hierarchy

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Business Hierarchy**.

2. From Business Hierarchy, click **Add** to create a Business hierarchy definition. In the *Business Hierarchy Definition (Add mode)* window, select the **Hierarchy Type**. Hierarchy Type can be :

   - **Regular** - for representing non-time and non-measure dimensions in a hierarchical format. Examples of this type are Product, Organization Unit, and so on.

- **Measure** - for representing the measures in the hierarchical format. An example of this type is Management Reporting Line.

- **Time** - for representing the calendar or date dimension in a hierarchical format. An example of this type is Calendar hierarchy.

3. Select **Hierarchy Sub Type**.

   Hierarchy Sub Type can be:

   - **Non Business Intelligence Enabled** - for representing the hierarchy with underlying data store containing just leaves and nodes are built within the metadata of the hierarchy. This sub type is useful for modelling bucket/range, ragged and non-additive hierarchies.

   - **Business Intelligence Enabled** - for representing the hierarchy with underlying data store as level-based dimension table. This sub-type is useful for modelling balanced hierarchies.

   - Parent **Child** - for representing the hierarchy with underlying data store as a parent-child dimension table. This sub type is useful for modelling ragged hierarchies.

4. Select **Total Required** property, if a TOTAL is required to be included as the root node of the hierarchy.

5. Select **List** property, if hierarchy is a flat list of members without any levels.

6. Select the **Entity** and **Attribute** on which the hierarchy is based.

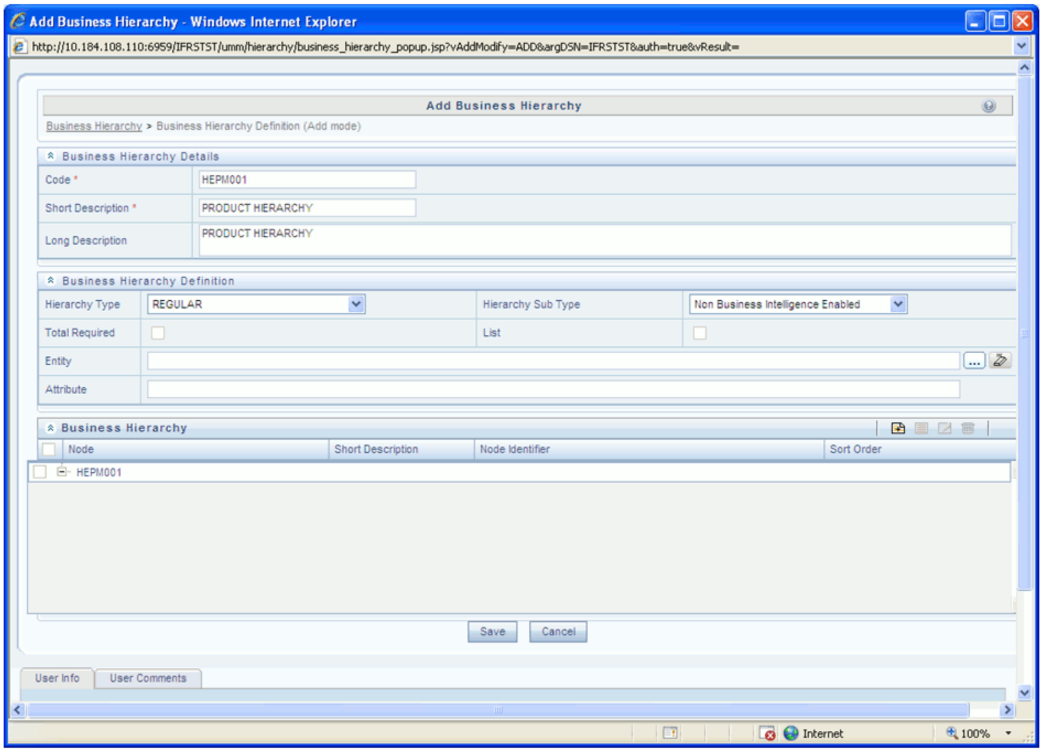
The components for hierarchy definition differ for each sub type of the hierarchy.

- If sub type is **Non Business Intelligence Enabled**, then the user can add nodes and the order in which the node should appear in the hierarchy (sort-order). Node identifiers are SQL expressions that are specified for leaf members and data is classified based on the node identifiers.

- If sub type is **Business Intelligence Enabled**, then the user can specify the levels and SQL expression for each level within the hierarchy.

- If sub type is **Parent Child**, then the user can specify the column that contains the parent member and the column that contains the child member.

For more details, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.



