Oracle Financial Services Institutional Performance Analytics User Guide

Release 8.0.5.0.0
October 2020

ORACLE

Oracle Financial Services Institutional Performance Analytics User Guide

Release 8.0.5.0.0
October 2020

Part Number: E73551-02

Oracle Financial Services Software, Inc.
1900 Oracle Way
Reston, VA 20190

Oracle Financial Services Software, Inc.
1900 Oracle Way
Reston, VA 20190
Phone: (703) 478-9000
Fax: (703) 318-6340
Internet: www.oracle.com/financialservices

Contents

List of Tables

List of Figures

Preface

Intended Audience

Welcome to Release 8.0.4.0.0 of the *Oracle Financial Services Institutional Performance Analytics* User Guide.

This user guide is intended for the users of Oracle Financial Services Institutional Performance Analytics application.

See *Related Information Sources* 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 have access to electronic support through My Oracle Support. For information, visit:

- http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info
- http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Structure

This user guide has been segregated into the following chapters:

- Chapter 1-Introduction
- Chapter 2-Overview of Process Flow
- Chapter 3-Dimension Loading Process
- Chapter 4-Time Dimension Population
- Chapter 5-Customer Dimension Population
- Chapter 6-Account Dimension Population
- Chapter 7-Exchange Rate History Population

- Chapter 8-Account Summary Population
- Chapter 9-Fact Transaction Summary
- Chapter 10-Customer Summary Population
- Chapter 11-Fact Data Population
- Chapter 12-Cube Build Process
- Chapter 13-Time Series Forecasting
- Chapter 14-Segmentation
- Chapter 15-Overview of OFSIPA Reports
- Chapter 16-What-If Analysis
- Chapter 17-Service Calls to IPA
- Chapter 18-Visibility
- Appendix A, How to Add a New Dimension
- Appendix B, How to Add a New Measure
- Appendix C, How to Develop a New Cube
- Appendix D, How to Define a Batch
- Appendix E, List of Hard-Coded Members
- Appendix F, Run Rule Framework
- Appendix G, Loading Multiple Load Runs in OFSAA

Related Information Sources

- Oracle Financial Services Advanced Analytical Applications Infrastructure Installation and Configuration Guide
- Oracle Financial Services Advanced Analytical Applications Infrastructure User Guide
- Oracle Financial Services Retail Performance Analytics User Guide
- Oracle Financial Services Retails Customer Analytics User Guide

CHAPTER 1

Introduction

Overview of Oracle Financial Services Institutional Performance Analytics (OFSIPA)

Oracle Financial Services Institutional Performance Analytics (OFSIPA) is a complete end-to-end web-based Business Intelligence solution for Customer Analytics.

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.

It enables you to actively plan, manage, and track marketing investments with pre-built reports, dashboards, and underlying data structures.

The OFSIPA solution is a part of Profitability Pack and is packaged along with AAI 8.0.4.0.0 and other applications. This OFSIPA is supported for Oracle 11g and 12c.

OFSIPA solution is built using:

- OBIEE 11.1.1.7.1 for Dashboard and Reports activities
- Essbase 11.1.2.3+ for 12c database

This manual deals with essential Oracle Financial Services Analytical Applications (OFSAA) Infrastructure required for OFSIPA activities, process flow for the data transformation and cube building processes, and functional details about the dashboards and reports. In addition, it includes subject areas which could be used for ad-hoc reporting using OBIEE Answers tool.

CHAPTER 2

Overview of Process Flow

This chapter discusses the following topics:

Introduction

Oracle Financial Services Institutional Performance Analytics (OFSIPA) 8.0.4.0.0 utilizes OBIEE technology to present:

- Behavioral and Engagement trends of its target segments - exposures, commitments, line utilization, assets/
- liabilities, deposits, withdrawals, fees, income, recent transactions, and so on.
- Performance of the business and underlying customers.
- Product holdings and across the organization (that is Corporate client and any of its sub-divisions or subsidiaries).
- Efficiency of the sales force in terms of ongoing customer revenue generation, cross-sell and up-sell, product usage, and pipeline.
- Efficiency of investments such as marketing, partner development, and so on.

Following diagram depicts the product objectives of OFSIPA 8.0.4.0.0:

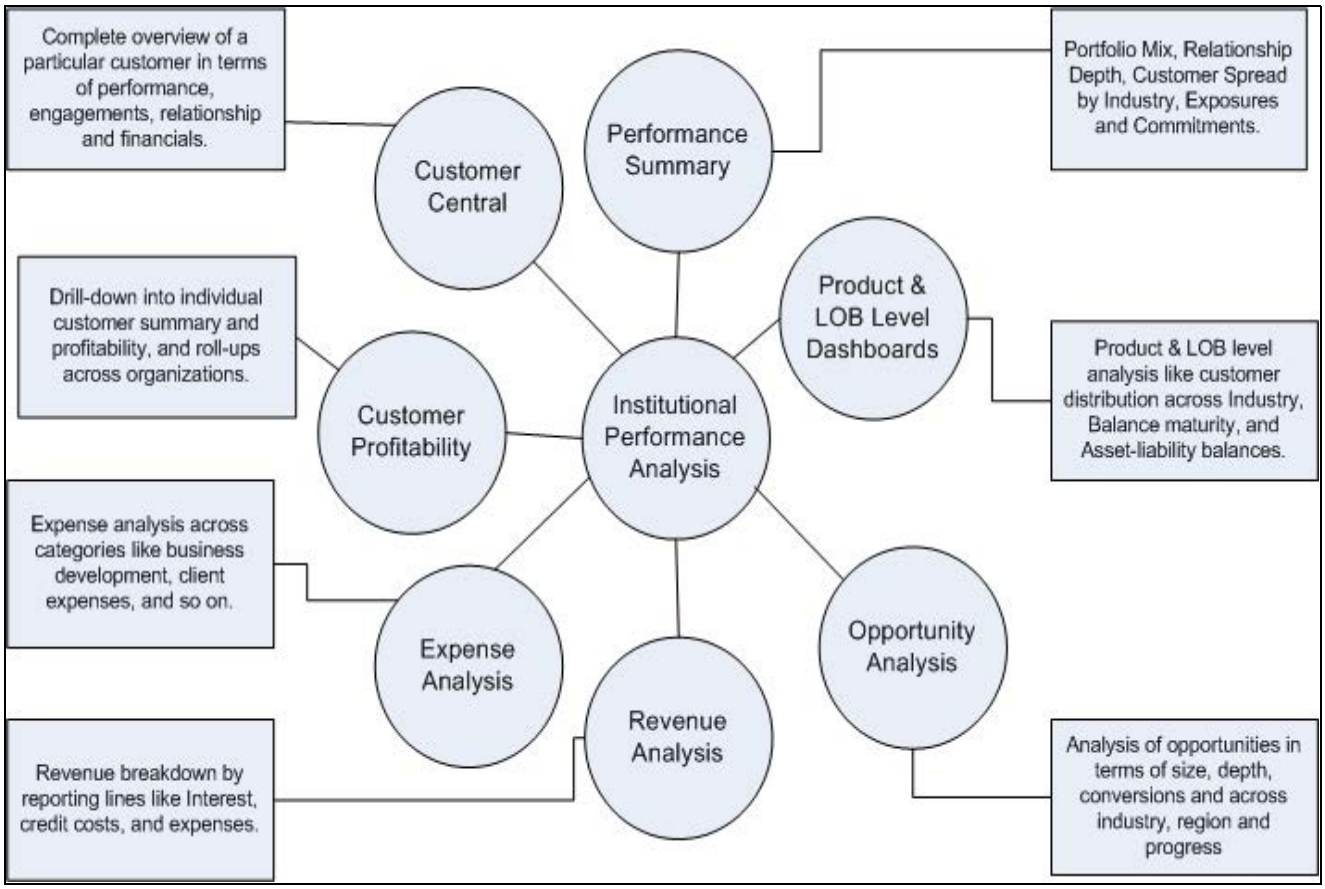


Figure 1. Product Objectives of OFSIPA

For details on OFSIPA reports and how OBIEE is being utilized, see *Overview of OFSIPA Reports*.

OFSIPA is designed for OBIEE reading data from relational database. The relational database comprises of various dimensions and facts in the BI data model. OFSIPA is also designed for OBIEE reading data from Essbase cubes, which stores aggregated data. The Essbase cubes are built from the fact data of the BI data model.

OFSIPA 8.0.4.0.0 can be independently licensed and installed to work on top of the OFSAAI 8.0.4.0.0 infrastructure.

Data Flow

Institutional Performance Analytics data model contains the staging tables from which data is loaded in to the dimensions and fact tables. Staging tables include the master staging tables, detail staging tables, staging product processor tables, and so on. The user has to populate data into these staging tables.

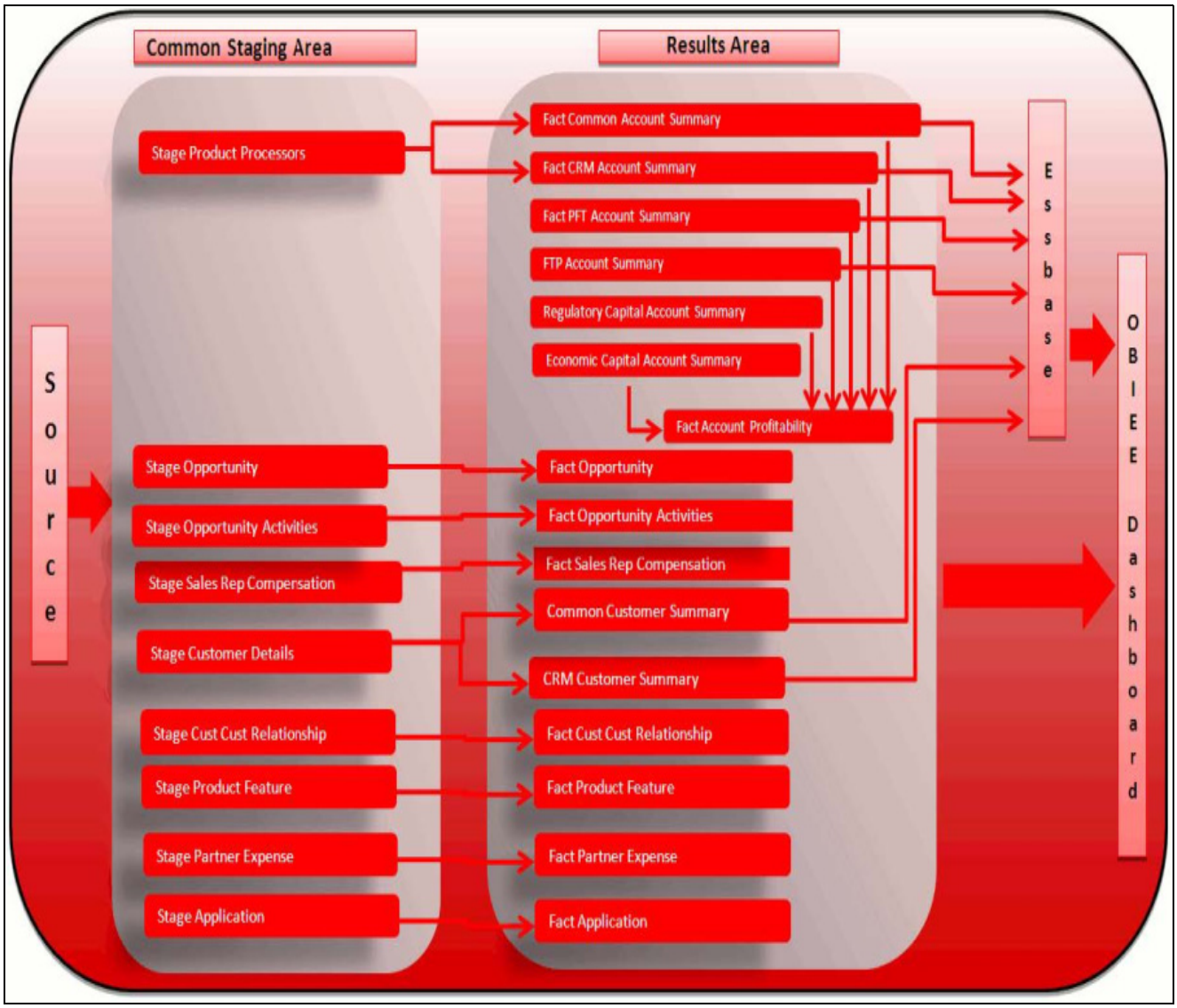


Figure 2. Staging Tables

Dimension Data Flow

Dimension data in OFSIPA application is loaded from staging master tables using the Slowly Changing Dimensions (SCD) process. Data from source systems can be loaded into staging through flat file or source system interfaces. SCD process tracks the changes in the dimensional attributes and loads data into dimension tables. Examples of dimension tables that follow the SCD process are Product, Customer Type, Customer, and so on.

Some dimensions are static or maintained internally within the application and are not expected as a download from source system. Examples of such dimensions are Reporting Line. These dimensions are maintained through the AMHM (Attribute Member Hierarchy Maintenance) component of OFSAAI or through other framework components like DEFI.

Following are the list of Dimensions used in OFSIPA:

Table 1. OFSIPA Dimensions

Dimension Entity Name	Staging Entity Name(s)	Loading/Maintenance method
Account Status Dimension	Stage Account Status Master	SCD
Application Reject Reasons Dimension	Stage Application Reject Reason Master	SCD
Application Type Dimension	Stage Application Type Master	SCD
Attrition Dimension	Stage Attrition Reason Master	SCD
Account Management Dimension	Stage Account Mgmt Master	SCD
Country Dimension	Stage Country Master	SCD
Credit Center Dimension	Stage Credit Center Master	SCD
Credit Officer Dimension	Stage Credit Officer Master	SCD
Customer Dimension	Stage Customer Master	DT
Customer Type Dimension	Stage Customer Type Master	SCD
Decision Status Dimension	Stage Decision Status Master	SCD
Deviation Reasons Dimension	Stage Deviation Reason Master	SCD
Education Dimension	Stage Customer Education Master	SCD
Geography Dimension	Stage Geography Master	SCD
Industry Dimension	Stage Industry Master	SCD
Management Dimension	Stage Account Mgmt Master	SCD
Migration Reasons Dimension	Stage Migration Reason Master	SCD
Offer Dimension	Stage Offer Master	SCD
Opportunity Dimension	Stage Opportunity	SCD
Opportunity Activity Type Dimension	Stage Activity Type Master	SCD
Organization Structure Dimension	Stage Organization Structure Dimension	SCD
Partner Dimension	Stage Partner Master	SCD
Product Dimension	Stage Product Master	SCD
Product Feature Dimension	Stage Product Feature Master	SCD

Table 1. OFSIPA Dimensions

Dimension Entity Name	Staging Entity Name(s)	Loading/Maintenance method
Product Type Dimension	Stage Product Type Master	SCD
Prospect Dimension	Stage Prospect Master	SCD
Reason Dimension	Stage Opportunity Win Loss Reason Master	SCD
Retention Offer Type Dimension	Stage Retention Offer Master	SCD
Sales Representative Dimension	Stage Sales Rep Master	SCD
Vendor Dimension	Stage Vendor Master	SCD
Vintage Dimension	Stage Vintage Master	SCD
Line of Business Dimension	Stage Line of Business Master	SCD
Common Chart Of Accounts Dimension	Common COA Dimension Members, Common COA Hierarchies, Common COA Member Attributes, Common COA Member Translations	SCD
General Ledger Account Dimension	General Ledger Member Attributes, General Ledger Dimension Members, General Ledger Hierarchies, General Ledger Member Translations	SCD
DIM_ORG_UNIT	Organization Unit Member Attributes, Organization Unit Dimension Members, Organization Unit Hierarchies, Organization Unit Member Translations	SCD
Product Dimension	Product Member Attributes, Product Dimension Members, Product Hierarchies, Product Member Translations	SCD
Reporting Line Dimension	Reporting Line Dimension Members, Reporting Line Member Translation, Reporting Line Member Attributes, Reporting Line Hierarchies	AMHM/DT

Table 1. OFSIPA Dimensions

Dimension Entity Name	Staging Entity Name(s)	Loading/Maintenance method
Band Dimension	Band Dimension Members, Band Member Translation, Band Member Attributes	AMHM/SCD **Note:** When updating DIM_BANDS, the lower bound of one band can not start with the upper bound of the previous band. For example, for a Customer Balance band, if the upper bound of the first band is 10,000 USD, the lower bound of the next band must start with 10,000.01 USD, if the dataload convention being followed is for two decimal points. In case of integer bands, for example, Number of Transactions; if the upper bound of a band ends with 5, the lower bound of the next band must begin with 6.
Region Dimension		Direct Load
Acquisition Channel Dimension	Stage Sales Channel Master	SCD
Instrument Category Dimension		Seeded
Currency Dimension		Seeded
Consolidation Dimension		Seeded
Calendar Dimension		DT
Account Dimension	Stage LC Contracts	SCD
	Stage Commitment Contracts	SCD
Party Dimension	Stage Party	SCD
Location Dimension	Stage Location Master	SCD

Table 1. OFSIPA Dimensions

Dimension Entity Name	Staging Entity Name(s)	Loading/Maintenance method
Account Dimension	Stage Stage OD accounts	SCD
	Stage Stage TD contracts	SCD
	Stage Stage Trusts	SCD
	Stage Stage Loan Contracts	SCD
	Stage Stage Mutual Funds	SCD
	Stage Bills Contracts	SCD
	Stage CASA Accounts	SCD
	Stage Guarantees	SCD
	Stage Stage leases contracts	SCD
	Stage Stage mm contracts	SCD
	Stage Annuity Contracts	SCD
	Stage Borrowings, Stage Card Accounts	SCD
	Stage Investments	SCD

Some of the stage data can also come from master data management interfaces. In such a case, data from interface is loaded into staging interface tables and SCD is run on the interface tables. Mapping of dimensional attributes to staging can be obtained by querying SYS_STG_JOIN_MASTER and SYS_TBL_MASTER table in the atomic schema.

Key dimensions for reporting

The following key dimensions are required for OFSIPA reporting as these dimensions are being directly consumed by the reports.

- Opportunity Activity Type Dimension
- Attrition Dimension
- Bands Dimension
- Acquisition Channel Dimension
- Consolidation Dimension
- Currency Dimension
- Customer Dimension
- Customer Type Dimension
- Date Dimension
- Geography Dimension
- Account Dimension
- Industry Dimension

- Line of Business Dimension
- Account Management Dimension
- Migration Reasons Dimension
- Dimension
- Organization Structure Dimension
- Org Unit BI Hierarchy
- Partner Dimension
- Product Dimension
- Product Type Dimension
- Product Family Holding Dimension
- Prospect Dimension
- Reporting Line Dimension
- Run Dimension
- Sales Representative Dimension
- Sales Stage Dimension
- Vintage Dimension
- Location Dimension

Fact Data Flow

Most of the Fact tables are mapped to staging counterparts through Table to Table (T2T) mappings. Data from source systems can be loaded into staging through flat file or source system interfaces. T2T process then loads data to fact tables. Examples include Fact Common Account Summary, Fact Opportunity, and so on. Some of the Fact tables are loaded with processed fact information from other fact tables. Examples include Fact CRM Customer Summary, Fact Account Profitability, and so on.

Table 2. Fact Table Flow

Fact Entity Name	Source	Source Entities	Method of populating measures
Fact Common Account Summary	Stage	Stage Annuity Contracts, Stage Bill Contracts, Stage Borrowings, Stage Cards, Stage CASA Accounts, Stage Guarantees, Stage Investments, Stage LC Contracts, Stage Leases Contracts, Stage Loan Contracts, Stage Money Market Contracts, Stage Over Draft Accounts, Stage Term Deposit, Stage Trusts, Stage Commitment Contracts, Stage Mutual Funds	T2T
Fact PFT Account Summary	Instrument	Annuity Contracts, Borrowings, Checking and Savings Account, Credit Cards, Credit Lines, Guarantees, Investments, Leases, Loan Contracts, Mortgages, Term Deposits, Trusts Stage Mutual Funds	T2T
Fact FTP Account Summary	Instrument	Annuity Contracts, Borrowings, Checking and Savings Account, Credit Cards, Credit Lines, Guarantees, Investments, Leases, Loan Contracts, Money Market Contracts, Mortgages, Term Deposits, Trusts Stage Mutual Funds	T2T

Table 2. Fact Table Flow

Fact Entity Name	Source	Source Entities	Method of populating measures
Fact CRM Account Summary	Stage	Stage Annuity Contracts, Stage Bill Contracts, Stage Borrowings, Stage Cards, Stage CASA Accounts, Stage Guarantees, Stage Investments, Stage LC Contracts, Stage Leases Contracts, Stage Loan Contracts, Stage Money Market Contracts, Stage Over Draft Accounts, Stage Term Deposit Contracts, Stage Trusts, Stage Commitment Contracts	T2T
Fact Common Customer Summary	Stage	Stage Customer Details, Stage Party Rating Details, Stage Party Financials	T2T
Fact CRM Customer Summary	Stage and Fact	Stage Customer Master, Stage Customer Details, Fact Common Account Summary	T2T
Fact Account Feature Map	Stage	Stage Account Feature Map	T2T
Fact Customer to Customer Relationship	Stage	Stage Customer to Customer Relationships	T2T
Fact Opportunity	Stage	STG_OPPORTUNITY	T2T
FCT_OPPORTUNITY_ACTIVITY	Stage	STG_OPPORTUNITY_ACTIVITY	T2T
Fact Account Profitability	Fact	Fact Common Account Summary, Fact FTP Account Summary, Fact PFT Account Summary, Fact Regulatory Capital Account Summary, Fact Economic Capital Account Summary	DT
Fact Account Customer Relationship	Stage	Stage Customer Relationships	T2T
Fact Account Manager Relationship	Stage	Stage Account Manager Relationship	T2T
Fact Forecast And Plan Data	Stage	Stage Forecast and Plan Data	
Exchange Rate History	Stage	Stage Exchange Rates	T2T
Exchange rates	View	View on Stage Exchange Rates	T2T

Table 2. Fact Table Flow

Fact Entity Name	Source	Source Entities	Method of populating measures
Fact Party Account Role Map	Stage	Stage Party Account Role Map	T2T
Fact Party Financials	Stage	Stage Party Financials	T2T
Fact Account Segment MOB Summary	Fact	Fact Account Profitability, Fact Common Account Summary, Fact Account Segment Score	DT
Fact Account Segment Score	Fact	Fact Common Account Summary	DT

The OFSIPA uses some materialized views registered as "Derived Entity", that has to be refreshed as and when the dependent table has fresh data. The MVs can be refreshed by running the batches crated for the purpose.

The list of Derived Entity and the dependent objects can be found in the following table.
Summary, Fact Account Profitability, and so on.

Table 3. Derived Entity and Dependent Objects

Materialized View	Referenced Name	Referenced Object Type
ACNTSMRM	FCT_COMMON_ACCOUNT_SUMMARY	Table
	FCT_CRM_ACCOUNT_SUMMARY	Table
CUSTDETM	DIM_CUSTOMER	Table
	DIM_CUSTOMER_TYPE	Table
	DIM_GENDER	Table
	FCT_COMMON_CUSTOMER_SUMMARY	Table
FCSTCUSA	VW_ACCT_VAL_FCST_CUSTAGG_IPA	Table
FCSTLTVM	VW_FORECAST_LTV_IPA	Table
FCSTREPA	VW_ACCT_VAL_FCST_REPAGG_IPA	Table
FSIUSRD	FSI_USER_DATA_ACCESS	Table
MGMTPFTM	ACNTSMRM	Table
	FCT_ACCOUNT_MGR_REL	Table
	FCT_ACCOUNT_PROFITABILITY	Table

Table 3. Derived Entity and Dependent Objects

Materialized View	Referenced Name	Referenced Object Type
MVCACPRO	A_DIM_REP_CURRENCY	Table
	DIM_ACCOUNT	Table
	DIM_CONSOLIDATION	
	DIM_CURRENCY	Table
	DIM_CUSTOMER	Table
	DIM_CUSTOMER_TYPE	Table
	DIM_DATES	Table
	DIM_LOB	Table
	DIM_ORG_UNIT	Table
	DIM_PRODUCT	Table
	DIM_REP_LINE	Table
	FCT_COMMON_CUSTOMER_SUMMARY	Table
	FCT_CRM_ACCOUNT_SUMMARY	Table
	MVUSRACC	Table
MVCCUSAG	A_DIM_REP_CURRENCY	Table
MGMTPFTM	DIM_ACCOUNT	Table
MVCCUSAG	DIM_CONSOLIDATION	Table
	DIM_CURRENCY	Table
	DIM_CUSTOMER	Table
	DIM_CUSTOMER_TYPE	Table
	DIM_DATES	Table
	DIM_LOB	Table
	DIM_ORG_UNIT	Table
	DIM_PRODUCT	Table
	DIM_REP_LINE	Table
	FCT_ACCOUNT_PROFITABILITY	Table
	FCT_COMMON_CUSTOMER_SUMMARY	Table
	FCT_CRM_ACCOUNT_SUMMARY	Table
	MVUSRACC	Table

Table 3. Derived Entity and Dependent Objects

Materialized View	Referenced Name	Referenced Object Type
MVCPROAG	A_DIM_REP_CURRENCY	Table
	DIM_ACCOUNT	Table
	DIM_CONSOLIDATION	Table
	DIM_CURRENCY	Table
	DIM_CUSTOMER	Table
	DIM_CUSTOMER_TYPE	Table
	DIM_DATES	Table
	DIM_LOB	Table
	DIM_ORG_UNIT	Table
	DIM_PRODUCT	Table
	DIM_REP_LINE	Table
	DIM_VINTAGE	Table
	FCT_ACCOUNT_PROFITABILITY	Table
	FCT_COMMON_CUSTOMER_SUMMARY	Table
	FCT_CRM_ACCOUNT_SUMMARY	Table
	MVUSRACC	Table
MVUSRACC	DIM_ACCOUNT	Table
	FCT_COMMON_ACCOUNT_SUMMARY	Table
	FSIUSRD	Table
USRMGRMV	FSI_M_USER_MANAGER_MAP	Table
WTHREPMV	WITH_REP_LINE_DIRECT_INDIRECT	Table

Execute the batches `<INFODOM>_FN_REFRSH_DE` – Task1 to `<INFODOM>_FN_REFRSH_DE` - Task8 for refreshing the derived entities. The DT <INFODOM>_FN_REFRSH_DE is invoked from this task. This function refreshes the derived entities (materialized views) when ever the task is executed.