Add Business Hierarchy - Windows Internet Explorer
http://10.184.108.110:6959/IFRSTST/umm/hierarchy/business_hierarchy_popup.jsp?vAddModify=ADD&argDSN=IFRSTST&auth=true&vResult=
Add Business Hierarchy
Business Hierarchy > Business Hierarchy Definition (Add mode)

Business Hierarchy Details

Code *	HEPM001
Short Description *	PRODUCT HIERARCHY
Long Description	PRODUCT HIERARCHY

Business Hierarchy Definition

Hierarchy Type	REGULAR	Hierarchy Sub Type	Non Business Intelligence Enabled
Total Required	□	List	□
Entity			
Attribute			

Business Hierarchy

□	Node	Short Description	Node Identifier	Sort Order
□	HEPM001			

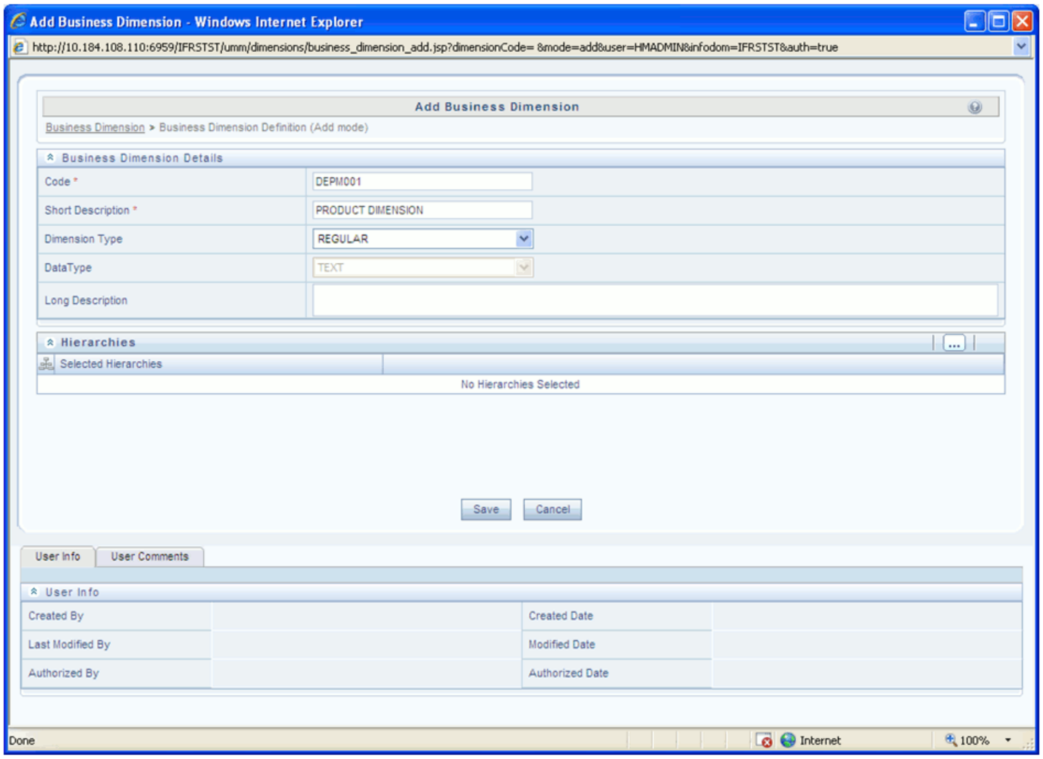
Save
Cancel
User Info
User Comments
Internet
100%

Create Business Dimension

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Business Dimension**.

2. From Business Dimension, click **Add** to create a Business dimension definition. In the *Business Dimension Definition (Add mode)* window, select the **Dimension Type**. Dimension type is same as Hierarchy type and helps to filter the hierarchies that will be part of the dimension.

3. A dimension will contain one or many hierarchies. Select the hierarchies that are part of the dimension.



Add Business Dimension - Windows Internet Explorer
http://10.184.108.110:6959/IFRSTST/umm/dimensions/business_dimension_add.jsp?dimensionCode= &mode=add&user=HMADMIN&infodom=IFRSTST&auth=true
Save
Cancel
User Info
User Comments
Done
Internet
100%

Add Business Dimension

Business Dimension > Business Dimension Definition (Add mode)

Business Dimension Details

Code *	DEPM001
Short Description *	PRODUCT DIMENSION
Dimension Type	REGULAR
DataType	TEXT
Long Description	

Hierarchies

Selected Hierarchies

No Hierarchies Selected

User Info

Created By		Created Date	
Last Modified By		Modified Date	
Authorized By		Authorized Date	

Modify Dataset

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Data Sets**.