Note: If user gets *Runtime Exception* error while accessing the **Derived Entity** screen, user is required to update the java setting by adding OFSAAI URI in the Exception Site List.

BI Data Model

The BI data model is a star schema for the fact table FCT_<APPLICATION>_ACCOUNT_SUMMARY.

Following are the subject areas in ERwin data model:

- Fact Account Feature Map

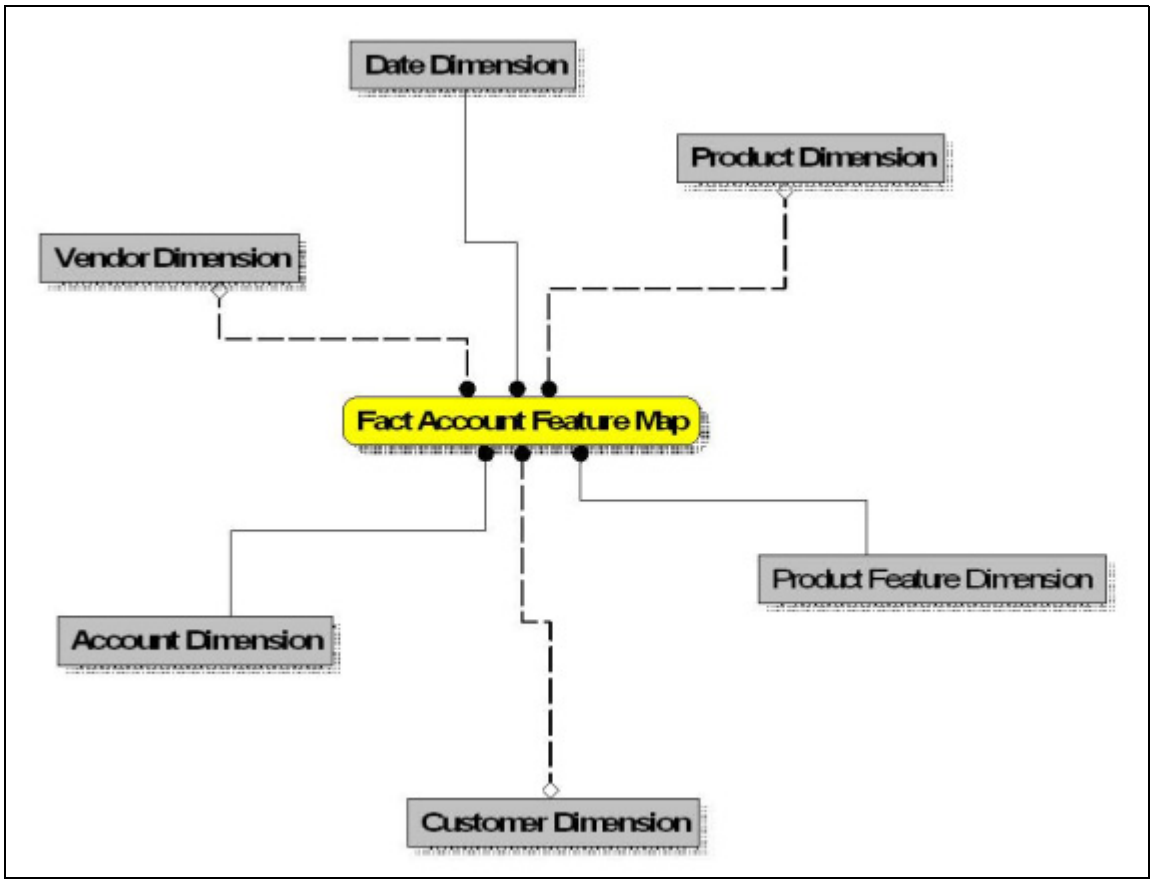


Figure 3. Fact Account Feature Map

- Fact Account Manager Relationship

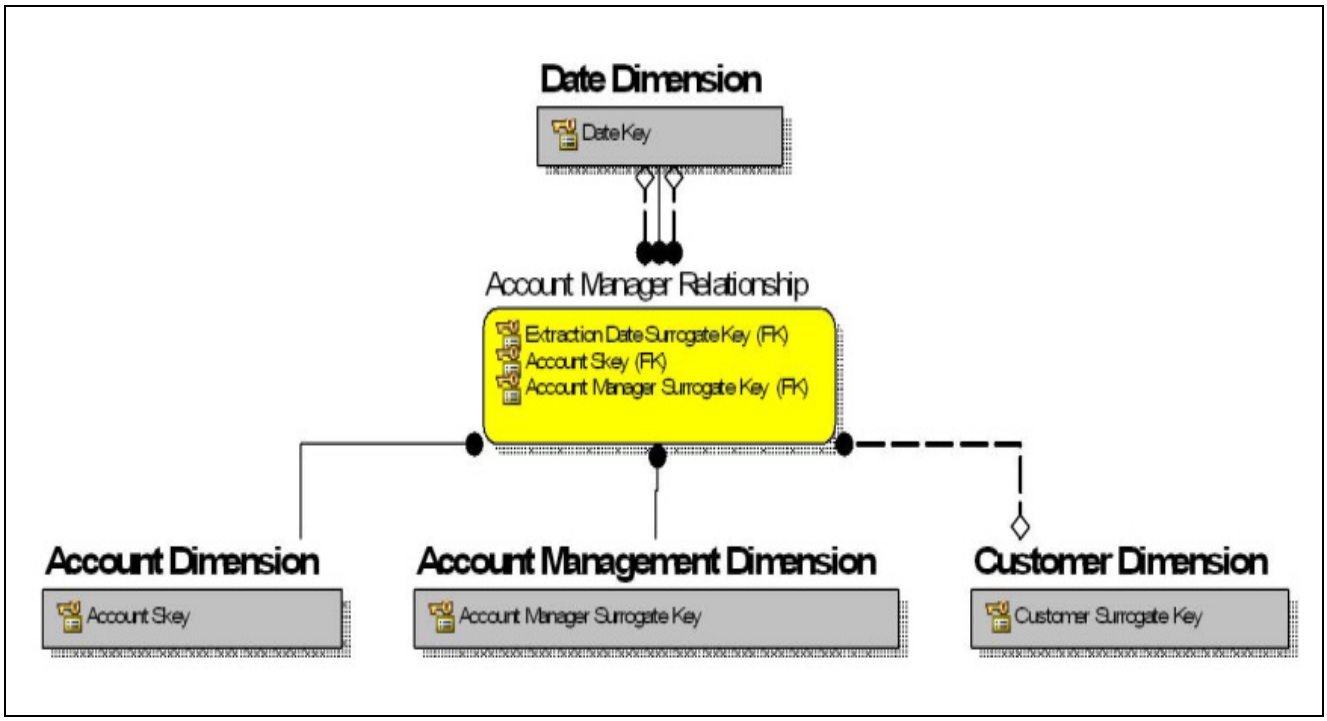


Figure 4. Fact Account Manager Relationship

- Fact Account Party Role

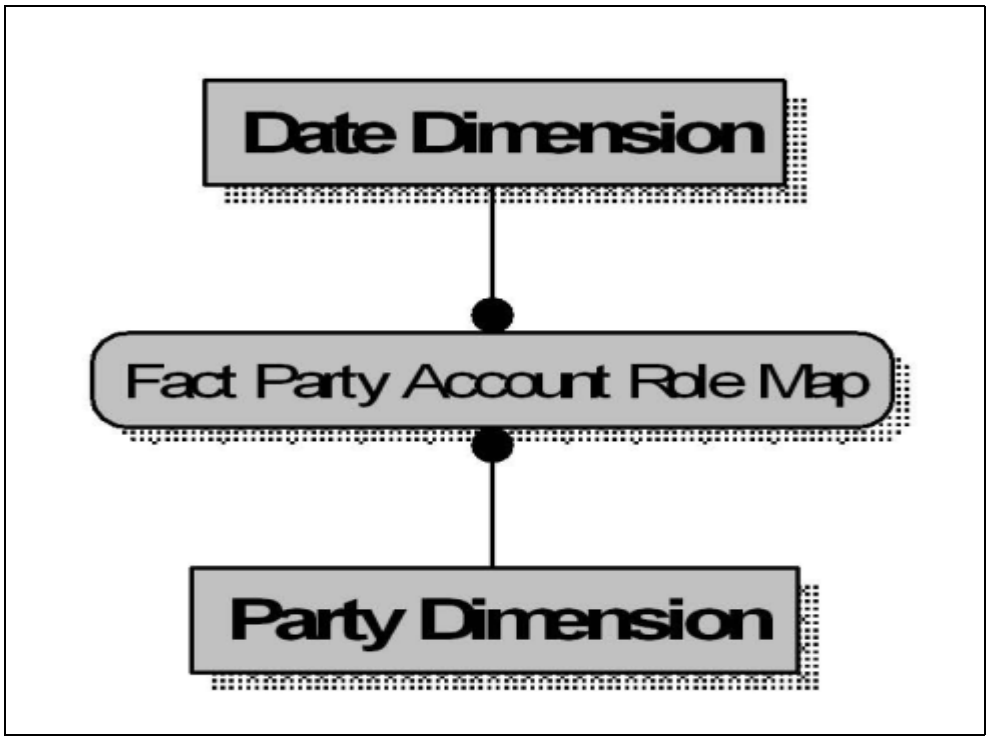


Figure 5. Fact Account Party Role

- Fact Account Profitability

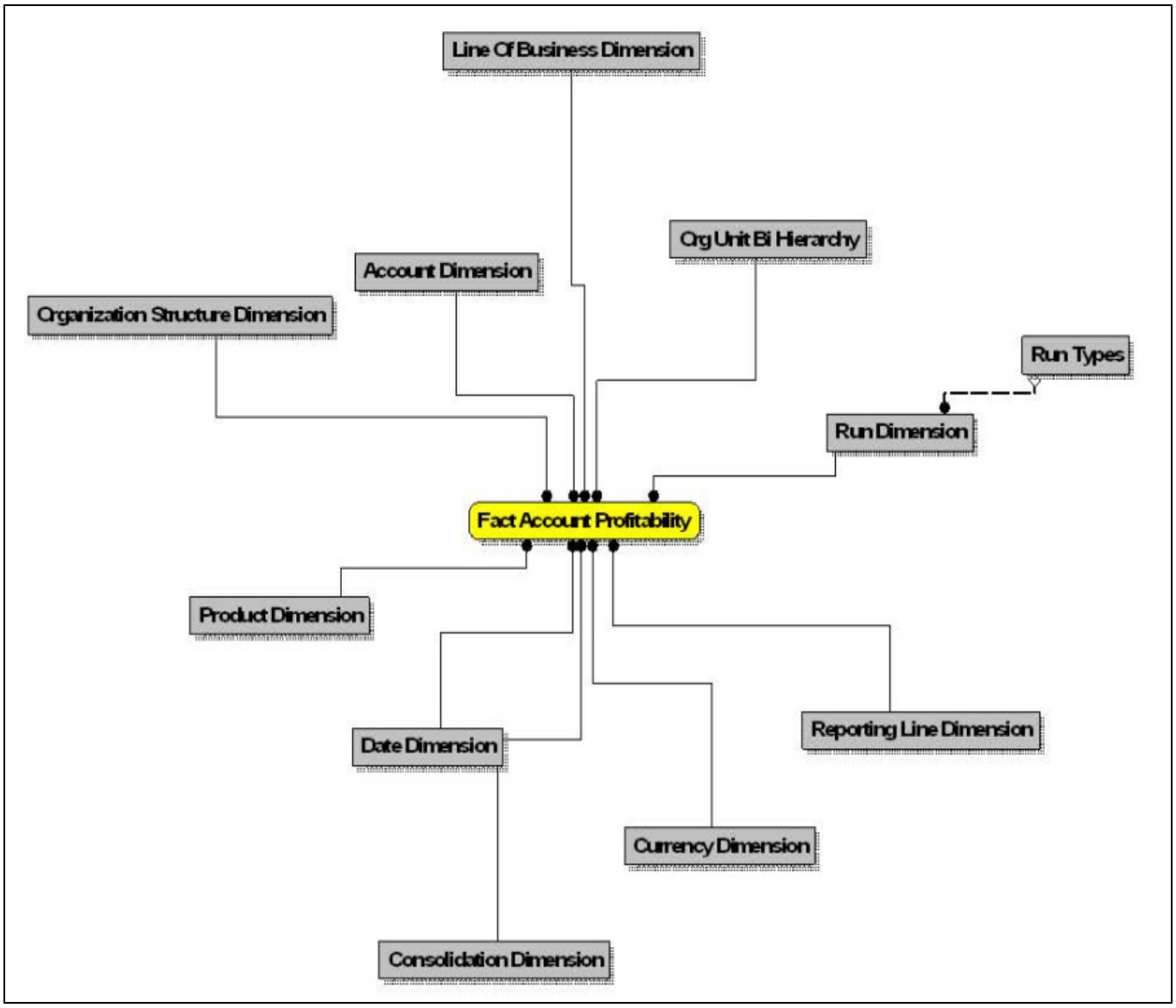


Figure 6. Fact Account Profitability

- Fact Account Segment MOB Summary

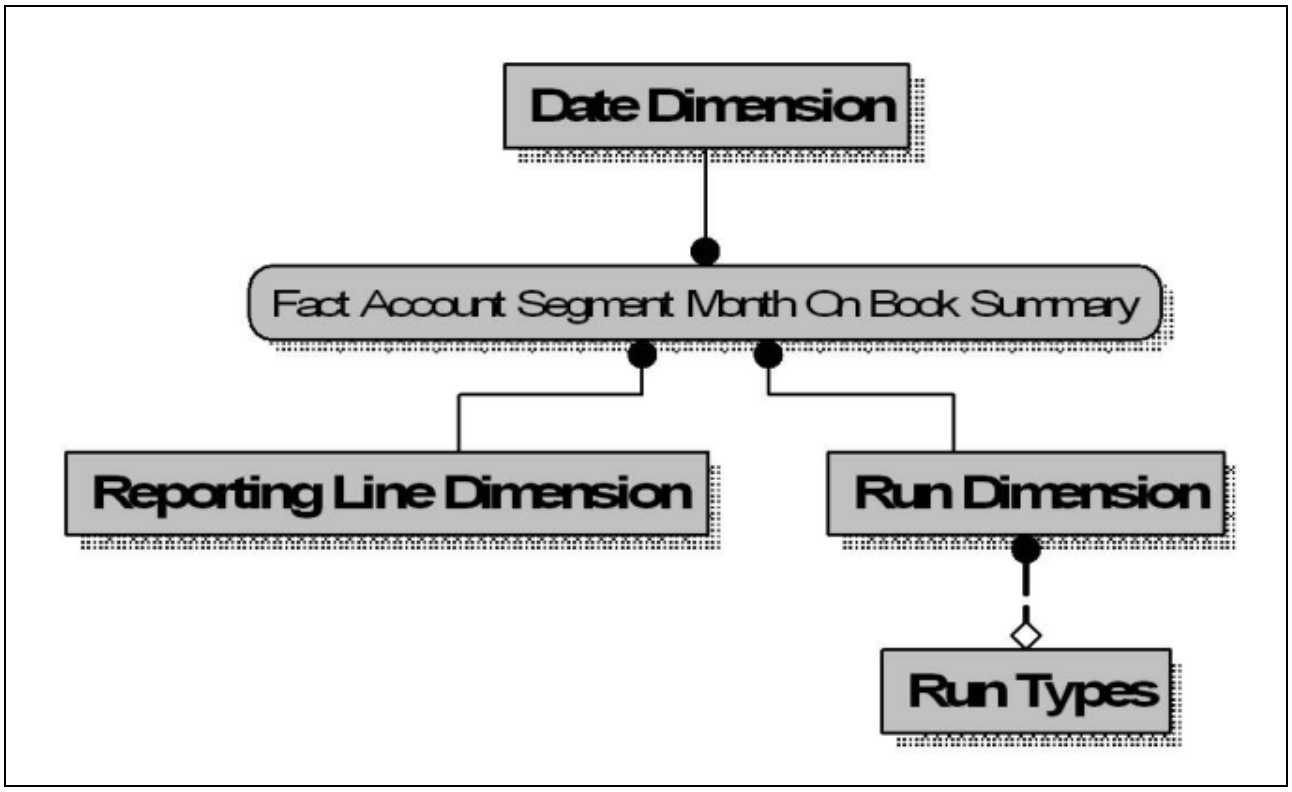


Figure 7. Fact Account Segment MOB Summary

- Fact Account Segment Score

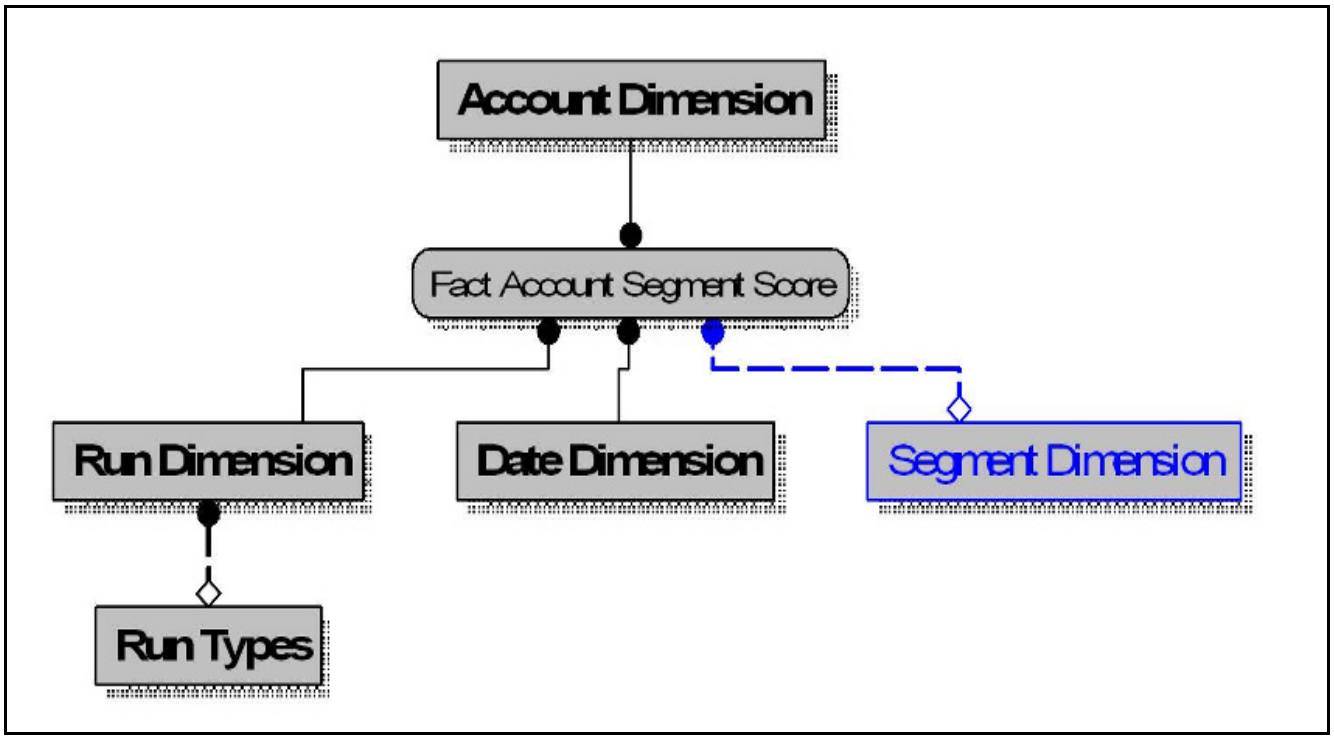


Figure 8. Fact Account Segment Score

- Fact Applications Summary

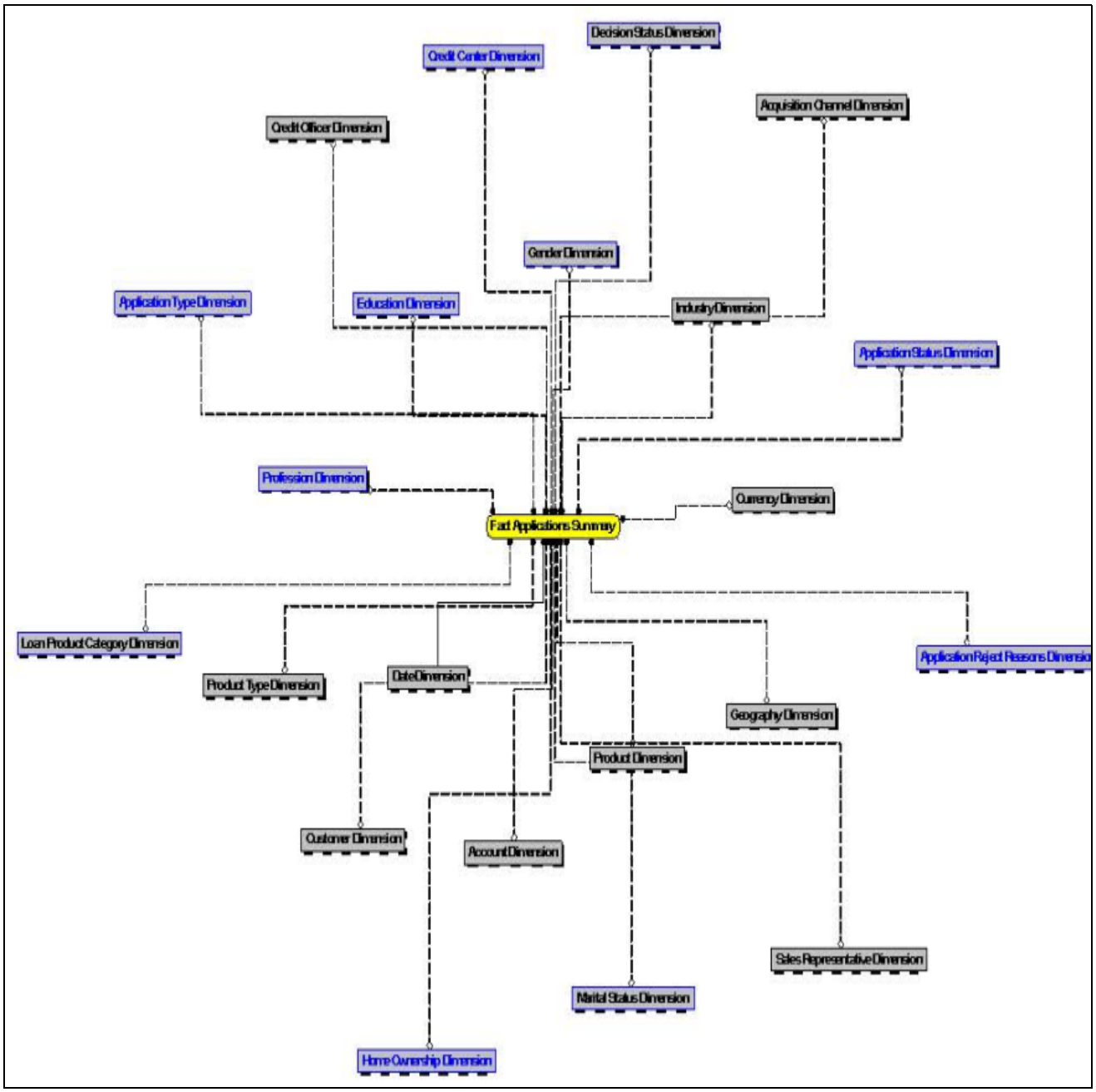


Figure 9. Fact Applications Summary

- Fact Common Account Summary

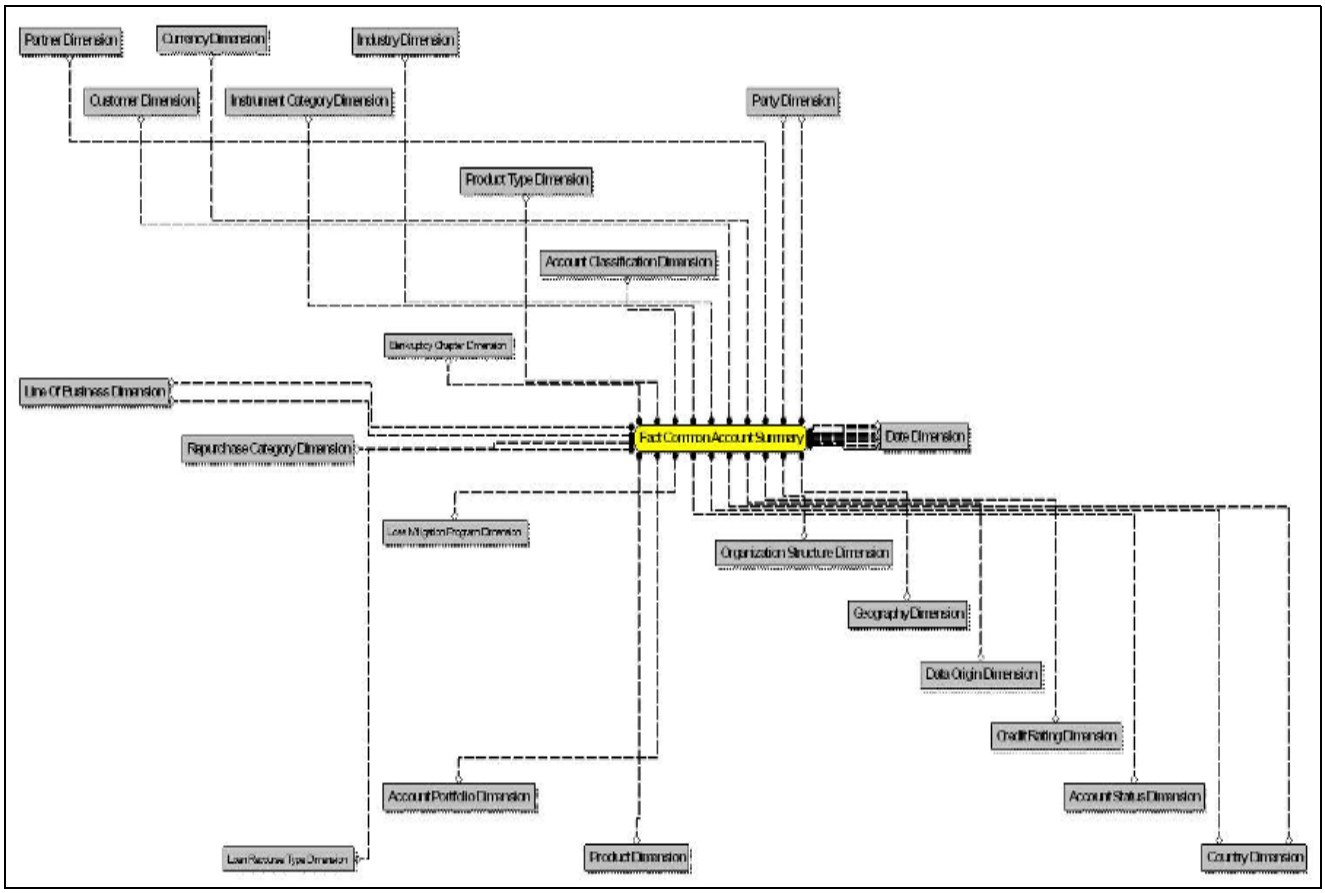


Figure 10. Fact Common Account Summary

- Fact Common Customer Summary

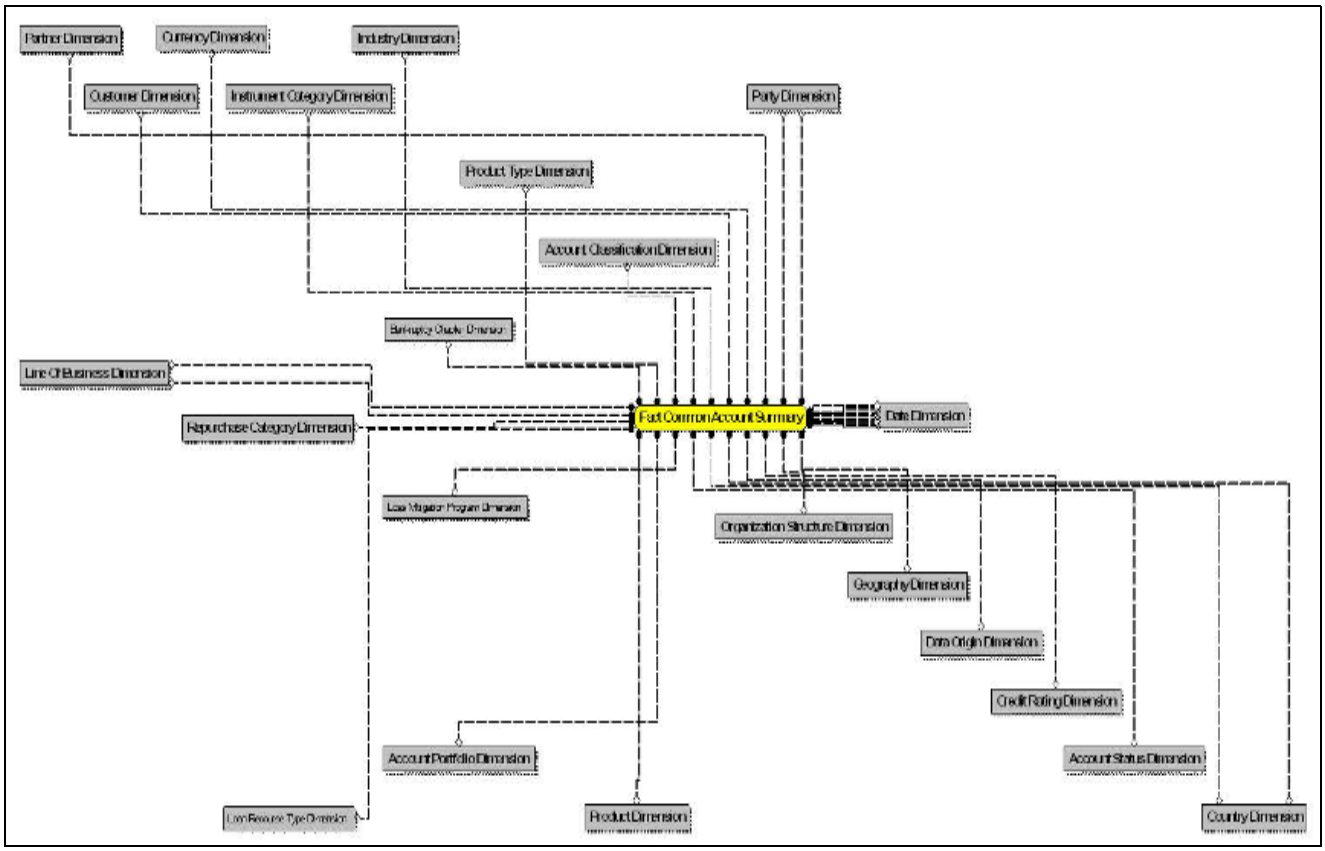


Figure 11. Fact Common Customer Summary

- Fact CRM Account Summary

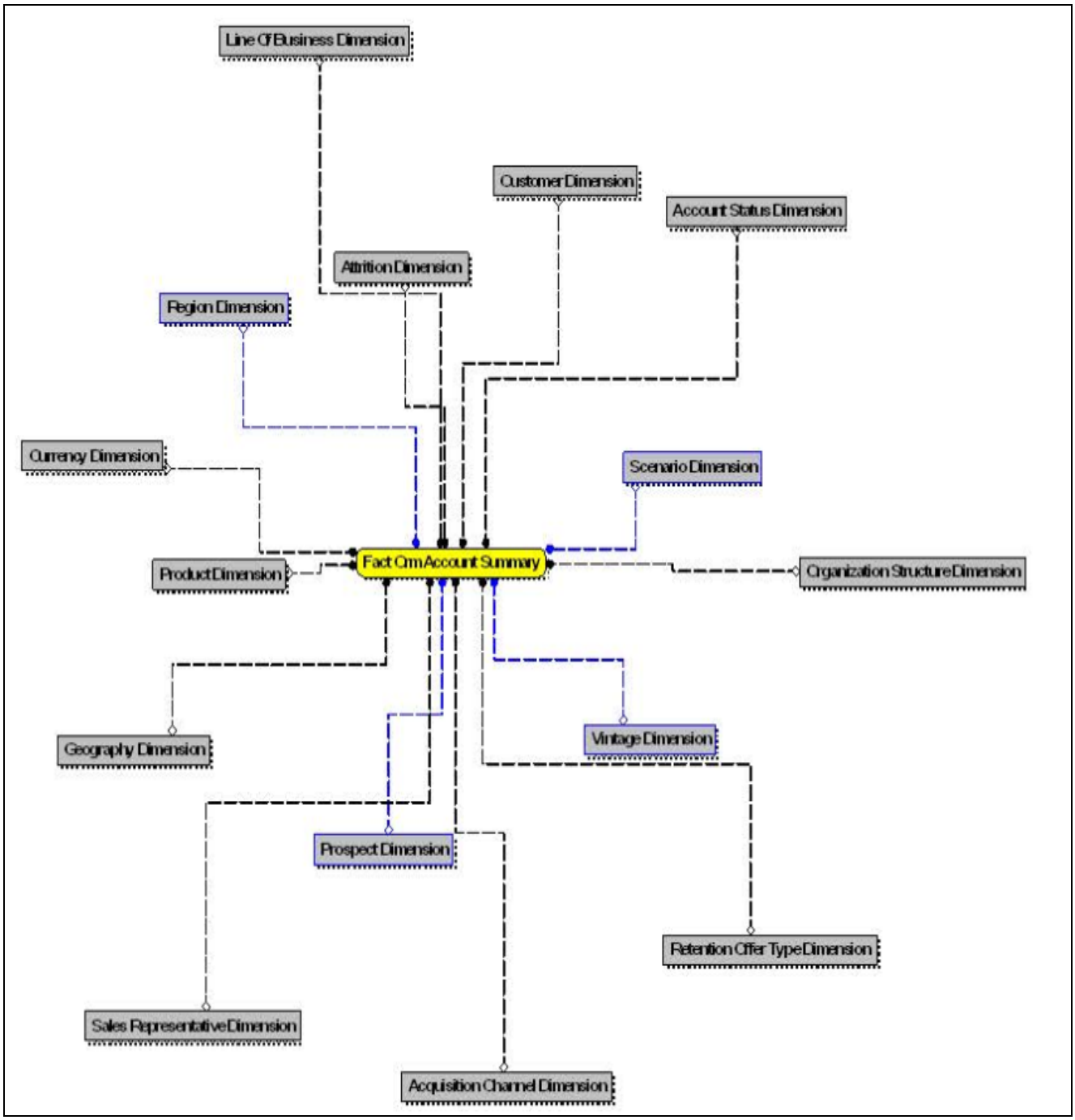


Figure 12. Fact CRM Account Summary

- Fact Cust Cust Relationship

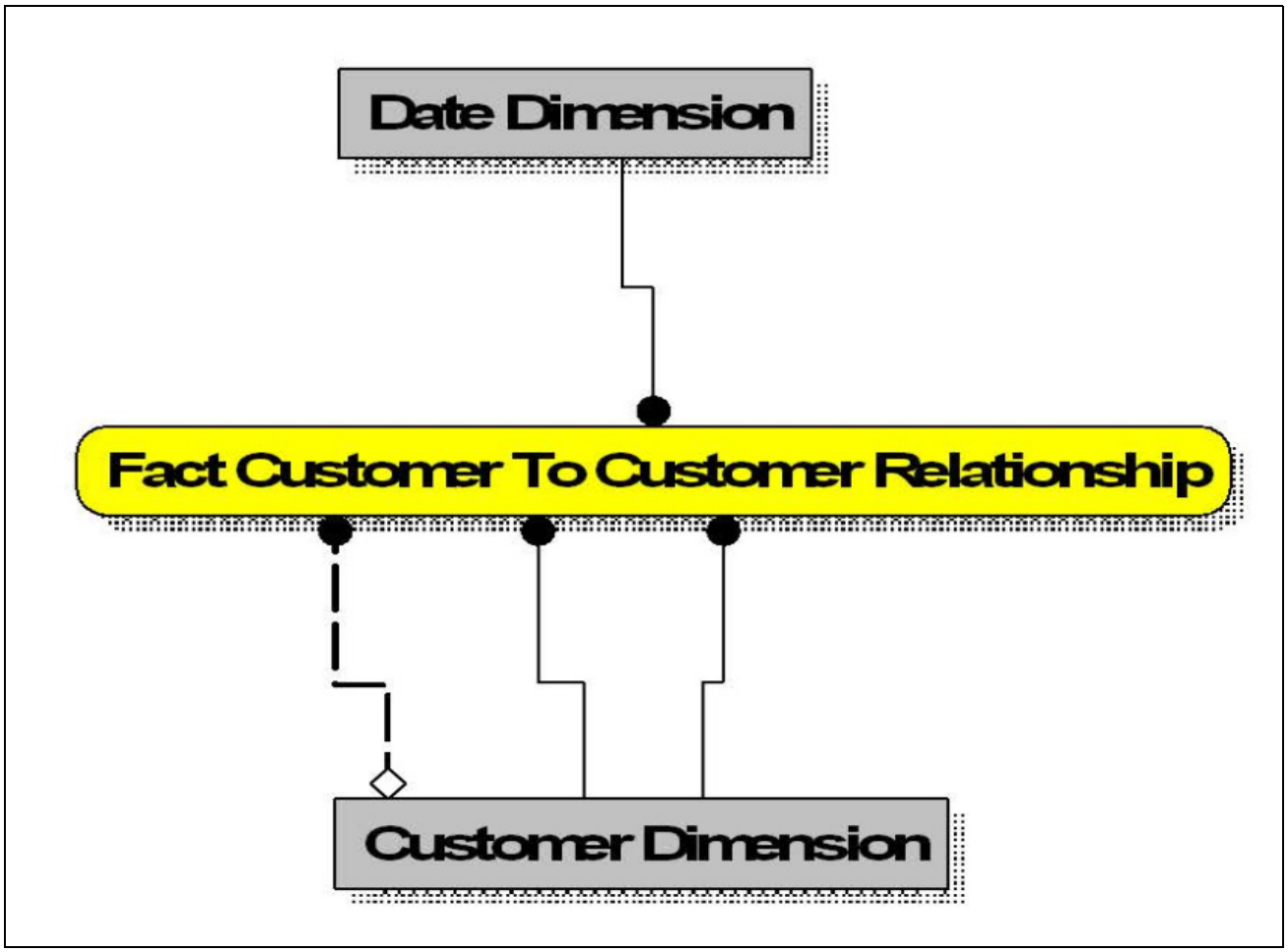


Figure 13. Fact Cust Cust Relationship

- Fact Eco Cap Account Summary

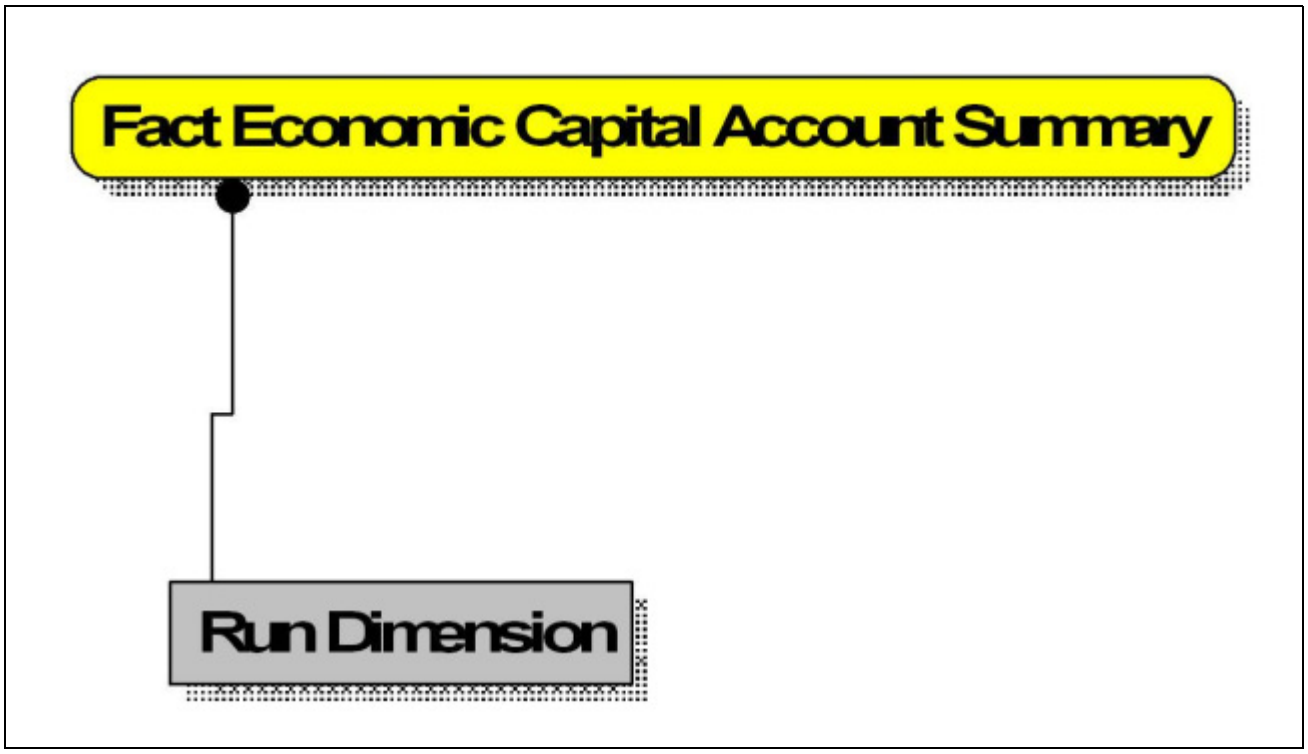


Figure 14. Fact Eco Cap Account Summary

- Fact Opportunity

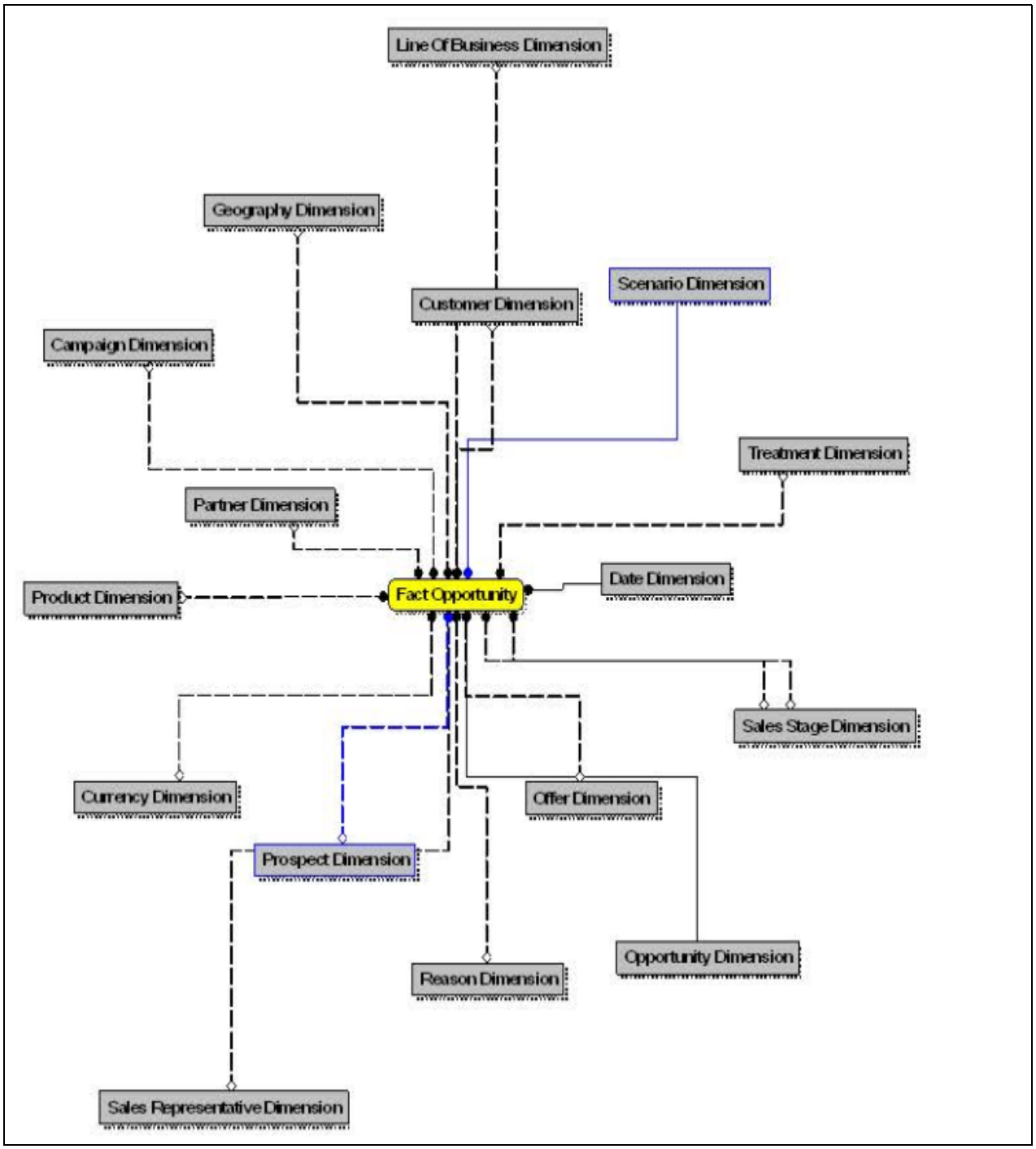


Figure 15. Fact Opportunity

- Fact Opportunity Activity

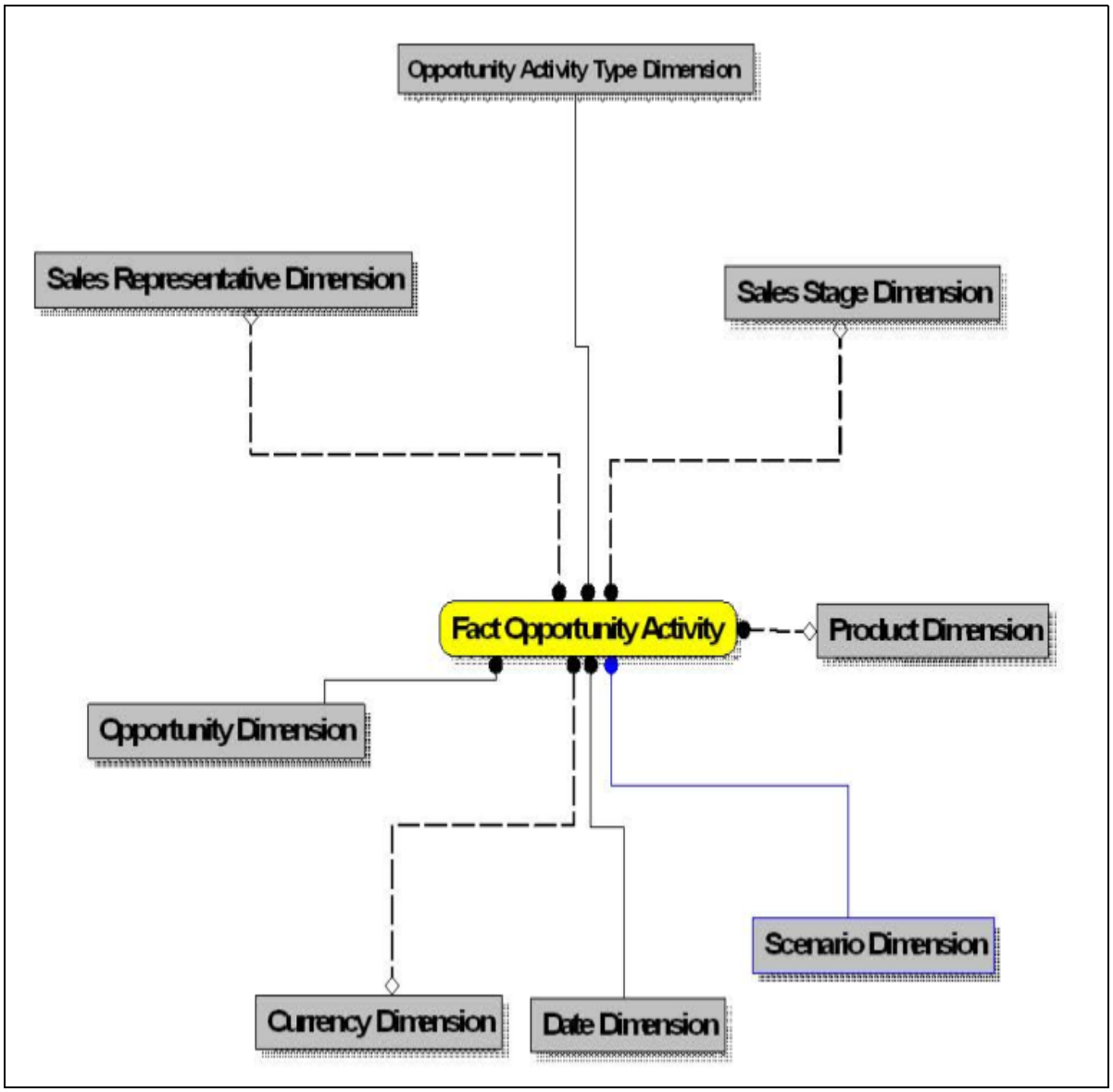


Figure 16. Fact Opportunity Activity

- Fact Reg Cap Account Summary

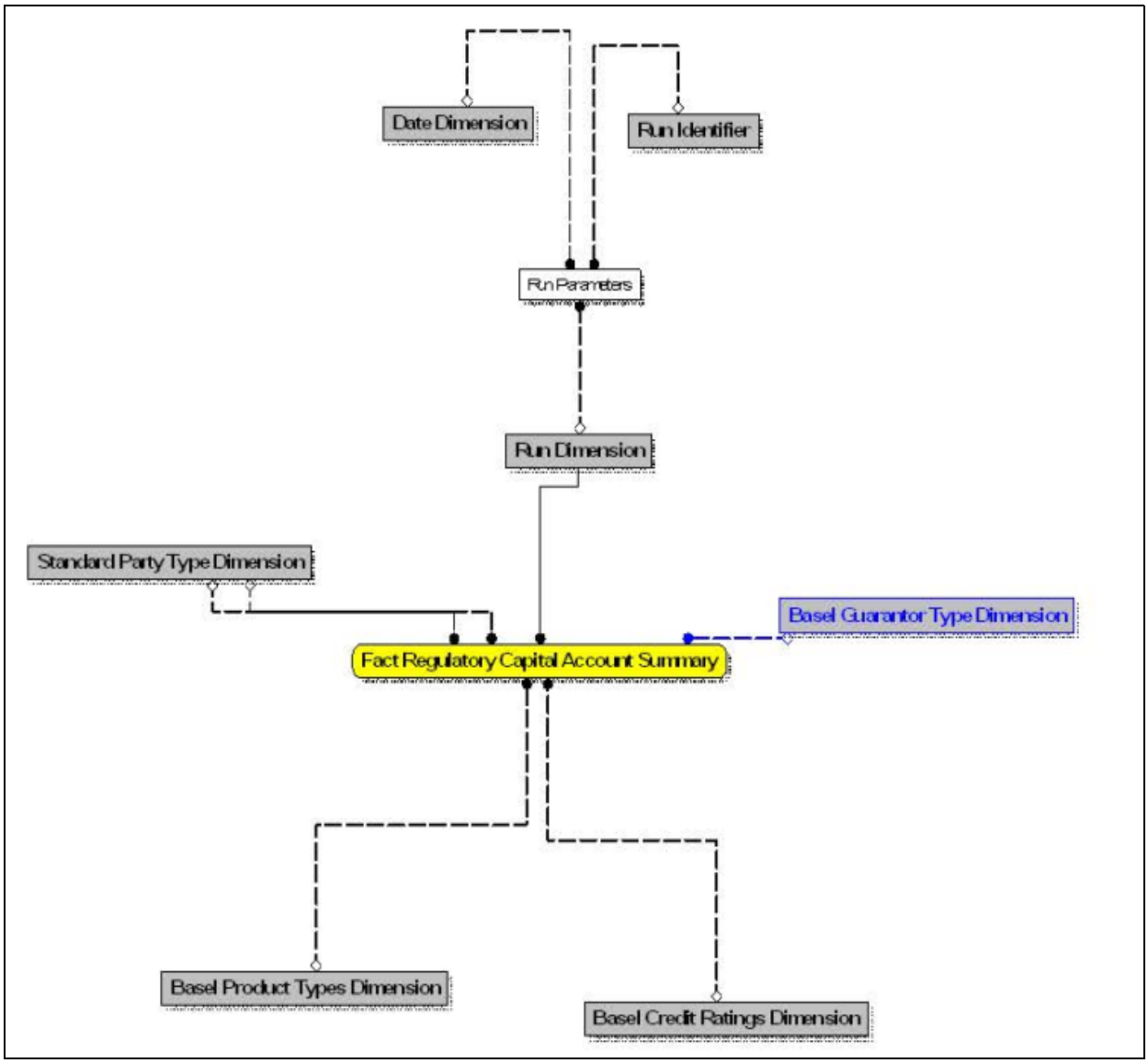


Figure 17. Fact Reg Cap Account Summary

- Fact Sales Representative Compensation

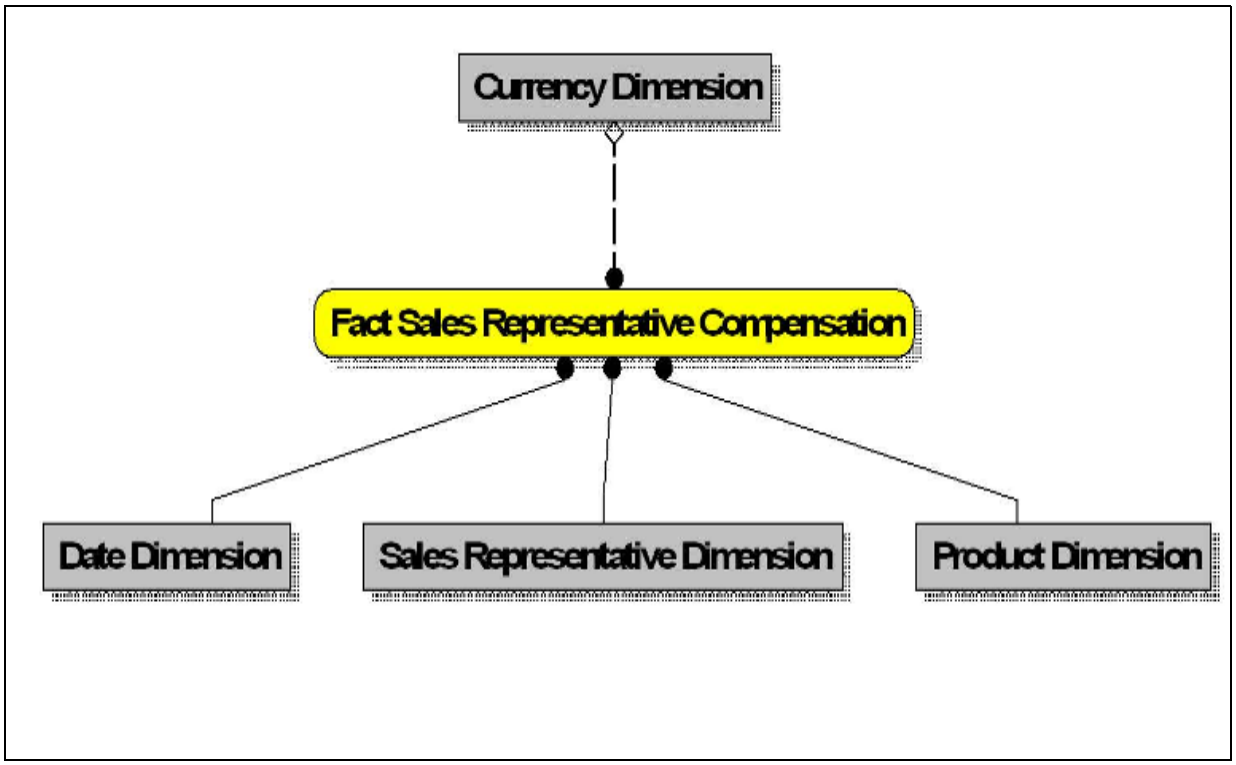


Figure 18. Fact Sales Representative Compensation

- Fact Transaction Summary

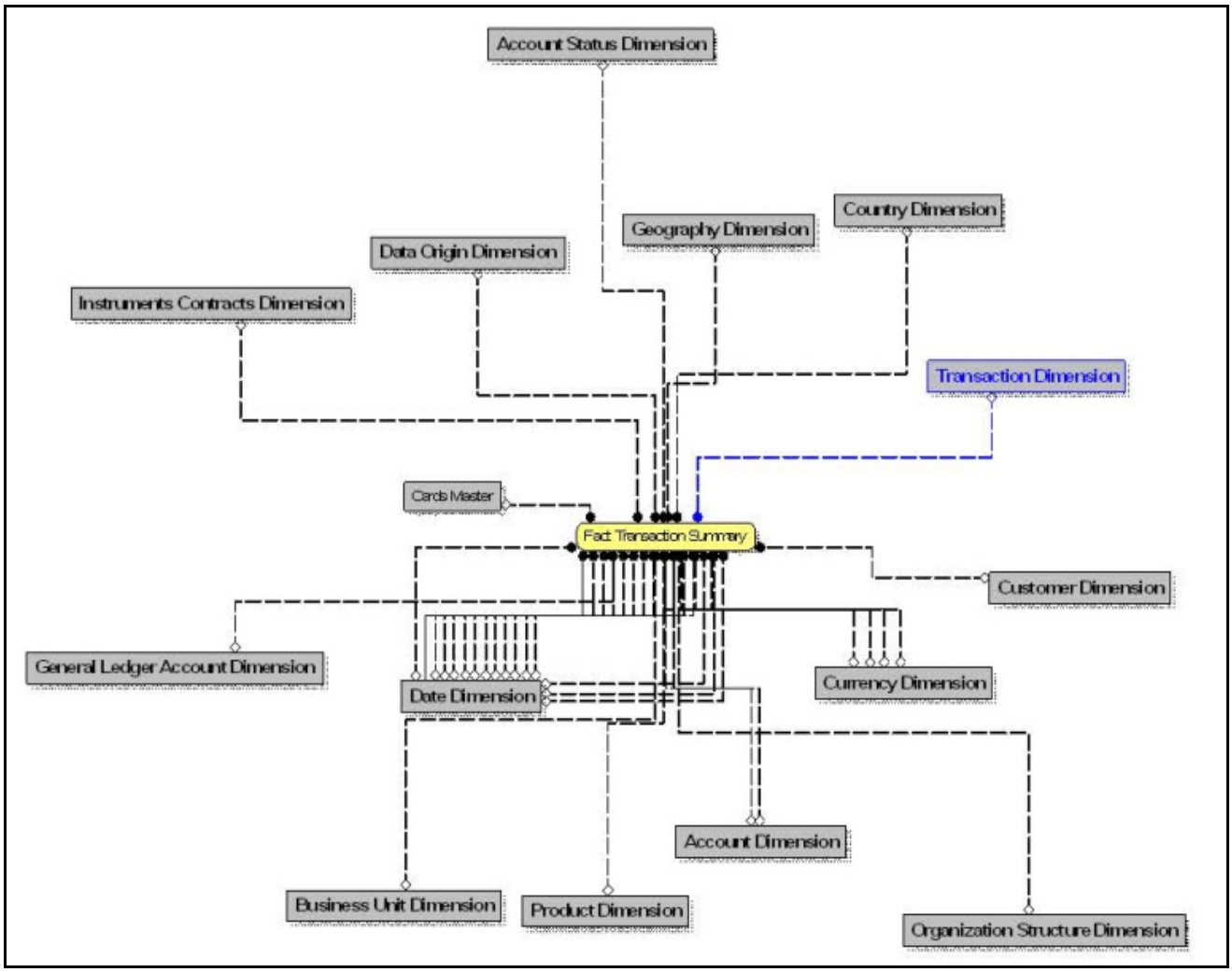


Figure 19. Fact Transaction Summary

- FTP Account Summary

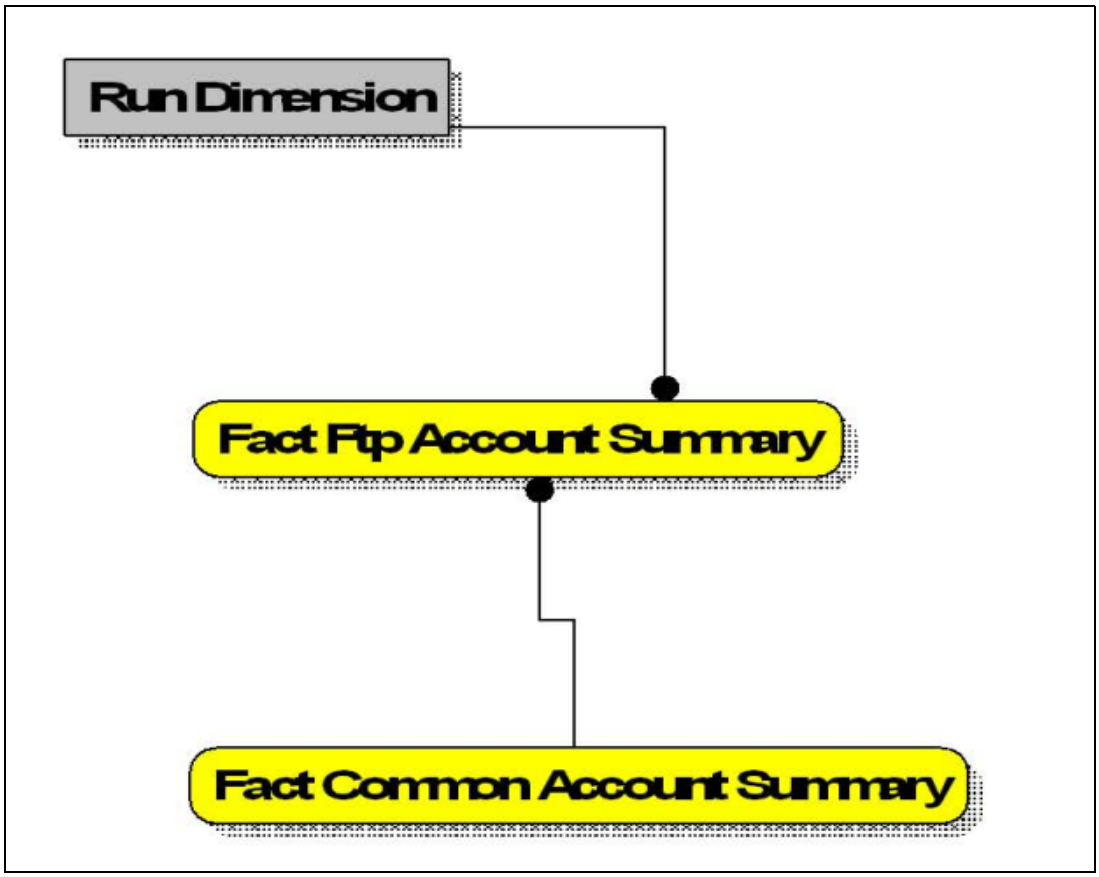


Figure 20. FTP Account Summary

- PFT Account Summary

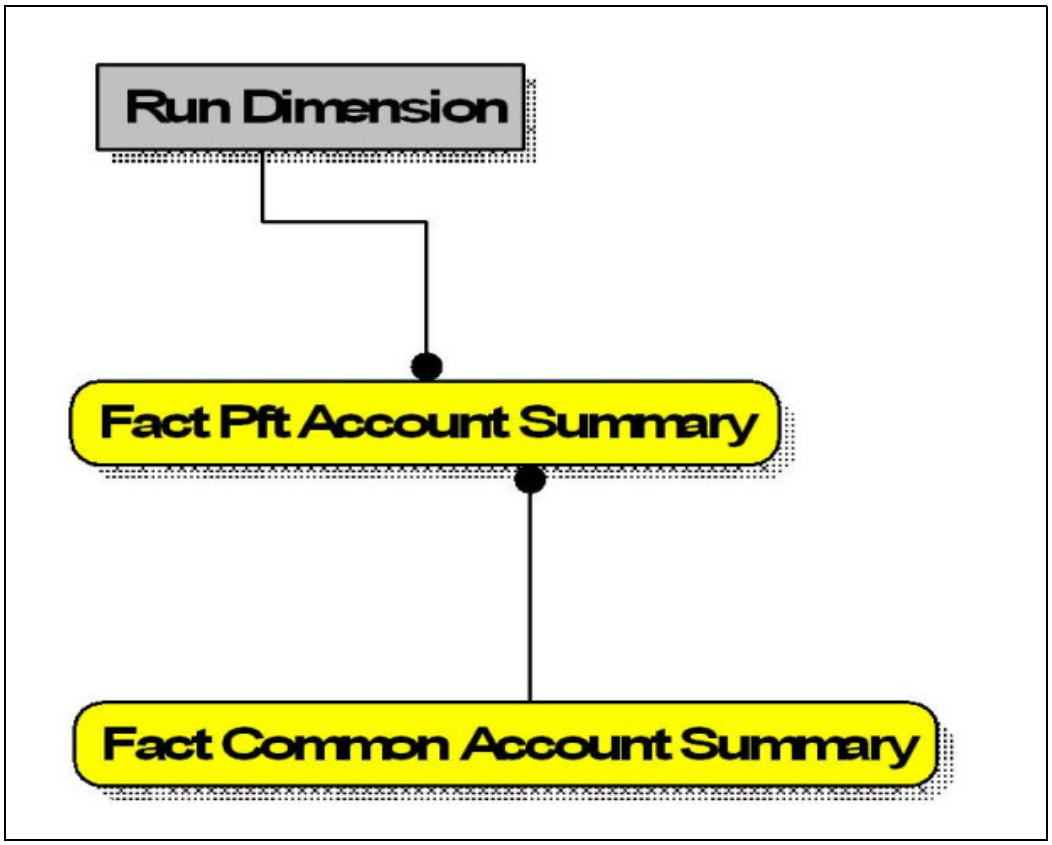


Figure 21. PFT Account Summary

- PFT Customer Summary

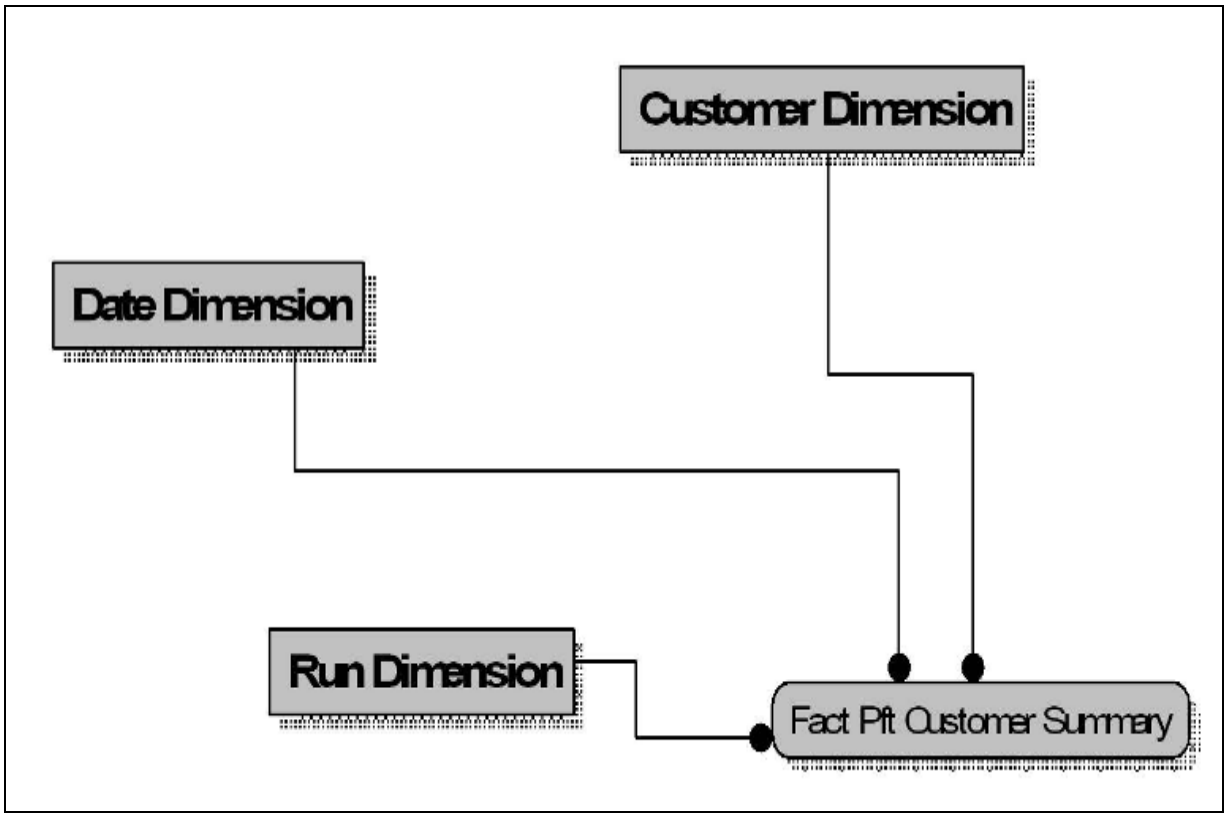


Figure 22. PFT Customer Summary