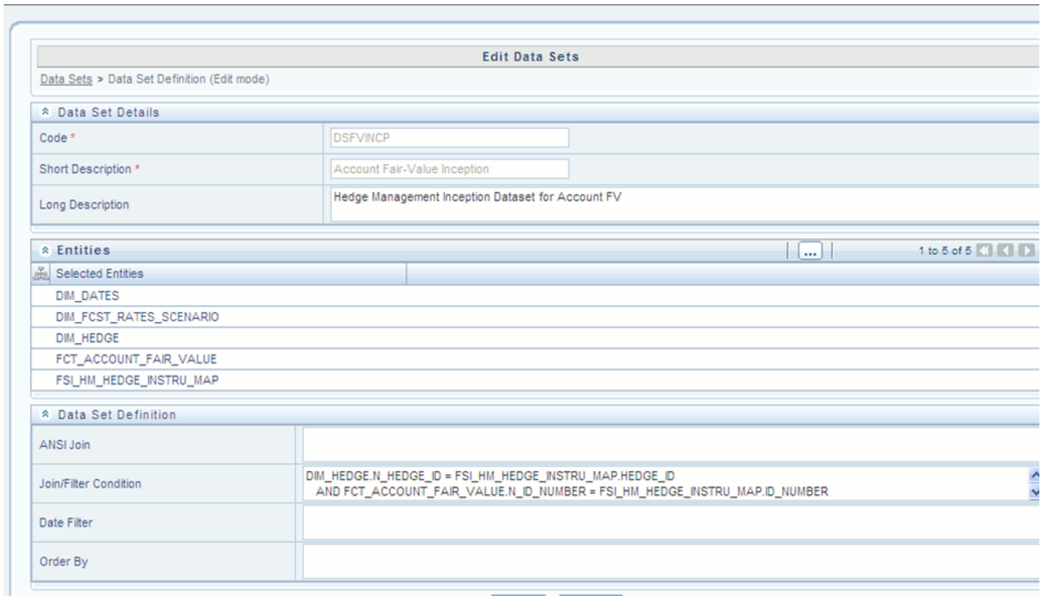
2. Identify data sets that are based on the modified fact table.

3. Edit the data set definition.

4. Include the new dimension table in the data set.

5. Modify the data set JOIN to include the join clause between the fact table and new dimension table.

6. Save the data set.



Edit Data Sets

Data Sets > Data Set Definition (Edit mode)

Data Set Details

Code *	DSFVINCP
Short Description *	Account Fair-Value Inception
Long Description	Hedge Management Inception Dataset for Account FV

Entities

Selected Entities
DIM_DATES
DIM_FCST_RATES_SCENARIO
DIM_HEDGE
FCT_ACCOUNT_FAIR_VALUE
FSI_HM_HEDGE_INSTRU_MAP

Data Set Definition

ANSI Join	
Join/Filter Condition	DIM_HEDGE.N_HEDGE_ID = FSI_HM_HEDGE_INSTRU_MAP.HEDGE_ID AND FCT_ACCOUNT_FAIR_VALUE.N_ID_NUMBER = FSI_HM_HEDGE_INSTRU_MAP.ID_NUMBER
Date Filter	
Order By	

B

How to Define a Measure

Introduction to Measure Definition

As a prerequisite, the fact table needs to have the column that holds values for the new measure.