Data Flow: OFSIPA BI Data Model to Essbase Cubes

Reports of OFSIPA application can be configured to work on Relational database or Hyperion Essbase Multi-dimensional databases, that is cubes. 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 about the cube and for building the Essbase cubes. Essbase cubes can be built out of reporting fact entities to improve performance.

OFSIPA application has the following seeded cube metadata:

Table 4. Seeded Cube Metadata

Cube Code	Cube Name	Fact Entities in dataset
ADCRM001	Institutional Analysis	Fact Common Account Summary Fact CRM Account Summary Fact Common Customer Summary Fact CRM Customer Summary Fact FTP Account Summary Fact PFT Account Summary
Adiparm2	RM L and P	DIM_MANAGEMENT DIM_RUN DIM_LOB DIM_PRODUCT DIM_ORG_UNIT DIM_DATES DIM_REP_LINE WTHREPMV USRMGRMV

CHAPTER 3 Dimension Loading Process

This chapter discussed the following topics:

Dimension Tables Population

OFSIPA solution use the SCD component to handle dimensional data changes.

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, see

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

  http://www.oracle.com/technetwork/middleware/data-integrator/overview/odi-best practices-datawarehouse-whi-129686.pdf
- *Oracle® Warehouse Builder Data Modeling, ETL, and Data Quality Guide* at

  http://docs.oracle.com/cd/E14072_01/owb.112/e10935.pdf

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

An excellent published resource that covers SCD in detail is *"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 OFSPA solution are Type 1 and Type 2.

Prerequisites

1. The SCD executable should be present under <installation home>ficdb/bin. The file name is **scd.**

2. The user executing the SCD component should have execute rights on the file mentioned as prerequisite in point 2.
3. The setup tables accessed by SCD component are SYS_TBL_MASTER and SYS_STG_JOIN_MASTER. SYS_TBL_MASTER stores the information like which is the source stage table and the target dimension tables. The source sometimes can be the database views which could be simple or a complex view. SYS_STG_JOIN_MASTER stores the information like which source column is mapped to which column of a target dimension table. It makes use of data base sequence to populate into surrogate key columns of dimension tables.

Tables Used by the SCD Component

The database tables used by the SCD component are:

- SYS_TBL_MASTER

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

Table 5. SYS_TBL_MASTER 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.
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.
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	

Sample Data: This is the row put in by the solution installer for the Line of Business dimension.

MAP_REF_NUM	6
TBL_NM	DIM_LOB

STG_TBL_NM	STG_LOB_MASTER
SRC_PRTY	
SRC_PROC_SEQ	23
SRC_TYP	MASTER
DT_OFFSET	0
SRC_KEY	

Note: For any new dimension added, a row will have to be inserted to this table manually.

- SYS_STG_JOIN_MASTER

The solution installer will populate this table for the seeded dimensions..

Table 6. SYS_STG_JOIN_MASTER 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, 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` are:

- PK – Primary Dimension Value (may be multiple for a given "Mapping Reference Number")
- SK – Surrogate Key
- DA – Dimensional Attribute (may be multiple for a given "Mapping Reference Number")
- SD – Start Date
- ED – End Date
- LRI – Latest Record Indicator (Current Flag)

- CSK – Current Surrogate Key
- PSK – Previous Surrogate Key
- SS – Source Key
- LUD – Last Updated Date / Time
- LUB – Last Updated By

Sample Data: This is the row put in by the solution installer for the Line of Business dimension.

MAP_REF_NUM	6
COL_NM	V_LOB_CODE
COL_TYP	PK
STG_COL_NM	V_LOB_CODE
SCD_TYP_ID	
PRTY_LOOKUP_REQD_FLG	N
COL_DATATYPE	VARCHAR
COL_FORMAT	61

Note: *For any new dimension added, the column details will have to be* inserted to this table manually.

- DIM_<dimensionname>_V – The database view which SCD uses as the source.

Example

Dim_Bands_V

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

Note: *For any new dimension added, a view will have to be created* similar to DIM_BANDS_V.

- DIM_<dimensionname> – Output table to which SCD writes the dimension data.

A sequence should be added for every user-defined dimension.

Example

```
create sequence SEQ_DIM_<DIM> minvalue 1
maxvalue 999999999999999999999999999
increment by 1
```

Executing the SCD Component

To execute the SCD component from OFSAAI ICC framework create a batch according to the following steps:

Note: For a more comprehensive coverage of configuration and execution of a batch, see *Oracle Financial Services Analytical Applications* Infrastructure *User Guide*.

1. From the **Home** menu, select **Operations**, then select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container) and enter the Batch Name and Description.
3. Click **Save.**
4. Select the Batch you created in the earlier step by clicking the check box in the Batch Name container.