Measure Definition Process

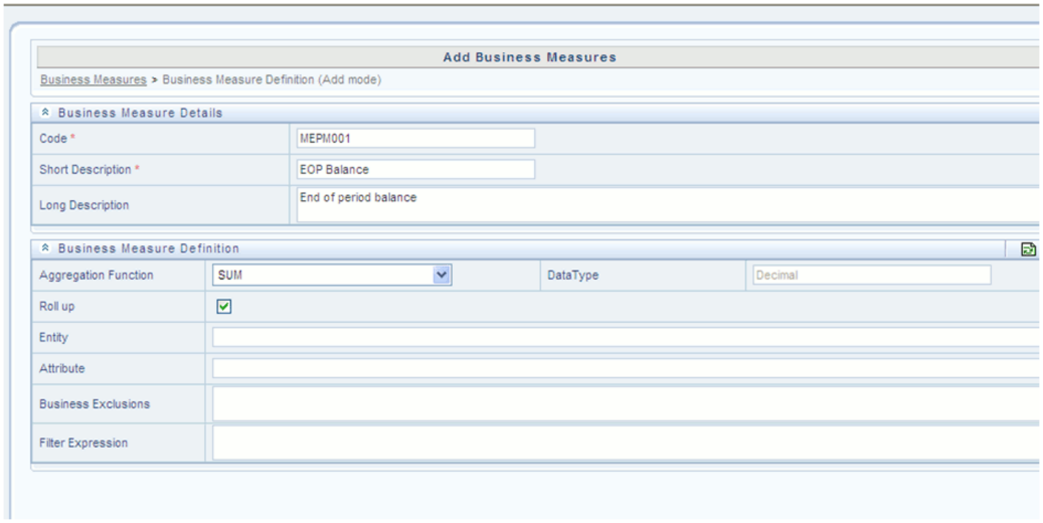
Create Business Measure

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Business Measures**.
2. From Business Measures, click **Add** to create a Business measure definition. In the *Business Measure Definition (Add mode)* window, Select **Aggregation Function**. Aggregation Function can be:
   - SUM - for summing up the values in the column of the fact table.
   - COUNT - for determining the number of records in the fact table.
   - MAXIMUM - for identifying the maximum value of a column in the fact table.
   - MINIMUM - for identifying the minimum value of a column in the fact table.
   - COUNT DISTINCT - for determining the distinct count of records in the fact table.
3. Specify if this measure needs to be rolled up against hierarchies.
4. Select the fact table as part of the Entity.

5. Select the column of the fact table as part of the Attribute. This column will hold the value of the measure.

6. Specify Business Exclusions and Filters, if required.

7. Save the measure.



Add Business Measures

Business Measures > Business Measure Definition (Add mode)

Business Measure Details

Code *	MEPM001
Short Description *	EOP Balance
Long Description	End of period balance

Business Measure Definition

Aggregation Function	SUM	DataType	Decimal
Roll up	☑		
Entity			
Attribute			
Business Exclusions			
Filter Expression			

C

How to Develop a New Cube

Introduction

This section details the steps to be performed by the user for developing a new cube. Make sure that the existing cubes do not provide the required analytics/reporting coverage before deciding to define a new cube. If you like to see measures against a new dimension that is not part of the existing seeded metadata, then include the new dimension as part of the existing cubes instead of creating a new cube. As a prerequisite, you should have defined datasets, measures, hierarchies, and dimensions.

Cube Definition

Procedure

1. From OFSAAI Home screen, navigate to **Unified Metadata Manager > Business Metadata Management > Cube** and specify the MDB details to be created in ESSBASE.

2. Include dimensions that are part of the cube definition. It is mandatory that you include TIME and MEASURE dimensions.

3. Specify variations between each of the measures to the respective dimensions.

   All the measures that are part of the cube need not vary against all of the dimensions. Depending on business needs, variations can be specified to control the rollup of measures against a set of dimensions.

4. Specify dataset corresponding to the selected dimensions and measures.

   Dataset supplies required data to the cube.

5. Specify the node level formulas for the nodes within the hierarchy.

6. Save the cube.

Define and execute the Batch from *Operations* module of OFSAAI.

For more information on Cubes, refer to *Cubes* section under *Unified Metadata Manager* chapter in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

D

How to Define a Batch

Batch Definition

Create a batch from the OFSAAI *Batch Maintenance* screen as follows:

Procedure

1. From the OFSAAI **Home** menu, navigate to **Operations > Batch Maintenance**.
2. In the *Batch Maintenance* window, Select '+' button from the *Batch Name* tool bar.

   The *New Batch Definition* window is displayed.
3. Enter the Batch details as tabulated.

Field	Description
Batch Name	The **Batch Name** is auto generated by the system. You can edit to specify a Batch name based on the following conditions: • The Batch Name should be unique across the Information Domain. • The Batch Name must be alphanumeric and should not start with a number. • The Batch Name should not exceed 41 characters in length. • The Batch Name should not contain special characters "." and "-".

Field	Description
Batch Description	Enter a description for the Batch based on the Batch Name.
Duplicate Batch	(Optional) Select the check box to create a new Batch by duplicating the existing Batch details. On selection, the **Batch ID** field is enabled.
Batch ID (If duplicate Batch is selected)	It is mandatory to specify the Batch ID if Duplicate Batch option is selected. Select the required **Batch ID** from the list.
Sequential Batch	Select the check box if the Batch has to be created sequentially based on the task specified. For example, if there are 3 tasks defined in a Batch, task 3 should have precedence as task 2, and task 2 should have precedence as task 1.

4. Click **Save** to save the Batch definition details.

   The new Batch definition details are displayed in the *Batch Name* section of *Batch Maintenance* window with the specified **Batch ID**.