5. Click **New Task** ('+' symbol in Task Details container).
6. Enter the Task ID and Description.
7. Select **Run Executable**, from the Component ID list.
8. Click **Parameters.** Select the following from the Dynamic Parameters List and then click **Save**:
    - Datastore Type - Select the appropriate datastore from the list
    - Datastore Name - Select the appropriate name from the list
    - IP address - Select the IP address from the list
    - Executable - scd,<map ref num>

Example

scd, 61 (Refer the following sections for details)

- Wait: When the file is being executed you have the choice to either wait till the execution is complete or proceed with the next task. Click the list box of the field provided for Wait in the Value field to select 'Yes' or 'No'. Clicking **Yes** that you wish to wait for the execution to be complete. Clicking **No** indicates that you wish to proceed.
- Batch Parameter: Clicking **Yes** would mean that the batch parameters are also passed to the executable being started; else the batch parameters will not be passed to the executable.

Important: Always select **Y** in Batch Parameter.

For the Parameter Executable earlier mentioned, the map ref num values are

- -1 (if you want to process all the dimensions). The *Executable* parameter mentioned earlier would be 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.
- Execute the batch from Batch Execution by choosing the batch created following the steps mentioned in the preceding sections for a date.

Note: A seeded batch <Infodom>_SCD_Institutional_Perf_Dim is provided which has all the required dimensions as different tasks that are part of SCD.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen. You can access this from the Left Hand Side (LHS) menu as follows:

From the **Home** menu, select **Operations**, then select **Batch Monitor.**

Note: For a more comprehensive coverage, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

The status messages in Batch Monitor are :

N - Not Started

O - On Going

F - Failure

S – Success

The ICC execution log can be accessed on the application server in the following directory: `$FIC_DB_HOME/log/ficgen`.

The file name will have the batch execution id.

Sample

`/dbfiles/home/oracle/OFSAAI/ficdb/log/ficgen`

The detailed SCD component log can be accessed on the application server in the directory $FIC_HOME, go one folder up from there and then accessing the following path /ftpshare/<infodom name>/logs

The file name will have the batch execution id.

Sample

`/dbfiles/home/oracle/ftpshare/OFSAADEMO/logs`

Check the **.profile** file in the installation home if you are not able to find the paths mentioned earlier.

CHAPTER 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). Time dimension makes it possible to report the balances by Year, Quarter or Month using the rollup functionality of cubes. Cubes makes it possible to rollup the monthly balances to a quarter and then to a year level. For example, the monthly data for January, February and March gets rolled up to Quarter 1 and the Quarter 1, 2, 3 and 4 data get rolled up to, say Year 2011. 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).

Point in time reporting is supported for all the reports. The report is represented as of the data selected in the dashboard time prompts. By default, reports is always displayed for the latest available data.

This chapter discusses the following topics:

Overview of Time Dimension Population

Time dimension population transformation is used to populate the DIM_DATES table with values between two dates specified by the user as a batch parameter.

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 database function FN_DIM_DATES.

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 *Oracle Financial Services Institutional 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 (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, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*).
    - Iccserver
    - Router
    - AM Server

- Messageserver

4. Batches will have to be created for executing the function. For more details see, **Executing the Time dimension population transformation, page 4-2**.

Tables used by the Time Dimension Population Transformation

DIM_DATES - This table stores the date details to be used for building the cubes.

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

Executing the Time Dimension Population Transformation

To execute the function from OFSAAI Information Command Center (ICC) frame work, create a batch by performing the following steps:

Note: For a more comprehensive coverage of configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

1. From the **Home** menu, select **Operations**, then select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container) and enter the Batch Name and description.
3. Click **Save**.
4. Select the Batch you have created in the earlier step by clicking on the checkbox in the Batch Name container.
5. Click **New Task** ('+' symbol in Task Details container).
6. Enter the Task ID and Description.
7. Select **Transform Data**, from the components list.
8. Select the following from the Dynamic Parameters List and then click **Save**:
   - Datastore Type - Select appropriate datastore from the list
   - Datastore Name - Select appropriate name from the list
   - IP address - Select the IP address from the list
   - Rule Name - Select **Dim_Dates_Population** from the list of all available transformations. (This is a seeded Data Transformation which is installed as part of the OFSIPA solution installer. If you don't see this in the list, contact Oracle support)
   - Parameter List – Start Date, End Date (Refer the following for details on Parameter list)

Explanation for the parameter list is:

- Start Date – This is the date starting from which the Transformation will populate Dim_Dates table. Date should be specified in the format 'YYYYMMDD'.
- End Date - This is the date up to which the Transformation will populate Dim_Dates table. Date should be specified in the format 'YYYYMMDD'. Sample parameter for this task is '20081131', '20091231'.

9. You can execute the batch in two ways:

    - Execute the batch from Batch Execution by choosing the batch created following the steps mentioned in the preceding sections for a date.

Note: A seeded batch <INFODOM>_aCRM_CommonTasks - Task2 is provided so that the user can just modify the parameters and execute the batch.

- 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'

Checking the Execution Status

The status of execution can be monitored using the batch monitor screen.

Note: For a more comprehensive coverage of configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

The status messages in batch monitor are :

- N - Not Started
- O - On Going
- F - Failure
- S – Success

The Event Log window in Batch Monitor provides logs for execution with the top row being the most recent. If there is any error during execution, it will get listed here. Even if you see Successful as the status in Batch Monitor it is advisable to go through the Log and re-check if there are any errors. The execution log can be accessed on the application server by going to the following directory `$FIC_DB_HOME/log/date`. The file name will have the batch execution id. 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.

Check the **.profile** file in the installation home if you are not able to find the paths mentioned earlier.

CHAPTER 5 Customer Dimension Population

This chapter discusses the following topics:

Overview

In the current setup, Customer Dimension is populated as part of Party Model, where DIM_CUSTOMER derives its attributes from DIM_PARTY based on the Parties that have been assigned the role of a Customer.

Populating Party Dimension

DIM_PARTY table will be populated first from stage table STG_PARTY_MASTER using SCD. Function FN_PARTY_DENORMALIZE_DT will populate DIM_CUSTOMER from DIM_PARTY. The function is used to populate DIM_CUSTOMER table using a sequence.

The primary key for DIM_PARTY – N_PARTY_SKEY will be the surrogate key generated for the natural key - Party ID, an alphanumeric unique identifier within each staging instrument tables. This information is stored in DIM_CUSTOMER table as N_CUST_SKEY.

FSI_MERGE_SETUP_DETAILS

Customer dimension population makes use of setup table FSI_MERGE_SETUP_DETAILS. It would have seeded entries from the application installation. This table stores the mapping between source and target columns.

Table 7. Columns in FSI_MERGE_SETUP_DETAILS

Column Name	Data Type	Column Description
MERGE_CODE	VARCHAR2 (50 CHAR)	This is the role of the party, that is, customer, issuer, and so on.
SOURCE_TABLE	VARCHAR2 (30 CHAR)	This is the source table for Customer dimension population.
SOURCE_COLUMN	VARCHAR2 (30 CHAR)	This is the source column for Customer dimension population.
TARGET_COLUMN	VARCHAR2 (30 CHAR)	This is the target column for Customer dimension population.

Table 7. Columns in FSI_MERGE_SETUP_DETAILS

DEFAULT_VALUE	VARCHAR2 (4000 CHAR)	This is the default value for some target columns.
NVL_EXPRESSION	VARCHAR2 (30 CHAR)	This is the nvl expression applied on the source column for Customer dimension population.
AGGREGATE_FUNCTION	VARCHAR2 (30 CHAR)	This is used for aggregating data for some source columns.

Here is a sample data:

MERGE_CODE	MI		
TABLE SOURCE	DIM_PARTY		
SOURCE_COLUMN	V_PARTY_ID	V_MIDDLE_NAME	V_LAST_NAME
TARGET_COLUMN	V_ISSUER_CODE	V_D_CUST_MIDDLE_NAME	V_D_CUST_LAST_NAME
DEFAULT_VALUE			
NVL_EXPRESSION			
AGGREGATE_FUNCTION			

FSI_MERGE_SETUP_MASTER

Customer dimension population makes use of setup table FSI_MERGE_SETUP_MASTER as well. It would have seeded entries from the application installation. This table stores the mapping between source and target tables.

Table 8. Columns in FSI_MERGE_SETUP_MASTER

Column Name	Data Type	Column Description
MERGE_CODE	VARCHAR2 (50 CHAR)	This is the role of the party, that is, customer, issuer, and so on.
SOURCE_TABLES	VARCHAR2 (4000 CHAR)	This is the list of source tables for Customer dimension population.
TARGET_TABLE	VARCHAR2 (30 CHAR)	This is the target column for Customer dimension population.
ANSI_JOIN	VARCHAR2 (4000 CHAR)	This is the join condition that results in dataset.
FILTER_CONDITION	VARCHAR2 (4000 CHAR)	This is used for filtering thevalues in where clause.

Here is a sample data:

MERGE_CODE	MI
SOURCE TABLES	DIM_PARTY
TARGET_TABLE	V_PARTY_ID
ANSI JOIN	V_ISSUER_CODE
FILTER CONDITION	

Executing the Customer Dimension Population

To execute the customer dimension population, create a batch by performing the following steps:

1. **1.** From the **Home** menu, select **Operations**, then select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container) and enter the Batch Name and description.
3. Click **Save**.
4. Select the Batch you have created in the earlier step by clicking on the check box in the Batch Name container.
5. Click **New Task** ('+' symbol in Task Details container).
6. Enter the Task ID and Description.
7. Select **Transform Data** from the components list.
8. Select the following from the Dynamic Parameters List and then click **Save**:
   - Datastore Type - Select appropriate datastore from the list
   - Datastore Name - Select appropriate name from the list. Generally, it is the infodom name.
   - IP address - Select the IP address from the list
   - Rule Name - FN_PARTY_DENORMALIZE_DT
   - Parameter List: Surrogate Key Required Flag – Y or N

Batch run ID and As of Date are passed internally by the ICC to the Data Transformation task.

9. Execute the batch.

Execute the batch from Batch Execution by choosing the batch created following the steps mentioned in the preceding sections for a required date.

Note: A seeded batch<INFODOM> aCRM_CommonTasks – Task3 is provided so that the user can just modify the parameters and execute the batch.

Checking the Execution Status

The status of execution can be monitored from the *Batch Monitor* screen of OFSAAI Operations module.

Note: For a more comprehensive coverage of configuration & execution of a batch, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

The status messages in Batch Monitor are:

- N - Not Started
- O - On Going
- F - Failure
- S - Success

The Event Log window in Batch Monitor provides logs for execution with the top row being the most recent. If there is any error during execution, it will get listed here. Even if you see *Successful* as the status in Batch Monitor it is advisable to go through the Event Log and re-check if there are any errors. The execution log can be accessed on the application server by going to the directory*$FIC_DB_HOME/log/date*. The file name will have the Batch Execution ID.

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.

Check the **.profile** file in the installation home if you are not able to find the paths mentioned above.

CHAPTER 6

Account Dimension Population

This chapter discusses the following topics:

Dimension Tables Population

Data Foundation solutions use the SCD component to handle dimensional data changes.

Overview of SCD process

A Slowly Changing Dimension (SCD) is a dimension that stores and manages both current and historical data over time in a data warehouse. SCDs are dimensions that have data that changes slowly, rather than changing on a time-based, regular schedule. It is considered and implemented as one of the most critical ETL tasks in tracking the history of dimension records. There are three types of SCDs and you can use Warehouse Builder to define, deploy, and load all three types of SCDs.

- Type 1 SCDs - Overwriting

  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.

Table 9. Type 1 SCDs - Overwriting

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

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:

Table 10. Type 1 SCDs - Overwriting1

N_PRODUCT_ SKEY	V_PRODUCT_ NAME	D_START_DATE	D_END_DATE	F_LATEST_RECORD_IN DICATOR
1	Personal Loan	6/30/2010	12/31/9999	Y

- Type 2 SCDs - Creating another dimension record

  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:

Table 11. Type 2 SCDs - Creating another dimension record

N_PRODUCT_ SKEY	V_PRODUCT_ NAME	D_START_DATE	D_END_DATE	F_LATEST_RECORD_IN DICATOR
1	PL	6/30/2010	12/31/9999	N
2	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 that this is the latest record for the personal loan product. The same flag for the earlier record was set to 'N'.

- Type 3 SCDs - Creating a current value field

  A Type 3 SCD stores two versions of values for certain selected level attributes. Each record stores the previous value and the current value of the selected attribute.

  When the value of any of the selected attributes changes, the current value is stored as the old value and the new value becomes the current value.

For more information on SCDs, see

Oracle Data Integrator Best Practices for a Data Warehouse at

Oracle Data Integrator Best Practices for a Data Warehouse at
http://www.oracle.com/technetwork/middleware/data-integrator/overview/odi- best practices-datawarehouse-whi-129686.pdf

Oracle® Warehouse Builder Data Modeling, ETL, and Data Quality Guide at
http://docs.oracle.com/cd/E11882_01/owb.112/e10935.pdf
[http://docs.oracle.com/cd/E14072_01/owb.112/e10935.pdf]

Additional online sources include:

http://en.wikipedia.org/wiki/Slowly_changing_dimension

http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/10g/r2/owb/o wb10 gr2_gs/owb/lesson3/slowlychangingdimensions.htm

http://www.oraclebidwh.com/2008/11/slowly-changing-dimension-scd/

http://www.informationweek.com/news/software/bi/showArticle.jhtml?articleI D=2 04800027 and page no=1

http://www.informationweek.com/news/software/bi/showArticle.jhtml?articleI D=5 9301280

An excellent published resource that covers SCD in detail is *"The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling"* by Ralph Kimball and Margy Ross.

Prerequisites

Following are the prerequisites:

1. The SCD executable should be present under <installation home>ficdb/bin. The file name is scd.
2. The user executing the SCD component should have execute rights on the file mentioned as prerequisite in point 2.
3. The setup tables accessed by SCD component are SYS_TBL_MASTER and SYS_STG_JOIN_MASTER. SYS_TBL_MASTER stores the information like which is the source stage table andthe target dimension tables. The source sometimes can be the database views which could be simple or a complex view. SYS_STG_JOIN_MASTER stores the information like which source column is mapped to which column of a target dimension table. It makes use of data base sequence to populate into surrogate key columns of dimension tables.

Tables Used by the SCD Component

The database tables used by the SCD component are:

- SYS_TBL_MASTER

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

Table 12. SYS_TBL_MASTER dimensions

Table Name	Column Name	Expected Values
DIM_CUSTOMER_TYPE	V_CUST_CATEGORY	C
FCT_CRM_ACCOUNT_SUMMARY	V_SCENARIO_CODE	PLAN, BUDGET
FCT_OPPORTUNITY_ACTIVITY	V_ACTIVITY_STATUS	O, C
DIM_BANDS	V_BAND_TYPE	AGEONBOOK
		TURNOVER

Table 12. SYS_TBL_MASTER dimensions

FCT_ACCOUNT_PROFITABILITY	N_REP_LINE_CD	98000 - Net Income Before Taxes
		98500 - Tax Expense
		99000 - Net Income After Taxes
		107100 - Number of Customers
		107130 - Number of Open Customers
		107200 - Number of Accounts
		107230 - Number of Open Accounts
		107300 - Attrition Rate

Sample Data: This is the row put in by the solution installer for the Line of Business dimension.

Note: For any new dimension added, a row will have to be inserted to this table manually.

- SYS_STG_JOIN_MASTER

  The solution installer will populate this table for the seeded dimensions.

Table 13. SYS_STG_JOIN_MASTER dimensions

Table Name	Column Name	Expected Values
DIM_CUSTOMER_TYPE	V_CUST_CATEGORY	C
FCT_CRM_ACCOUNT_SUMMARY	V_SCENARIO_CODE	PLAN, BUDGET
FCT_OPPORTUNITY_ACTIVITY	V_ACTIVITY_STATUS	O, C
DIM_BANDS	V_BAND_TYPE	AGEONBOOK
		TURNOVER
FCT_ACCOUNT_PROFITABILITY	N_REP_LINE_CD	98000 - Net Income Before Taxes
		98500 - Tax Expense
		99000 - Net Income After Taxes
		107100 - Number of Customers
		107130 - Number of Open Customers
		107200 - Number of Accounts
		107230 - Number of Open Accounts
		107300 - Attrition Rate

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

- PK – Primary Dimension Value (may be multiple for a given "Mapping ReferenceNumber")
- SK – Surrogate Key
- DA – Dimensional Attribute (may be multiple for a given "Mapping Reference Number")
- SD – Start Date
- ED – End Date

- LRI – Latest Record Indicator (Current Flag)
- CSK – Current Surrogate Key
- CSK – Current Surrogate Key
- SS – Source Key
- LUD – Last Updated Date/Time
- LUB – Last Updated By

Sample Data: This is the row put in by the solution installer for the Line of Business dimension.

Table Name	Column Name	Expected Values
DIM_CUSTOMER_TYPE	V_CUST_CATEGORY	C
FCT_CRM_ACCOUNT_SUMMARY	V_SCENARIO_CODE	PLAN, BUDGET
FCT_OPPORTUNITY_ACTIVITY	V_ACTIVITY_STATUS	O, C
DIM_BANDS	V_BAND_TYPE	AGEONBOOK
		TURNOVER
FCT_ACCOUNT_PROFITABILITY	N_REP_LINE_CD	98000 - Net Income Before Taxes
		98500 - Tax Expense
		99000 - Net Income After Taxes
		107100 - Number of Customers
		107130 - Number of Open Customers
		107200 - Number of Accounts
		107230 - Number of Open Accounts
		107300 - Attrition Rate

Note: For any new dimension added, the column details will have to be inserted to this table manually.

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

Example

Example

Dim_Bands_V

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

Note: For any new dimension added, a view will have to be created similar to DIM_BANDS_V.

- DIM_<dimensionname> - Output table to which SCD writes the dimension data.

A sequence should be added for every user-defined dimension.

```
create sequence SEQ_DIM_<DIM> minvalue 1
maxvalue 999999999999999999999999999
increment by 1
```

Executing the SCD Component

To execute the SCD component from Operations module of OFSAAI, create a batch according to the following steps:

Note: For a more comprehensive coverage of configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

1. From the **Home** menu, select **Operations**, then select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container) and enter the Batch Name and Description.
3. Click **Save**.
4. Select the Batch you created in the earlier step by clicking the check box in the Batch Name container.
5. Click **New Task** ('+' symbol in Task Details container).
6. Enter the Task ID and Description.
7. Select **Run Executable** from the Component ID list.
8. Click **Parameters**. Select the following from the Dynamic Parameters List and then click **Save**:

- Datastore Type - Select the appropriate datastore from the list
- Datastore Name - Select the appropriate name from the list
- IP address - Select the IP address from the list
- Executable - scd, <map ref num>

Example

Example

scd, 61 (Refer the following sections for details)

- Wait: When the file is being executed you have the choice to either wait till the execution is complete or proceed with the next task. Click the list box of the field provided for **Wait in the Value** field to select **Yes** or **No**. Clicking **Yes** confirms that you wish to wait for the execution to be complete. Clicking **No** indicates that you wish to proceed.
- Batch Parameter: Clicking **Yes** would mean that the batch parameters are also passed to the executable being started else the batch parameters will not be passed to the executable.

Note: Always select **Y** in Batch Parameter.

For the Parameter Executable earlier mentioned, the map ref num values are

- -1 (if you want to process all the dimensions). The Executable parameter mentioned earlier would be 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.

9. Execute the batch from Batch Execution by choosing the batch created following the steps mentioned in the preceding sections for a date.

Note: A seeded batch <Infodom>_SCD is provided which has all the required dimensions as different tasks that are part of SCD.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen. You can access this from the Left Hand Side (LHS) menu as follows:

From the **Home** menu, select **Operations**, then select **Batch Monitor**.

Note: For a more comprehensive coverage, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

The status messages in Batch Monitor are:

- N - Not Started
- O - On Going
- F - Failure
- S – Success

The ICC execution log can be accessed on the application server in the following directory: `$FIC_DB_HOME/log/ficgen`.

The file name will have the batch execution id.

Sample

/dbfiles/home/oracle/OFSAAI/ficdb/log/ficgen

The detailed SCD component log can be accessed on the application server in the directory $FIC_HOME, go one folder up from there and then accessing the following path: /ftpshare/<infodom name>/logs

The file name will have the batch execution id.

Sample

/dbfiles/home/oracle/ftpshare/OFSAADEMO/logs

Check the **.profile** file in the installation home if you are not able to find the paths mentioned earlier.

Load DIM_ACCOUNT through SCD

The SCD population in DIM_ACCOUNT table generates individual numeric SKEYs for every account number with an additional leg skey. Following are the columns that will be loaded during SCD population:

- V_ACCOUNT_NUMBER
- N_ACCT_SKEY
- N_RCV_LEG_ACCT_SKEY
- FIC_MIS_DATE

This approach replaces the function load in which the table DIM_ACCOUNT is getting loaded through the function, `FN_POPDIMACCOUNT`. This loads the followingcolumns into DIM_ACCOUNT table:

- V_ACCOUNT_NUMBER
- N_ACCT_SKEY
- N_RCV_LEG_ACCT_SKEY
- FIC_MIS_DATE

Where, the sources are the different product processor tables present in the solution, which are configured in FSI_DIM_ACCOUNT_SETUP_DETAILS table.

DIM_ACCOUNT SCD

Batch <INFODOM>DIM_ACCOUNT_SCD has been introduced with 33 tasks under it. These 33 tasks represent the 33 SCD processes where different product processors would be the source and DIM_ACCOUNT would be the target. MAP_REF_NUMs 188 to 217 have been introduced into SYS_TBL_MASTER table, and subsequently into SYS_STG_JOIN_MASTER.

DIM_ACCOUNT_SCD has been introduced with 33 tasks under it. These 33 tasks represent the 33 SCD processes where different product processors would be the source and DIM_ACCOUNT would be the target. MAP_REF_NUMs 188 to 217 have been introduced into SYS_TBL_MASTER table, and subsequently into SYS_STG_JOIN_MASTER.

Depending on the requirement by an application, a task can be excluded or included from the batch execution.

LOAD DIM TABLES THROUGH SCD

Batch <INFODOM>_SCD has been introduced with 129 tasks under it. These 129 tasks represent the 129 SCD processes where different staging tables would be the source and Dimension Tables would be the targets. The required SCDs have been introduced into SYS_TBL_MASTER table, and subsequently into SYS_STG_JOIN_MASTER.

Depending on the requirement by an application, a task can be excluded or included from the batch execution.

Improve SCD Performance

SCD performance can be improved by providing hints and session alter statements. This requires the presence of the following four columns in SYS_TBL_MASTER:

- merge_hint
- select_hint
- session_enable_statement
- session_disable_statement

These columns are present in the OFSAAI versions 7.3.2.4.0 and higher. If these have to be used in OFSAAI versions 7.3.2.2.0 or 7.3.2.3.0 and higher, execute the following SQL queries:

```
ALTER TABLE SYS_TBL_MASTER ADD MERGE_HINT VARCHAR2(255)
/
ALTER TABLE SYS_TBL_MASTER ADD SELECT_HINT VARCHAR2(255)
/
ALTER TABLE SYS_TBL_MASTER ADD SESSION_ENABLE_STATEMENT VARCHAR2(255)
/
ALTER TABLE SYS_TBL_MASTER ADD SESSION_DISABLE_STATEMENT VARCHAR2(255)
/
```

During upgrade to OFSAAI 7.3.2.4.0, ensure to backup SYS_TBL_MASTER table and to drop the preceding four columns, if these scripts are executed in any of the OFSAAI versions prior to 7.3.2.4.0. Otherwise, an upgrade to OFSAAI 7.3.2.4.0 may throw an error, since the columns are existing.

- For improving performance, hints for the MERGE query which is generated internally by the SCD can be provided under MERGE_HINT. Session alters could be mentioned under SESSION_ENABLE_STATEMENT and SESSION_DISABLE_STATEMENTcolumns.
- SESSION_ENABLE_STATEMENTs will be executed before the MERGE in the SCD and SESSION_DISABLE_STATEMENTs will be executed after the SCD MERGE.
- Since all the tasks under the SCD batch for DIM_ACCOUNT works on the same target, the SESSION_DISABLE_STATEMENTs in SYS_TBL_MASTER cannot be provided when tasks are executed. In this case, there can be a separate SQL file to contain all the SESSION_DISABLE_STATEMENTs to be executed once after all the tasks in the SCD are done. The SESSION_DISABLE_STATEMENT will hold a null in SYS_TBL_MASTER table.
- SESSION_ENABLE_STATEMENTs are required to be mentioned only for the first task in the batch. Here the target is the same for all the tasks under a batch. In case any of the tasks are to be executed separately, then the SESSION_ENABLE_STATEMENTs should be mentioned for any one of the tasks which is included in the batch for the execution.

Table 14. MERGE_HINT and SESSION_ENABLE_STATEMENT in SYS_TBL_MASTER

Table Name	Stage Table Name	Stage Table Name	Session Enable Statement
DIM_ACCOUNT	STG_LOAN_CONTR ACTS_V	/*+ parallel (DIM_ACCOUNT,10) */	"alter session enable parallel dml query", "alter table DIM_ACCOUNT nologging parallel 10"

- All the tasks can be executed in parallel. This might cause the N_RCV_LEG_ACCT_SKEY to have an incremental value as compared to N_ACCT_SKEY.
- Execute the SQL file with all the SESSION_DISABLE_STATEMENTs, after the successful completion of the SCD batch.
  - Once the DIM_ACCOUNT table is populated using this approach, you will not be able to use the initial approach (FN_POPDIMACCOUNT) as this will lead to skey conflict.
  - Ensure that you have set the value of the sequence seq_dim_account_scd as max (value of skey in DIM_ACCOUNT) + 1, before moving from old to new approach.
  - The F_LATEST_RECORD_INDICATOR for an existing DIM_ACCOUNT data already loaded by the function should be updated to 'Y' before running the SCD, failing which a new skey might get generated for the same account number.
  - SCD execution occurs based on the GAAP code which is configured in SETUP_MASTER table. This has been introduced to tackle the scenario of multiple GAAP codes. Whether or not there exist multiple GAAP codes, SETUP_MASTER should be manually configured as follows:

Table 15. SETUP_MASTER configuration

V_COMPONENT_CO DE	V_COMPONENT_DE SC	V_COMPONENT_VA LUE
DEFAULT_GAAP	DEFAULT_GAAP	USGAAP

Where V_COMPONENT_VALUE should be manually populated with the required GAAP code.

Handling Multiple GAAP Codes for the Same Account Number for the Same MIS Date in SCD

When multiple GAAP codes exist for the same account number for the same MIS date, configure the `SETUP_MASTER` table manually as mentioned in the preceding section:

`V_COMPONENT_VALUE` will hold the GAAP code for which the SCD is to be executed.

If there are different GAAP codes for two distinct account numbers for the same MIS date, then the SCD has to be executed for each GAAP code by changing the `V_COMPONENT_VALUE` manually in setup_master table. The `SETUP_MASTER` table should have only one record `WHERE V_COMPONENT_DESC = 'DEFAULT_GAAP'`.

CHAPTER 7 Exchange Rate History Population

This chapter discusses the following topics:

Introduction

Exchange Rate History entity stores the exchange rates between the currencies for an effective date from one or multiple sources.

Exchange Rate History population should be executed before any fact table is populated to ensure exchange rates between currencies are available prior. Exchange Rate History entity is loaded by means of Table to Table Transformation process.

Following is the seeded Table-to-Table definition that loads data into Exchange Rate History:

Table 16. T2T Definition Exchange Rate History

T2T Definition Name	Source Table(s)	Destination Table
T2T_EXCHANGE_RATE_HIST	STG_EXCHANGE_RATE_HIST	FSI_EXCHANGE_RATE_HIST

Execution of Currency Exchange Rates Population T2T

The batch <INFODOM>_POP_EXCHANGE_RATES needs to be executed to populate fsi_exchange_rates as the entries in setup_master are seeded during installation.

Note: FSI_EXCHANGE_RATES table has to be loaded prior loading any of the other Account Summary tables.

- Metadata Browser
- Common Account Summary

Currency Execution Rates - Batch Execution

A seeded batch, <Infodom>_POP_EXCHANGE_RATES has to be executed for the required MIS Date.

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the Batch Name and Description.
3. Click **Save**.

4. Click the check box in the Batch Name container to select the Batch, you created in the earlier step.
5. Enter the Task ID and Description.
6. Select Load Data from the Components list.
7. Select the following from the Dynamic Parameters List:
   - Datastore Type - Select the appropriate datastore from the list.
   - Datastore Name - Select the appropriate name from the list.
   - IP address - Select the IP address from the list.
   - Load Mode - Select Table to Table from the list.
   - Source Name - Select the <T2T Source Name> from the list.
   - File Name - Select the T2T name for the source stage channel table you want to process.
8. Click **Save**.

Data file name will be blank for any Table to Table Load mode. Default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

9. Execute the batch created in the preceding steps.

Exchange Rate History Population

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<INFODOM>_aCRM_CommonTasks - Task4** has to be executed for the required date.

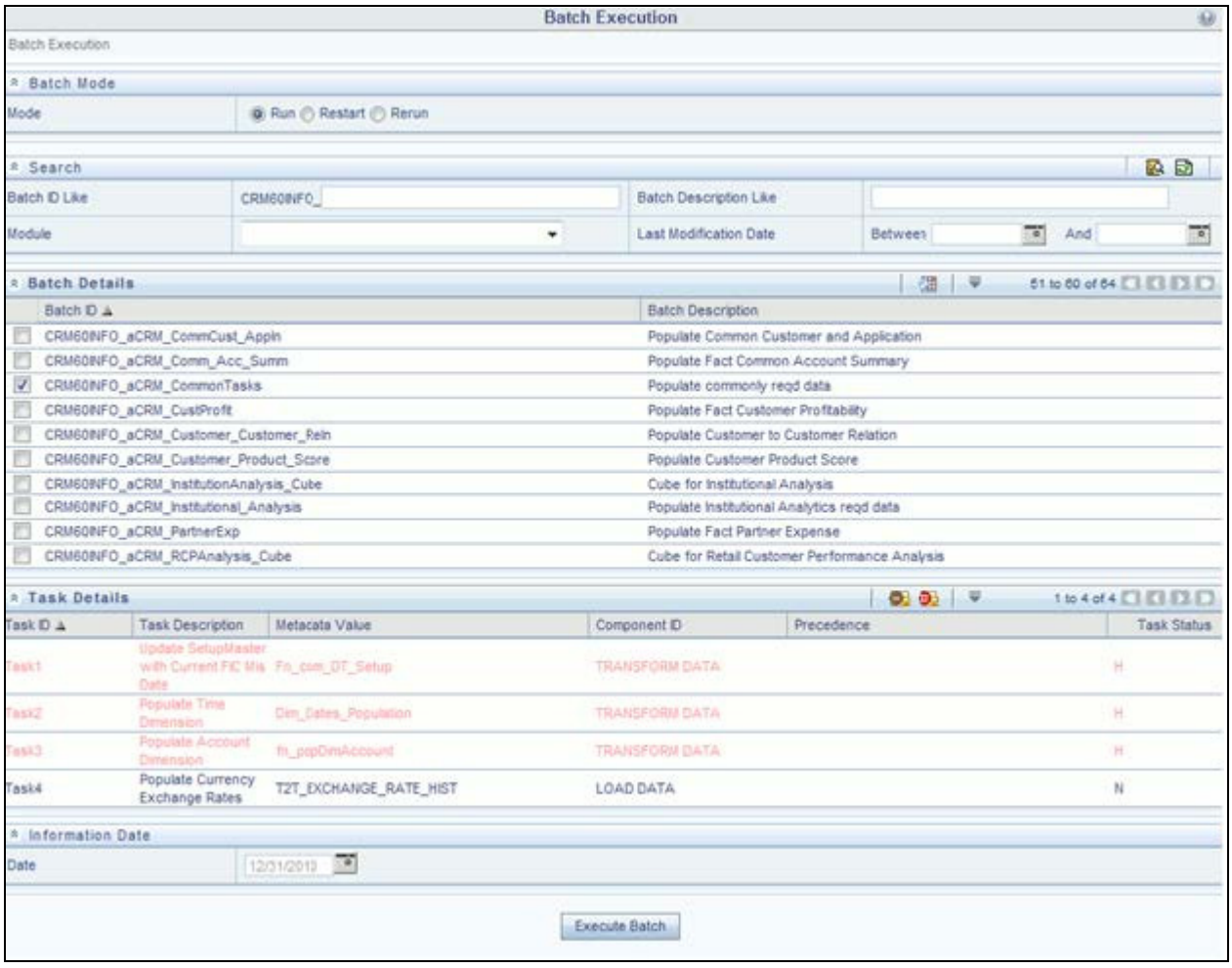


Figure 23. <INFODOM>_aCRM_CommonTasks - Task4

Alternatively, following steps will help to create a new batch task for Loading Historical Exchange Rates:

1. From the **Home** menu, select **Operations**, then select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the Batch Name and Description.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the Batch, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the components list.
7. Select the following from the Dynamic Parameters List and then click **Save**.
   - **Datastore Type** - Select appropriate datastore from the list.
   - **Datastore Name** - Select appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select Table to Table from the list.

- **Source Name** - Select <**T2T Source Name**> from the list.
- **File Name** - Select the table to table transformation **T2T_EXCHANGE_RATE_HIST**.

Data file name will be blank for any Table to Table Load mode.

8. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

10. Check T2T component logs and batch messages to check the status of load.

T2T component can fail because of following cases:

- Unique constraint error – Target table may already contain the primary keys that are part of the staging tables.
- NOT NULL constraint error – do not have values for NOT NULL columns in the target table.

Checking the Execution Status

The status of execution can be monitored using the batch monitor screen.

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 `$FIC_DB_HOME/log/t2t` directory: The file name will have the batch execution id.

Validating the Exchange Rate

The Function Fn_ratevalidation is executed using the task. Edit the "Task1" of the batch "<INFODOM>_FN_RATEVALIDATION" and pass the below parameters to the task:

- Starting date
- End date

All the exchange rates present in FSI_EXCHANGE_RATE_HIST table whose 'effective date' lies in the range of these values will be validated on execution of this batch. The validated rates will be available in the table FSI_EXCHNG_RATE_DIRECT_ACCESS.

Rate Triangulation is also achieved during this process

CHAPTER 8

Account Summary Population

Account Summary tables are loaded from the staging product processor tables using the Table to Table (T2T) component of Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) framework.

This chapter covers the following topics:

Overview of Account Summary Tables

Customer account level data from the Oracle Financial Services Analytical Applications (OFSAA) staging product processor tables must be consolidated into a standardized relational Business Intelligence (BI) data model. This consolidation is done to have all the staging product processor table data in a single Fact table.

The Account Summary table data can be used for building cubes which allow rollup of data for a dimension or a combination of dimensions.

This relational BI model consists of three vertically partitioned Account Summary tables that are organized by application subject area.

- FCT_COMMON_ACCOUNT_SUMMARY – This table is shared by all OFSAA BI applications which contain dimensional values, attributes, and financial measures which are generally applicable to the individual account records. This data is sourced directly from the staging area.
- FCT_CRM_ACCOUNT_SUMMARY – This table has the measures used by all the Customer Insight applications.

Yet, there are few other Account Summary tables which have been designed to store Enterprise Performance Management (EPM) data:

- FCT_PFT_ACCOUNT_SUMMARY – This table has Profitability Management (PFT) specific measures.
- FCT_FTP_ACCOUNT_SUMMARY – This table has Funds Transfer Pricing (FTP) specific measures.
- FCT_REG_CAP_ACCOUNT_SUMMARY – This table has Regulatory Capital specific measures.
- FCT_ECO_CAPITAL_ACCOUNT_SUMMARY – This table has Economic Capital specific measures.

Data Flow

The following diagram depicts the flow of data into account summary tables:

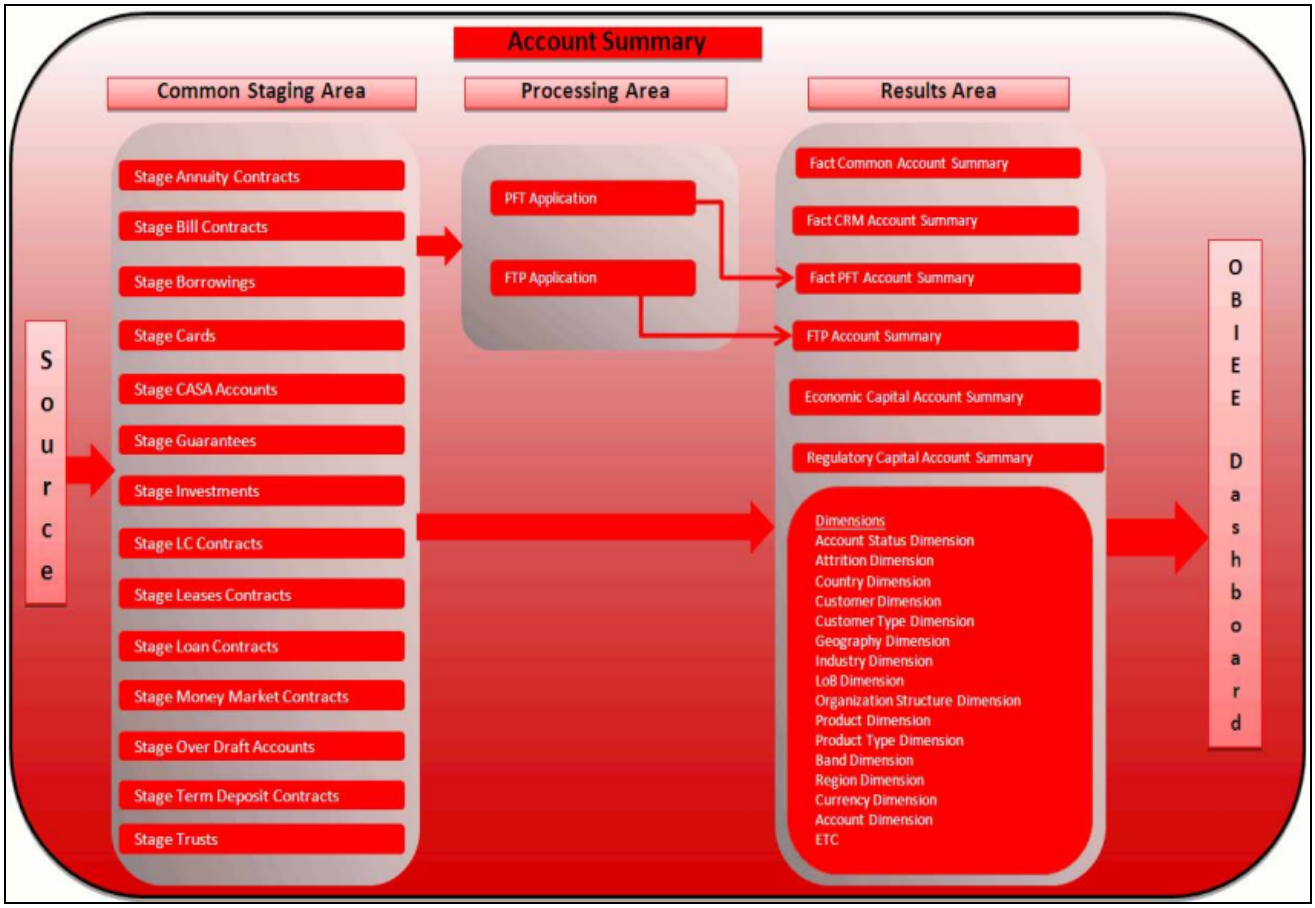


Figure 24. Account summary tables

Overview of Account Summary Population

Table to Table seeded definitions are provided for loading data into Common Account Summary and CRM Account summary tables.

Following are the lists for the same:

- Common Account Summary

Table 17. Common Account Summary definitions

SLNo	Source Table	T2T Definition Name	Destination Table
1	STG_ANNUITY_CONTRACTS	T2T_STG_ANNUITY_CONTRACTS_-CAS	FCT_COMMON_ACCOUNT_-SUMMARY
2	STG_BILLS_CONTRACTS	T2T_STG_BILLS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
3	STG_BORROWINGS	T2T_STG_BORROWINGS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
4	STG_CARDS	T2T_STG_CARDS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
5	STG_CASA	T2T_STG_CASA_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
6	STG_GUARANTEES	T2T_STG_GUARANTEES_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
7	STG_INVESTMENTS	T2T_STG_INVESTMENTS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
8	STG_LC_CONTRACTS	T2T_STG_LC_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
9	STG_LEASES_CONTRACTS	T2T_STG_LEASES_CONTRACTS_-CAS	FCT_COMMON_ACCOUNT_-SUMMARY
10	STG_LOAN_CONTRACTS	T2T_STG_LOANS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
11	STG_MM_CONTRACTS	T2T_STG_MM_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
12	STG_OD_ACCOUNTS	T2T_STG_OD_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
13	STG_TD_CONTRACTS	T2T_STG_TD_CONTRACTS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
14	STG_TRUSTS	T2T_STG_TRUSTS_CAS	FCT_COMMON_ACCOUNT_-SUMMARY
15	STG_COMMITMENT_CON-TRACTS	T2T_STG_COMMITMENT_CON-TRACTS_CAS	FCT_COMMON_ACCOU NT_SUMMARY
16	STG_MUTUAL_FUNDS	T2T_STG_MUTUAL_FUNDS_CAS	FCT_COMMON_ACCOU NT_SUMMARY

- CRM Account Summary

Sl No.	Source Table	T2T Definition Name	Destination Table
1	STG_ANNUITY_CONTRACTS	T2T_STG_CRMAS_ANNUITY_CON-TRACTS	FCT_CRM_ACCOUNT_SUM-MARY
2	STG_BILLS_CONTRACTS	T2T_STG_CRMAS_BILLS_CON-TRACTS	FCT_CRM_ACCOUNT_SUM-MARY
3	STG_BORROWINGS	T2T_STG_CRMAS_BORROWINGS	FCT_CRM_ACCOUNT_SUM-MARY
4	STG_CARDS	T2T_STG_CRMAS_CARDS	FCT_CRM_ACCOUNT_SUM-MARY
5	STG_CASA	T2T_STG_CRMAS_CASA	FCT_CRM_ACCOUNT_SUM-MARY
6	STG_GUARANTEES	T2T_STG_CRMAS_GUARANTEES	FCT_CRM_ACCOUNT_SUM-MARY
7	STG_INVESTMENTS	T2T_STG_CRMAS_INVESTMENTS	FCT_CRM_ACCOUNT_SUM-MARY
8	STG_LC_CONTRACTS	T2T_STG_CRMAS_LC_CONTRACTS	FCT_CRM_ACCOUNT_SUM-MARY
9	STG_LEASES_CONTRACTS	T2T_STG_CRMAS_LEASES_CON-TRACTS	FCT_CRM_ACCOUNT_SUM-MARY
10	STG_LOAN_CONTRACTS	T2T_STG_CRMAS_LOAN_CON-TRACTS	FCT_CRM_ACCOUNT_SUM-MARY
11	STG_MM_CONTRACTS	T2T_STG_CRMAS_MM_CON-TRACTS	FCT_CRM_ACCOUNT_SUM-MARY
12	STG_OD_ACCOUNTS	T2T_STG_CRMAS_OD_ACCOUNTS	FCT_CRM_ACCOUNT_SUM-MARY
13	STG_TD_CONTRACTS	T2T_STG_CRMAS_TD_CONTRACTS	FCT_CRM_ACCOUNT_SUM-MARY
14	STG_TRUSTS	T2T_STG_CRMAS_TRUSTS	FCT_CRM_ACCOUNT_SUM-MARY
15	STG_COMMITMENT_CON-TRACTS	T2T_STG_CRMAS_COMMITMENTS	FCT_CRM_ACCOUNT_SUM-MARY
16	STG_MUTUAL_FUNDS	T2T_STG_CRMAS_MUTUAL_FUNDS	FCT_COMMON_ACCOUNT_-SUMMARY

- FTP Account Summary

Table 18. FTP Account Summary definitions

Sl No	Source Table	T2T Definition Name	Destination Table
1	FSI_D_ANNUITY_CONTRACTS	T2T_FCT_FTP_ACCOUNT_ANNUITY	FCT_FTP_ACCOUNT_SUMMARY
2	FSI_D_BORROWINGS	T2T_FCT_FTP_ACCOUNT_BORROWINGS	FCT_FTP_ACCOUNT_SUMMARY
3	FSI_D_CASA	T2T_FCT_FTP_ACCOUNT_CASA	FCT_FTP_ACCOUNT_SUMMARY
4	FSI_D_CREDIT_LINES	T2T_FCT_FTP_ACCOUNT_CREDIT_LINES	FCT_FTP_ACCOUNT_SUMMARY
5	FSI_D_CREDIT_CARDS	T2T_FCT_FTP_ACCOUNT_CREDITCARDS	FCT_FTP_ACCOUNT_SUMMARY
6	FSI_D_GUARANTEES	T2T_FCT_FTP_ACCOUNT_GUARANTEES	FCT_FTP_ACCOUNT_SUMMARY
7	FSI_D_INVESTMENTS	T2T_FCT_FTP_ACCOUNT_INVESTMENTS	FCT_FTP_ACCOUNT_SUMMARY
8	FSI_D_LEASES	T2T_FCT_FTP_ACCOUNT_LEASES	FCT_FTP_ACCOUNT_SUMMARY
9	FSI_D_LOAN_CONTRACTS	T2T_FCT_FTP_ACCOUNT_LOANS	FCT_FTP_ACCOUNT_SUMMARY
10	FSI_D_MM_CONTRACTS	T2T_FCT_FTP_ACCOUNT_MM_-CONTRACTS	FCT_FTP_ACCOUNT_SUMMARY
11	FSI_D_MORTGAGES	T2T_FCT_FTP_ACCOUNT_MORTGAGES	FCT_FTP_ACCOUNT_SUMMARY
12	FSI_D_TERM_DEPOSITS	T2T_FCT_FTP_ACCOUNT_TDEPOSITS	FCT_FTP_ACCOUNT_SUMMARY
13	FSI_D_TRUSTS	T2T_FCT_FTP_ACCOUNT_TRUSTS	FCT_FTP_ACCOUNT_SUMMARY
14	FSI_D_MUTUAL_FUNDS	T2T_FCT_FTP_ACCOUNT_MUTUAL_FUND	FCT_FTP_ACCOUNT_SUMMARY

- PFT Account Summary

Table 19. PFT Account Summary definitions

Sl No	Source Table	T2T Definition Name	Destination Table
1	FSI_D_ANNUITY_CONTRACTS	T2T_FCT_PFT_ACCOUNT_ANNUITY	FCT_PFT_ACCOUNT_SUMMARY
2	FSI_D_BORROWINGS	T2T_FCT_PFT_ACCOUNT_BORROWINGS	FCT_PFT_ACCOUNT_SUMMARY

Table 19. PFT Account Summary definitions

3	FSI_D_CASA	T2T_FCT_PFT_ACCOUNT_CASA	FCT_PFT_ACCOUNT_SUMMARY
4	FSI_D_CREDIT_LINES	T2T_FCT_PFT_ACCOUNT_CREDIT_LINES	FCT_PFT_ACCOUNT_SUMMARY
5	FSI_D_CREDIT_CARDS	T2T_FCT_PFT_ACCOUNT_CREDITCARDS	FCT_PFT_ACCOUNT_SUMMARY
6	FSI_D_GUARANTEES	T2T_FCT_PFT_ACCOUNT_GUARANTEES	FCT_PFT_ACCOUNT_SUMMARY
7	FSI_D_INVESTMENTS	T2T_FCT_PFT_ACCOUNT_INVESTMENTS	FCT_PFT_ACCOUNT_SUMMARY
8	FSI_D_LEASES	T2T_FCT_PFT_ACCOUNT_LEASES	FCT_PFT_ACCOUNT_SUMMARY
9	FSI_D_LOAN_CONTRACTS	T2T_FCT_PFT_ACCOUNT_LOANS	FCT_PFT_ACCOUNT_SUMMARY
10	FSI_D_MORTGAGES	T2T_FCT_PFT_ACCOUNT_MORTGAGES	FCT_PFT_ACCOUNT_SUMMARY
11	FSI_D_TERM_DEPOSITS	T2T_FCT_PFT_ACCOUNT_DEPOSITS	FCT_PFT_ACCOUNT_SUMMARY
12	FSI_D_TRUSTS	T2T_FCT_PFT_ACCOUNT_TRUSTS	FCT_PFT_ACCOUNT_SUMMARY
13	FSI_D_MUTUAL_FUNDS	T2T_FCT_PFT_ACCOUNT_MUTUAL_FUND	FCT_PFT_ACCOUNT_SUMMARY

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 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 (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, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.)
   - Iccserver
   - Router
   - AM Server
   - Messageserver
4. Batches will have to be created for executing. This is explained in Executing the Account Summary Population T2T section.

5. Dimension Population should have been done before you execute the T2T batch. (For more information, refer to Chapter 3, "Dimension Loading Process," and Chapter 4, "Time Dimension Population,".)

Fact Common Account Summary

Following are the lists of tables used in the population of Fact Common Account Summary & Fact CRM Account Summary tables.

Following mentioned Dimension tables are required to be loaded prior to executing the T2T:

- DIM_DATES
- DIM_ACCOUNT
- DIM_CUSTOMER
- DIM_PRODUCT
- DIM_CHANNEL
- DIM_BANDS
- DIM_ORG_STRUCTURE and so on.

Fact CRM Account Summary

Fact Common Account Summary entity needs to be populated before executing the Fact CRM Account Summary T2Ts.

Following are the list of tables used in the population of Fact CRM Account Summary and these tables are required to be loaded prior to running the T2T:

- DIM_DATES
- DIM_ACCOUNT
- FCT_COMMON_ACCOUNT_SUMMARY
- DIM_ACCT_STATUS
- DIM_BANDS
- DIM_CHANNEL
- DIM_CUSTOMER
- DIM_ORG_STRUCTURE
- DIM_LOB
- DIM_OFFER
- DIM_OPPORTUNITY
- DIM_PRODUCT
- DIM_PROSPECT
- DIM_RETENTION_OFFER_TYPE

- DIM_SALES_REPRESENTATIVE
- DIM_TREATMENT
- DIM_VINTAGE

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to the section Dimension Tables Population.

For details on populating DIM_DATES dimension table, refer to section Overview of Time Dimension Population.

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification*.

For more information on the dimensions, refer to *ERwin Datamodel*.

Executing the Account Summary Population T2T

Fact Common Account Summary table has to be loaded prior loading any of the other Account Summary tables.

You can execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

Fact Common Account Summary

A seeded batch, **<Infodom>_aCRM_Comm_Acc_Summ** has to be executed for the required MIS Date.

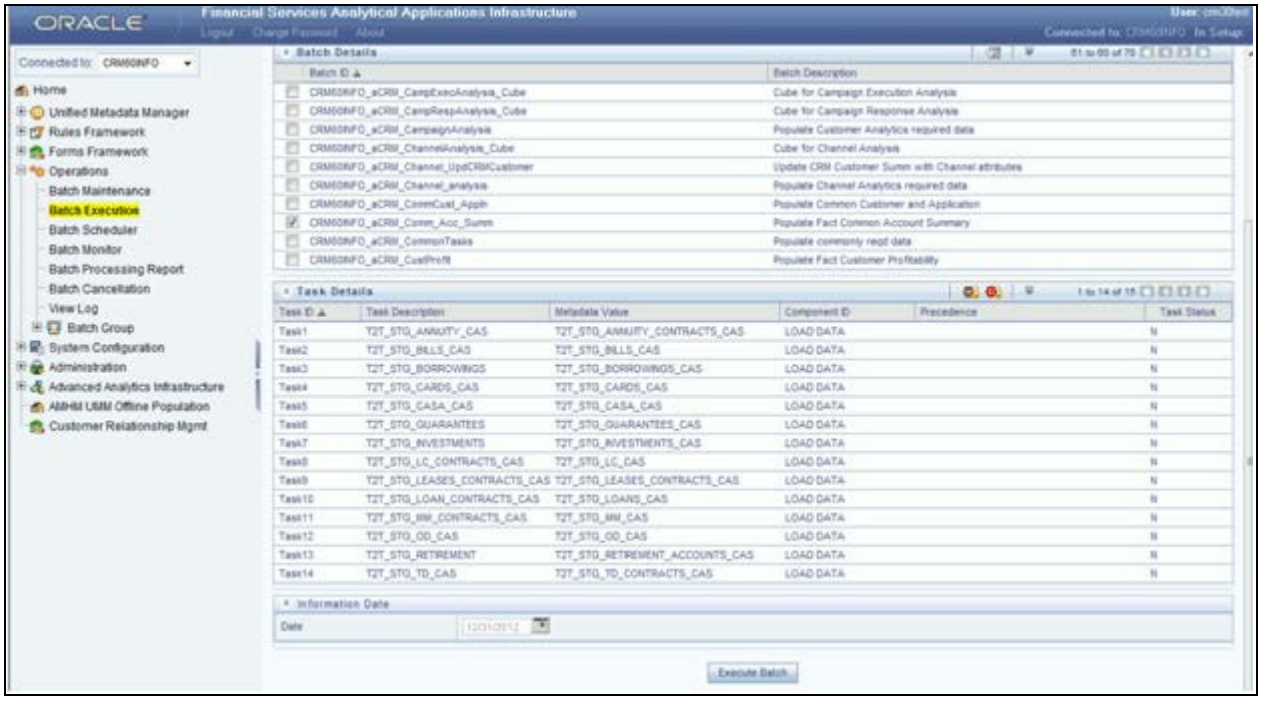


Figure 25. <Infodom>_aCRM_Comm_Acc_Summ

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.

2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save.**
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select Table to Table from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name for the source stage channel table you want to process.
8. Data file name will be blank for any Table to Table Load mode. Default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, Default value has to be provided. For example, default value is [DRCY]='USD' Here 'USD' acts as reporting currency parameter to T2T.
9. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
10. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact FTP Account Summary

A seeded batch, **<INFODOM>_FTP_Account_Summary** has to be executed for the required MIS Date.

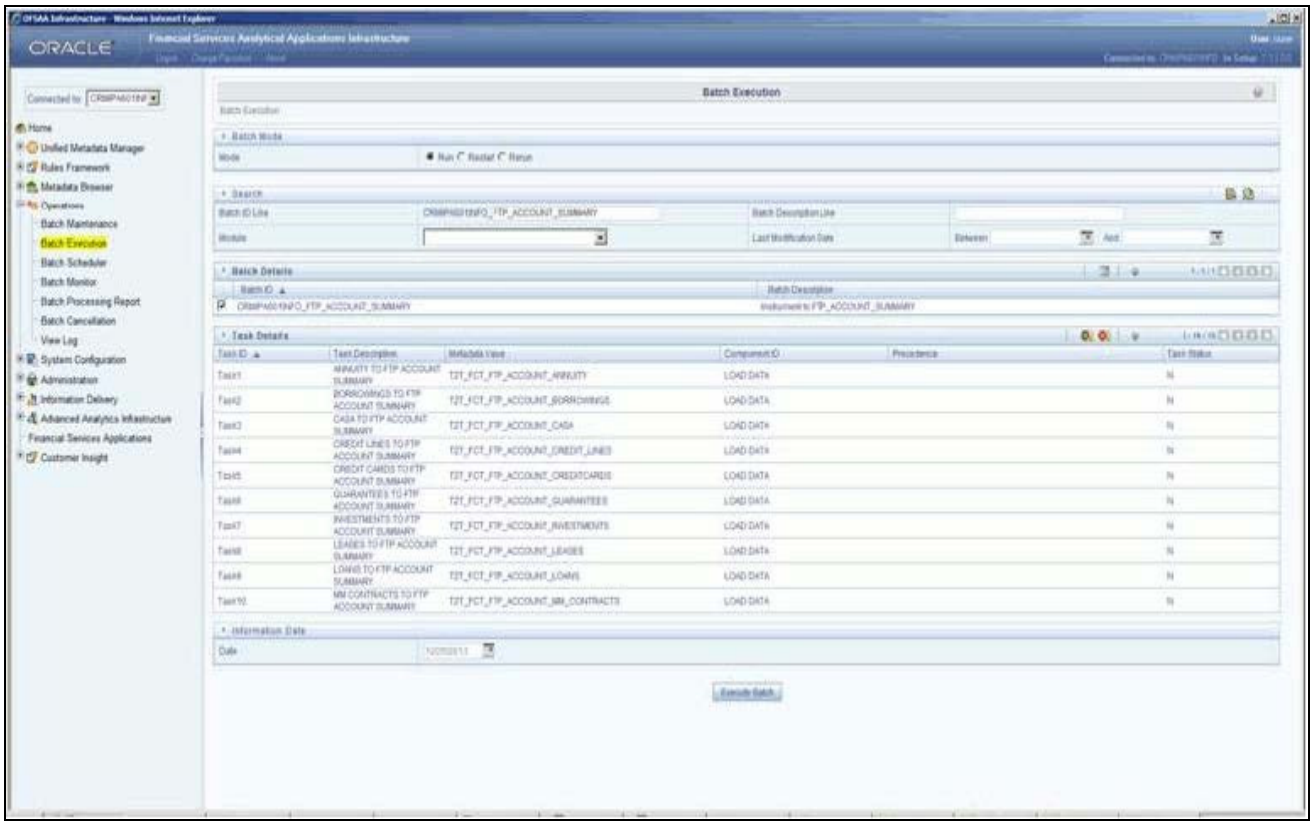


Figure 26. <INFODOM>_FTP_Account_Summary

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the Batch, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select Table to Table from the list.
   - **Source Name** - Select <T2T Source Name> from the list.
   - **File Name** - Select the T2T name for the source stage channel table you want to process.

8. Data file name will be blank for any Table to Table Load mode. Default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, Default value has to be provided. For example, default value is [DRCY]='USD'. Here 'USD' acts as reporting currency parameter to T2T.
9. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
10. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact PFT Account Summary

A seeded batch, **<INFOCOM>_PFT_ACCOUNT_SUMMARY** has to be executed for the required MIS Date.

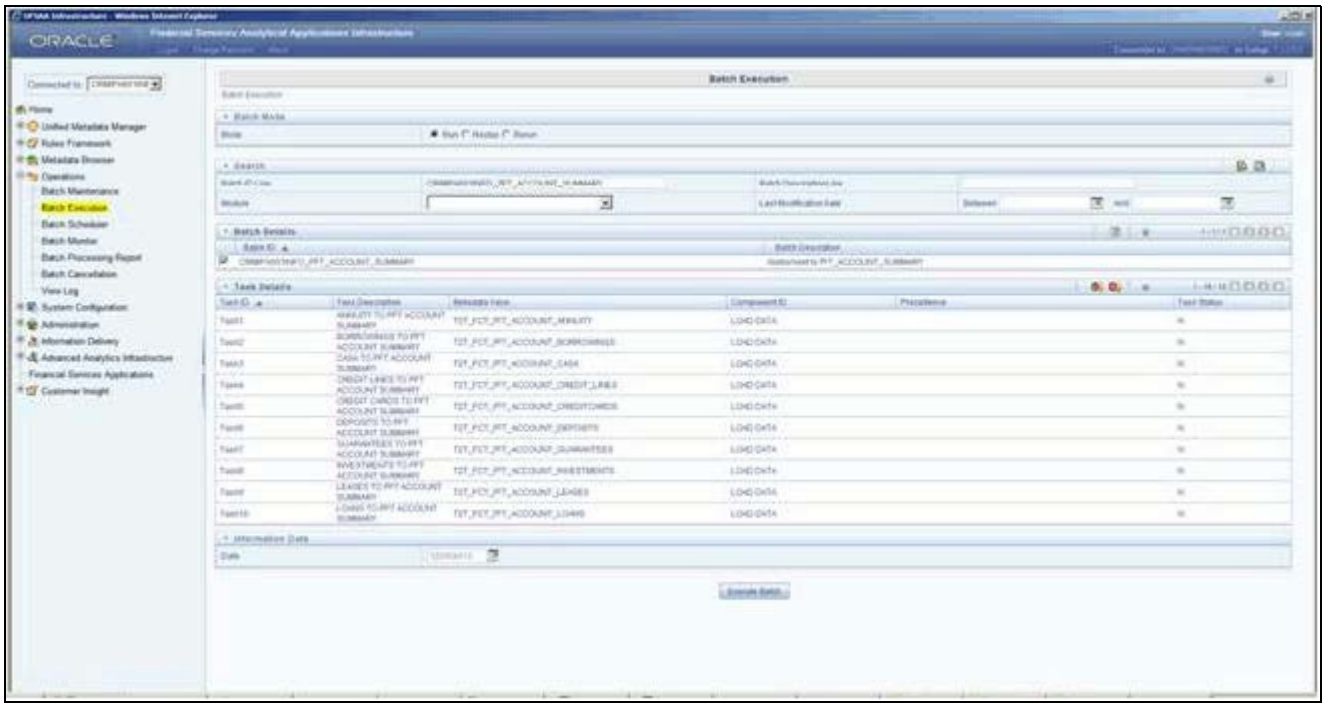


Figure 27. <INFOCOM>_PFT_ACCOUNT_SUMMARY

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Create a new task, enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
    - **Datastore Type** - Select the appropriate datastore from the list.

- **Datastore Name** - Select the appropriate name from the list.
- **IP address** - Select the IP address from the list.
- **Load Mode** - Select Table to Table from the list.
- **Source Name** - Select <**T2T Source Name**> from the list.
- **File Name** - Select the T2T name for the source stage channel table you want to process.

8. Data file name will be blank for any Table to Table Load mode. Default value field will be blank for CRM account summary T2Ts.
9. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
10. Create a Task by repeating steps 4 and 5.
11. Select **Transform Data** from components list.
12. Select the following from the Dynamic Parameters List and click **Save**.
    - **Datastore Type** - Select the appropriate datastore from the list.
    - **Datastore Name** - Select the appropriate name from the list.
    - **IP address** - Select the IP address from the list.
    - **Rule Name** - Select **fn_run_exe_param** from the list.
    - **Parameter List** - Pass the values 1, 180, '$RUNSK= -1', 'USD'.

      It is mandatory to pass all the five parameters. Currently, the first three does not have functional significance. The last two parameters are "Run Skey" and "Reporting Currency" values, that needs to be passed as required. If the batch is being re-run, make sure the run skey value passed is higher than the values (if any) found in "FCT_CRM_ACCOUNT_SUMMARY". If the "run_exe_parameters" table already have an entry for the desired Run Skey, delete the row from the "run_exe_parameters" table before executing the batch.
13. To set this task as a precedent task to each of the other tasks in this batch, click the **Precedence** button in the **Task Details** pane.
14. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact CRM Account Summary

A seeded batch, **<Infodom>_aCRM_CRM_Acc_Summ** has to be executed for the required MIS Date.

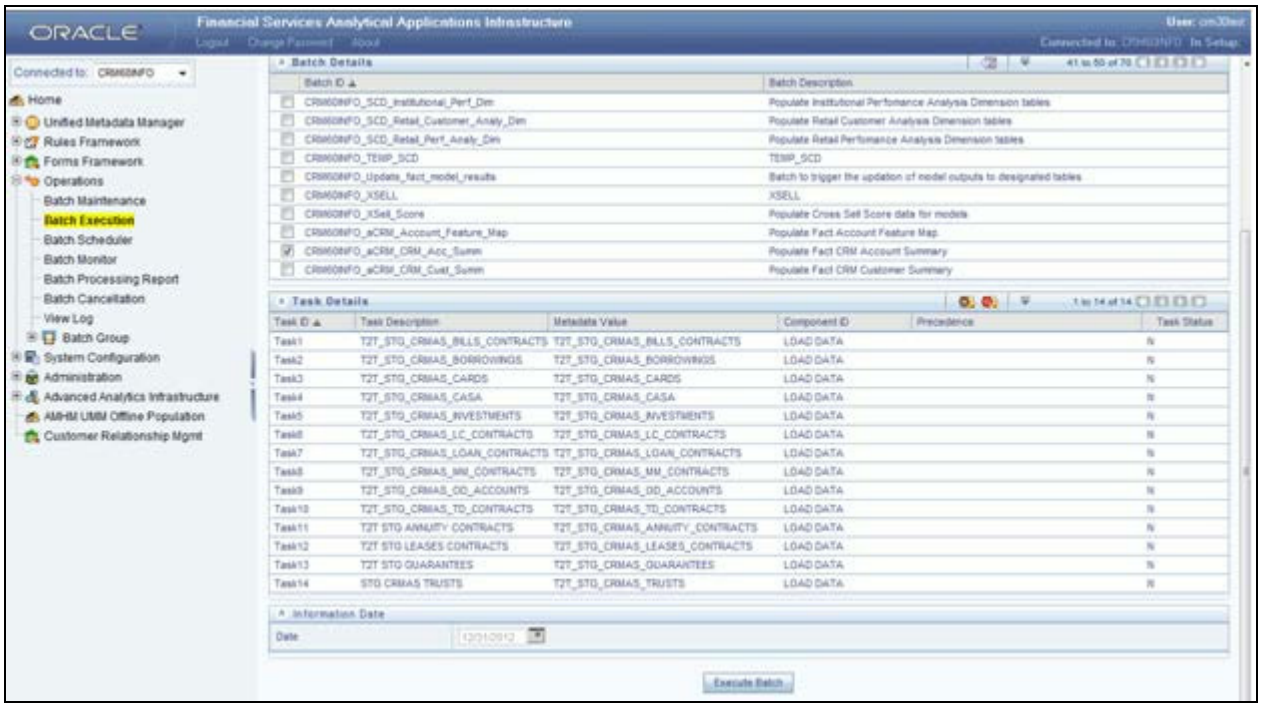


Figure 28. <Infodom>_aCRM_CRM_Acc_Summ

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Create a new task, enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name for the source stage product processor table you want to process.
8. Data file name will be blank for any Table to Table Load mode. Default value field will be blank for CRM account summary T2Ts.
9. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
10. Create a Task by repeating the steps 4 and 5.

11. Select **Transform Data** from components list.
12. Select the following from the Dynamic Parameters List and click **Save**.
    - **Datastore Type** - Select the appropriate datastore from the list.
    - **Datastore Name** - Select the appropriate name from the list.
    - **IP address** - Select the IP address from the list.
    - **Rule Name** - Select **fn_run_exe_param** from the list.
    - **Parameter List** - Pass the values 1, 180, '$RUNSK = -1', 'USD.

      It is mandatory to pass all the five parameters. Currently, the first three does not have functional significance. The last two parameters are "Run Skey" and "Reporting Currency" values, that needs to be passed as required. If the batch is being re-run, please make sure the run skey value passed is higher than the values (if any) found in "FCT_CRM_ACCOUNT_SUMMARY".
13. To set this task as a precedent task to each of the other tasks in this batch, click the **Precedence** button in the **Task Details** pane.
14. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

Note: For a more comprehensive coverage of configuration and execution of a batch, see *Oracle Financial Services Analytical Applications* Infrastructure User Guide.

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 following directory: $FIC_DB_HOME/log/t2t.

The file name will have the batch execution id.

The following tables can be queried for errors:

- FCT_COMMON_ACCOUNT_SUMMARY$
- FCT_CRM_ACCOUNT_SUMMARY$

Account Summary T2Ts

T2T definitions can be retrieved as an excel document for reference from the metadata browser of the Unified Metadata Manager (UMM) component of OFSAAI.

CHAPTER 9 Fact Transaction Summary

This chapter discusses the following topics:

Overview

The Fact Transaction Summary stores data from the stage transactions table for further operation reporting. The data is moved through a T2T process from stage to fact, which ensures that the stage data is available in a single table in the result area.

Table to Table

Table to Table seeded definitions are provided for loading data into Common Account Summary.

Table 20. Common Account Summary T2T Defintions

Sl No.	Source Table Name	T2T Definition Name	Target Table Name
1	STG_TRUSTS_TXNS	STG_TRUSTS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
2	STG_ANNUITY_TXNS	STG_ANNUITY_TXN_F TS	FCT_TRANSACTION_-SUMMARY
3	STG_BILL_CONTRACTS_ TXNS	STG_BILL_CONTRACTS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
4	STG_BORROWING_COMMIT-MENT_TXNS	STG_BORROWING_COMMIT-MENT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
5	STG_BORROWINGS_TXNS	STG_BORROWINGS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
6	STG_CARDS_PAYMENT_ TXNS	STG_CARDS_PAYMENT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
7	STG_CARDS_SETTLEMENT_TXNS	STG_CARDS_SETTLEM ENT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
8	STG_CASA_TXNS	STG_CASA_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
9	STG_COMMITMENT_CON-TRACT_TXNS	STG_COMMITMENT_CON-TRACT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY

Table 20. Common Account Summary T2T Defintions

10	STG_COMMODITIES_TXNS	STG_COMMODITIES_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
11	STG_CORRESPONDENT_ ACCT_TXNS	STG_CORRESPON-DENT_ACCT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
12	STG_CREDIT_DERIVATIVES_TXNS	STG_CREDIT_DERIVA-TIVES_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
13	STG_FOREX_TXNS_FTS	STG_FOREX_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
14	STG_GUARANTEES_TXNS	STG_GUARANTEES_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
15	STG_IJARAH_TXNS	STG_IJARAH_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
16	STG_INTERBANK_TXNS	STG_INTERBANK_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
17	STG_INVESTMENT_TXNS	STG_INVESTMENT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
18	STG_ISTISNA_TXNS	STG_ISTISNA_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
19	STG_LC_TXNS	STG_LC_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
20	STG_LEASES_TXNS	STG_LEASES_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
21	STG_LOAN_CONTRACT_TXNS	STG_LOAN_CON-TRACT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
22	STG_MERCHANT_CARDS_TXNS	STG_MERCHANT_-CARDS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
23	STG_MM_TXNS	STG_MM_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
24	STG_MURABAHAH_TXNS	STG_MURABAHAH_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
25	STG_MUSHARAKAH_TXNS	STG_MUSHARAKAH_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
26	STG_OD_ACCOUNTS_TXNS	STG_MUTUAL_FUNDS _TXNS_FTS	FCT_TRANSACTION_-SUMMARY
27	STG_OD_ACCOUNTS_TXNS	STG_OD_ACCOUNTS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
28	STG_OPTION_CONTRACTS_TXNS	STG_OPTION_CON-TRACTS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY

Table 20. Common Account Summary T2T Defintions

29	STG_RETIREMENT_AC-COUNTS_TXNS	STG_RETIREMENT_AC-COUNTS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
30	STG_SALAM_TXNS	STG_SALAM_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
31	STG_SUKUK_TXNS	STG_SUKUK_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
32	STG_SWAP_ACCOUNT_TXNS	STG_SWAP_ACCOUNT _TXNS_FTS	FCT_TRANSACTION_-SUMMARY
33	STG_TERMDEPOSITS_TXNS	STG_TERMDEPOSITS_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
34	STG_TRADING_ACCOUNT_TXNS	STG_TRADING_AC-COUNT_TXNS_FTS	FCT_TRANSACTION_-SUMMARY
35	STG_FUTURES_TXNS	STG_FUTURES_TXNS_F TS	FCT_TRANSACTION_-SUMMARY
36	STG_MUDARABAH_TXNS	STG_MUDARABAH_TXNS_FTS	FCT_TRANSACTION_-SUMMARY

Executing the Fact Transaction Summary

Fact Transaction Summary table has to be loaded prior loading any of the other Account Summary tables. You can execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations

screen).

Fact Common Account Summary - Batch Execution

A seeded batch, Infodom__STG_TO_FTS has to be executed for the required MIS Date.

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the Batch Name and Description.
3. Click **Save**.
4. Click the check box in the Batch Name container to select the Batch, you created in earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List:
   - Data Store Type

- Datastore Name - Select the appropriate name from the list.
- IP address - Select the IP address from the list.
- Load Mode - Select Table to Table from the list.
- Source Name - Select <T2T Source Name> from the list.
- File Name - Select the T2T name for the source stage channel table you want to process.

8. Click **Save**.

   Data file name will be blank for any Table to Table Load mode.

   Default value refers to currency calculation. If there is any need for currency

   conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY] = 'USD' Here 'USD' acts as reporting currency parameter to T2T.

9. Repeat steps 4 to 8 for adding the remaining T2Ts within the same batch definition.
10. Execute the batch created in the preceding steps.

CHAPTER 10

Customer Summary Population

This chapter explains the process flow for populating Fact Common Customer Summary table.

This chapter covers the following topics:

Overview of Common Customer Summary Tables

Fact Common Customer Summary table stores attributes pertaining to customer related data on an 'as-is' basis received from the source system. Data is populated into this table using T2T.

Customer balances are derived from account summary. Customer relationship table drives the relationship between accounts and customers. Common customer summary data is populated for all the active customers in customer dimension.

Following data flow diagram explains the process flow for populating Fact Common Customer Summary table:

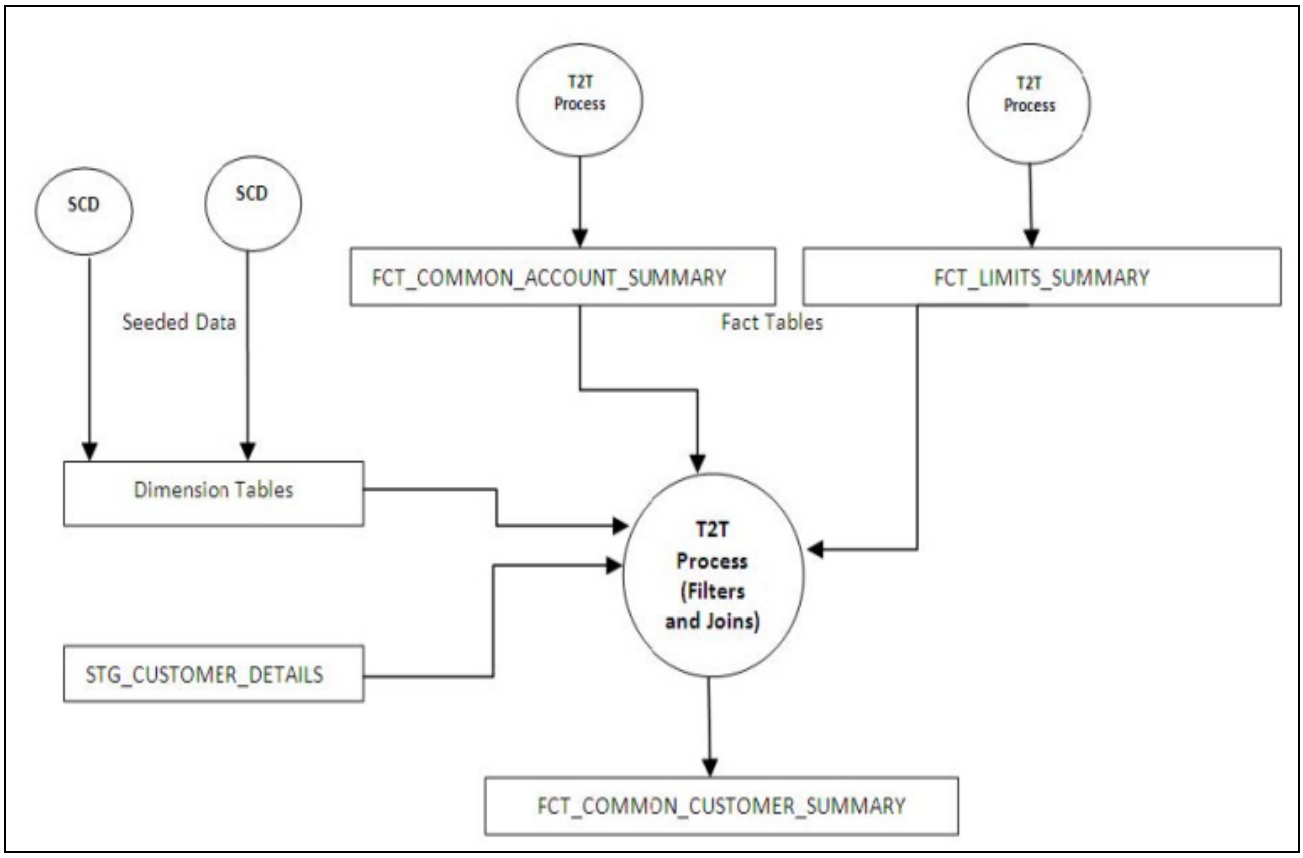


Figure 29. Fact Common Customer Summary dataflow

Prerequisites

Following are the lists of tables used in the population of Fact Common Customer Summary and these tables are required to be loaded prior to running the T2T:

- DIM_CUSTOMER
- DIM_BANDS
- DIM_EDUCATION
- DIM_CUSTOMER_TYPE
- DIM_GENDER
- DIM_INDUSTRY
- DIM_CHANNEL
- DIM_GEOGRAPHY
- DIM_MARITAL_STATUS
- DIM_MANAGEMENT

- DIM_PROFESSION
- DIM_CREDIT_RATING
- DIM_VINTAGE
- DIM_MIGRATION_REASONS
- FCT_COMMON_ACCOUNT_SUMMARY
- FCT_LIMITS_SUMMARY.
- STG_CUSTOMER_DETAILS
- STG_PARTY_RATING_DETAILS
- STG_PARTY_FINANCIALS

Dimensions tables are loaded through the SCD process. The fact tables such as FCT_COMMON_ACCOUNT_SUMMARY and FCT_LIMITS_SUMMARY are loaded from their respective T2T processes.

For more information on SCDs, refer to Chapter 3, "Dimension Loading Process,".

Executing the Customer Summary Population T2T

Fact Common Customer Summary T2T can be executed by executing task present in the seeded batch.

<INFODOM>_aCRM_CommCust_Appln.

Following steps will help you to execute the batch:

1. Navigate to the Batch Execution screen.
2. Select the seeded batch <INFODOM>_aCRM_CommCust_Appln where INFODOM is the information domain where application is installed.
3. Select the AS_OF_DATE for which source customer information is required to be loaded into the table.
4. Click **Execute Batch**.
5. Monitor the status of the batch using Batch Monitor.

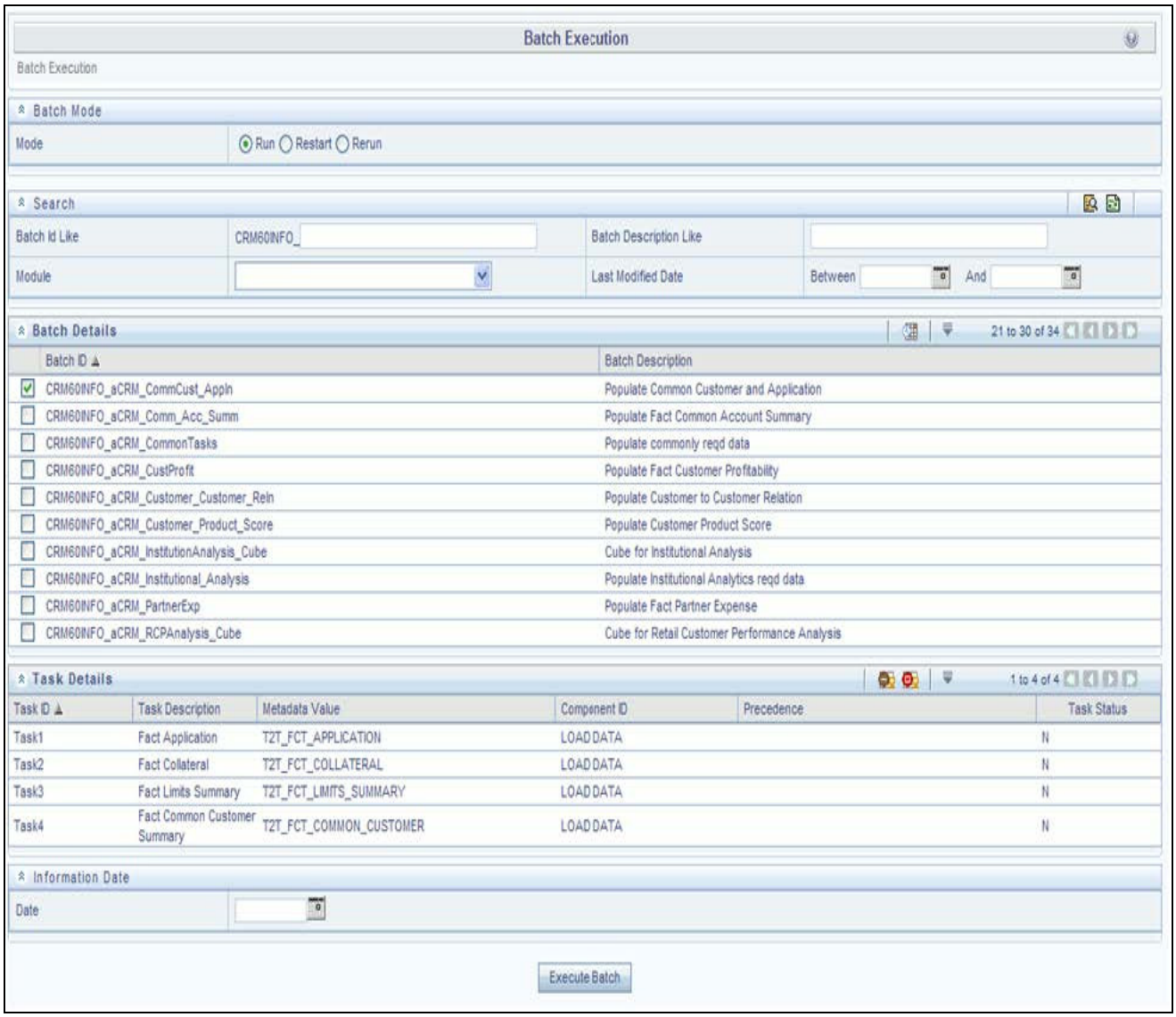


Figure 30. Batch Monitor

Error Messages

Following is the most common error message which will be logged in the T2T log file present in the `$FIC_DB_HOME/logs/t2t folder:`

Unique Constraint Violation: This occurs when attempting re-load or loading existing records for the already executed AS_OF_DATE.

CHAPTER 11

Fact Data Population

This chapter discusses the following topics:

Introduction

This chapter explains all the fact tables which within describe about the seeded T2T Definitions with related Source Table and Destination tables. Prerequisites needed in population of the Fact table and tables required to be loaded prior to running the T2T. Each fact table contains a section on how to execute the T2T component from OFSAA Infrastructure ICC framework and access the execution log to check the execution status.

Fact CRM Customer Summary

Fact CRM Customer Summary entity captures different derived/computed customer attributes pertaining to Customer Insight. Fact Common Customer Summary stores the generic application-agnostic source/raw customer attributes. Fact CRM Customer Summary is a vertical partitioned entity and has relationship to Fact Common Customer Summary.

Load Data into Fact CRM Customer Summary

Customer balances in the Fact CRM Customer Summary entity are derived from account summary. Customer relationship entity drives the relationship between accounts and customers.

Following is the seeded Table-to-Table definitions that loads data related to Fact CRM Customer Summary:

Table 21. Fact CRM Customer Summary definitions

T2T Definition Name	Source Table(s)	Destination Table
T2T_FCT_CRM_CUSTOMER_	STG_CUSTOMER_MASTER	FCT_CRM_CUSTOMER_S
SUMMARY	STG_CUSTOMER_DETAILS	SUMMARY
	FCT_COMMON_ACCOUNT_ SUMMARY	
	FCT_CRM_ACCOUNT_SUMMARY	

Refer to *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *Erwin Data Model* to view the detailed structure of the tables.

Prerequisites

Fact Common Customer Summary entity needs to be populated before executing the Fact CRM Customer Summary T2T. Refer to Chapter 8, "Account Summary Population,"for details related to Fact Common Customer Summary T2T.

Following tables that are used in the population of Fact CRM Customer Summary need to have relevant data prior to executing the T2T:

- STG_CUSTOMER_MASTER - Mandatory
- STG_CUSTOMER_DETAILS - Mandatory
- DIM_DATES - Mandatory
- DIM_CUSTOMER - Mandatory
- FCT_COMMON_ACCOUNT_SUMMARY - Mandatory
- FCT_CRM_ACCOUNT_SUMMARY - Mandatory
- DIM_BANDS - Optional

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Also, see Population of Fact CRM Customer Summary and Fact CRM Account Summary sections for details on populating these fact tables.

Executing the Fact CRM Customer Summary Population T2Ts

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the Operations module of OFSAAI). A seeded batch,

<Infodom>_aCRM_CRM_Cust_Summ has to be executed for the required MIS Date.

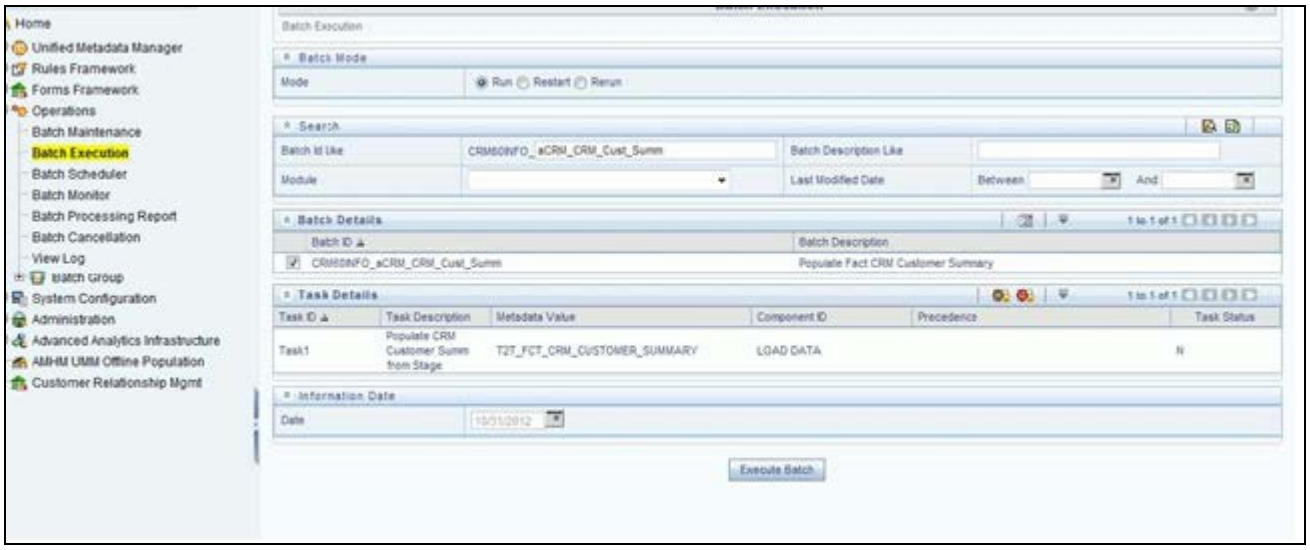


Figure 31. Fact CRM Customer Summary Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select Table to Table from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name "T2T_FCT_CRM_CUSTOMER_SUMMARY" you want to process.
8. Data file name will be blank for any Table to Table Load mode.
9. Default value refers to any parameter that has to be passed to T2T. It has to be blank.

10. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory:
`$FIC_DB_HOME/log/t2t.`

The file name will have the batch execution Id.

This table can be queried for errors: FCT_CRM_CUSTOMER_SUMMARY

> **Note:** For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure* User Guide.

Fact Partner Expense

Fact Partner Expense entity stores expense items like marketing cost, total project expense, business development expense, incentive, and so on that are incurred with the partner of financial institutions. These expenses are captured in the Stage Partner Expense entity for every partner and applicable time period.

Following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 22. Fact Partner Expense definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_PARTNER_EXPENSE	STG_PARTNER_EXPENSE	FCT_PARTNER_EXPENSE

For more information, see Customer Insight Erwin Data Model to view the detailed structure of the tables.

Prerequisites

Following are the lists of tables used in the population of Fact Partner Expense and these are required to be loaded prior to executing the T2T:

- DIM_DATES
- DIM_PARTNER
- STG_PARTNER_EXPENSE

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Partner Expense Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_PartnerExp** has to be executed for the required MIS Date.

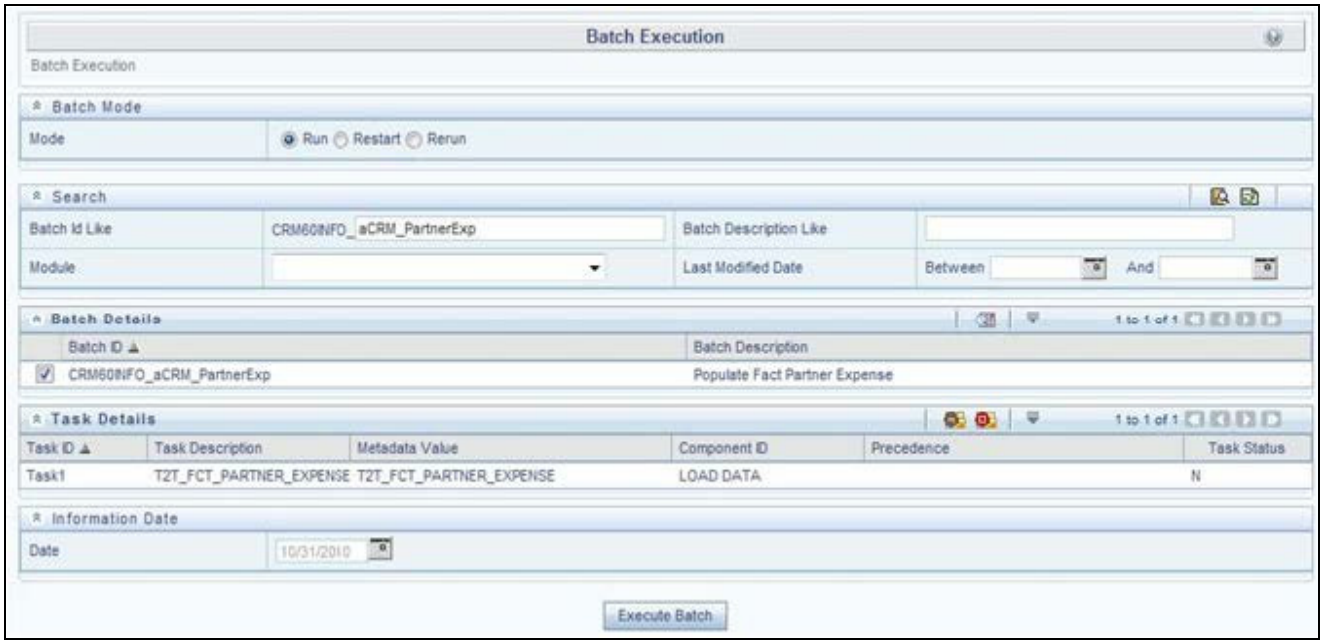


Figure 32. Execute Fact Partner Expense Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.

- **Load Mode** - Select **Table to Table** from the list.
- **Source Name** - Select **<T2T Source Name>** from the list.
- **File Name** - Select the T2T name "T2T_FCT_PARTNER_EXPENSE" you want to process.

8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY]='USD' Here 'USD' acts as reporting currency

   parameter to T2T.

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory:

`$FIC_DB_HOME/log/t2t.`

The file name will have the batch execution id.

This following table can be queried for errors: FCT_PARTNER_EXPENSE$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact Account Feature Map

A product might be facilitated with its own features. Fact Account Feature Map entity stores the mapping between the Account and Product Feature that is the features of the product availed by the customer account. Product processor tables in staging have information related to customer accounts.

Following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 23. Fact Account Feature Map definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_ACCOUNT_FEATURE_MAP	STG_ACCT_FEATURE_MAP	FCT_ACCOUNT_FEATURE_MAP

For more information, see Customer Insight Erwin Data Model to view the detailed structure of the tables.

Prerequisites

Following are the lists of tables used in the population of Fact Account Feature Map and these tables are required to be loaded prior to executing the T2T:

- DIM_DATES
- DIM_PRODUCT_FEATURE
- DIM_ACCOUNT
- DIM_CUSTOMER
- DIM_PRODUCT
- DIM_VENDOR
- DIM_CHANNEL
- STG_ACCT_FEATURE_MAP

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Account Feature Map Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_Account_Feature_Map** has to be executed for the required MIS Date.

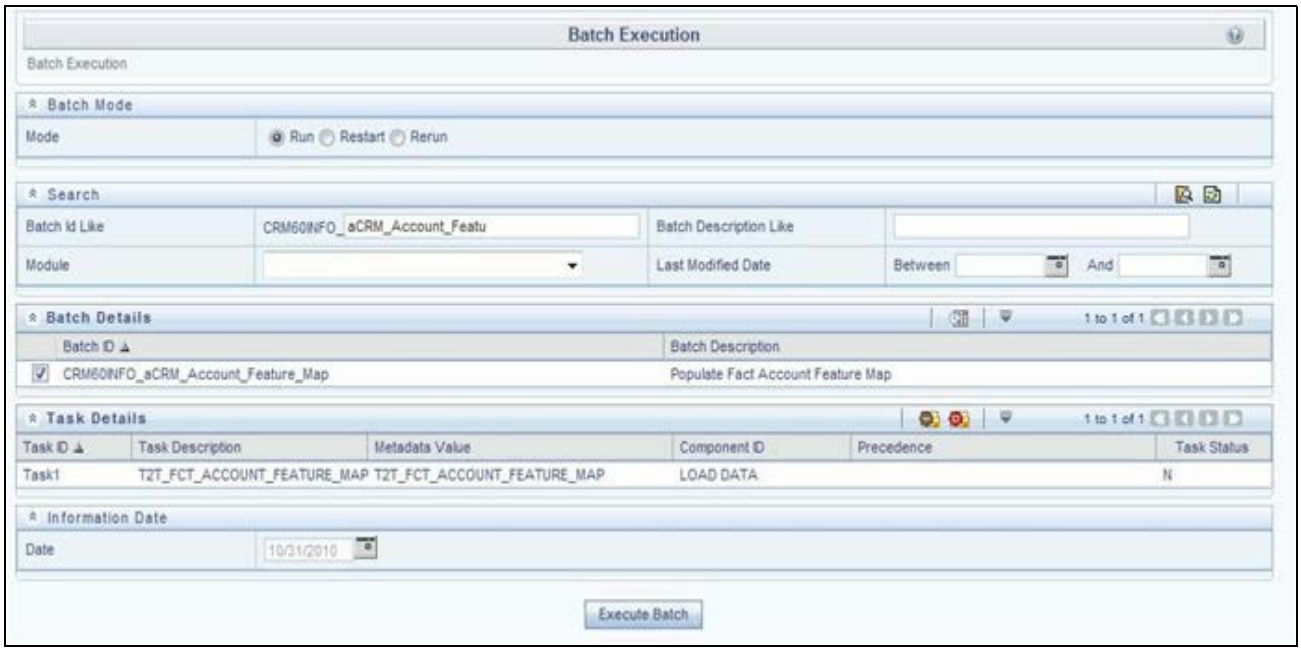


Figure 33. Execute Fact Account Feature Map Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name "T2T_FCT_ACCOUNT_FEATURE_MAP" you want to process.
8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. This should be blank.
9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory:

`$FIC_DB_HOME/log/t2t`.

The file name will have the batch execution id.

This table can be queried for errors: FCT_ACCOUNT_FEATURE_MAP$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact Customer to Customer Relationship

Fact Customer to Customer Relationship entity stores the relationship between the customers.

Following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 24. Fact Customer to Customer Relationship definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_CUST_CUST_RELATION	STG_CUST_CUST_RELATIONSHIP	FCT_ACCOUNT_FEATURE_MAP

For more information, see Customer Insight Erwin Data Model to view the detailed structure of the tables.

Prerequisites

Following are the lists of tables used in the population of Fact Customer to Customer Relationship and these tables are required to be loaded prior to running the T2T:

- DIM_DATES
- DIM_CUSTOMER
- STG_CUST_CUST_RELATIONSHIP

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Customer to Customer Relationship Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_Customer_Customer_Reln - Task1** has to be executed for the required MIS Date.

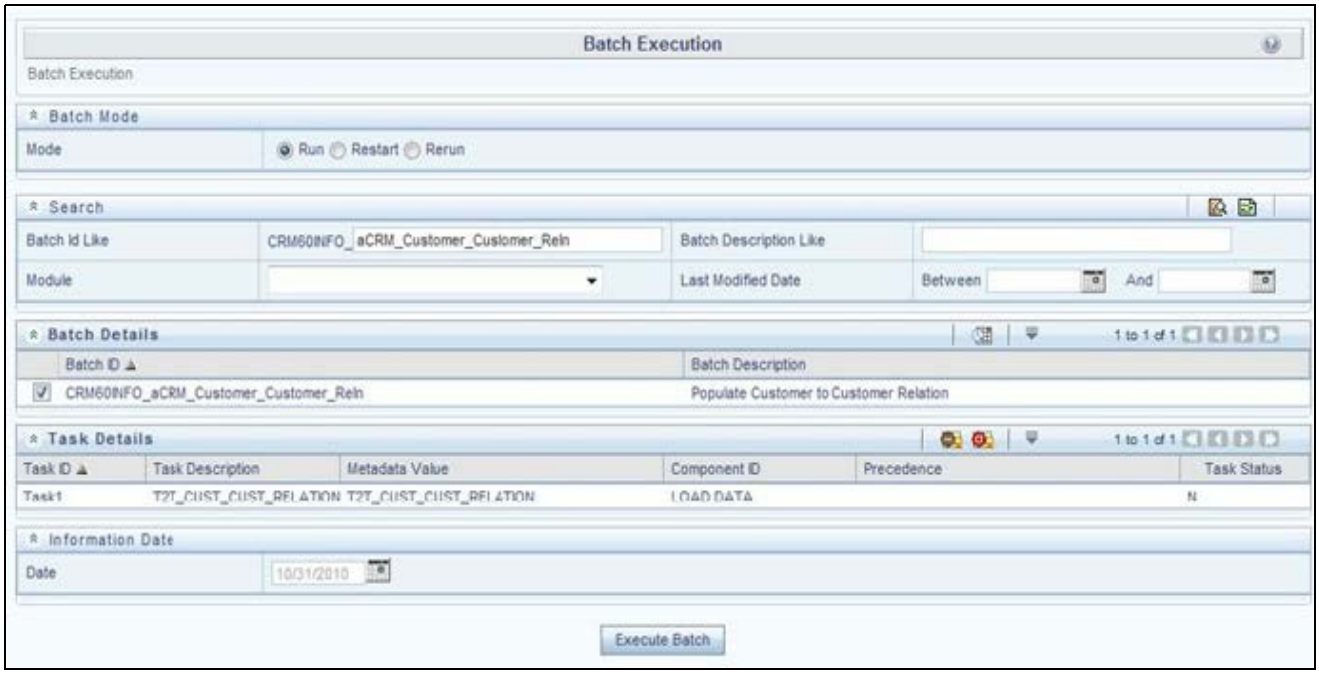


Figure 34. Execute Fact Customer to Customer Relationship Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.

- **File Name** - Select the T2T name "T2T_CUST_CUST_RELATION" you want to process.

8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. This should be blank.

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: `$FIC_DB_HOME/log/t2t`.

The file name will have the batch execution Id.

This table can be queried for errors: FCT_CUST_CUST_RELATIONSHIP$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Fact Opportunity

Fact Opportunity entity stores fact data of an opportunity in an opportunity life cycle. It stores information like cost, current stage of opportunity, current status of opportunity, expected revenue, probability of win, and so on.

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 25. Fact Opportunity definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_STG_OPPORTUNITY	STG_OPPORTUNITY	FCT_OPPORTUNITY

Prerequisites

Following are the lists of tables used in the population of Fact Customer to Customer Relationship and these tables are required to be loaded prior to running the T2T.

- DIM_DATES
- DIM_OPPORTUNITY
- DIM_PRODUCT

- DIM_GEOGRAPHY
- DIM_PROSPECT
- DIM_CUSTOMER
- DIM_SALES_REPRESENTATIVE
- DIM_OPTY_WL_REASON
- DIM_SALES_STAGE
- DIM_OFFER
- DIM_LOB
- STG_OPPORTUNITY

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Opportunity Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_Institutional_Analysis – Task1** has to be executed for the required MIS Date.

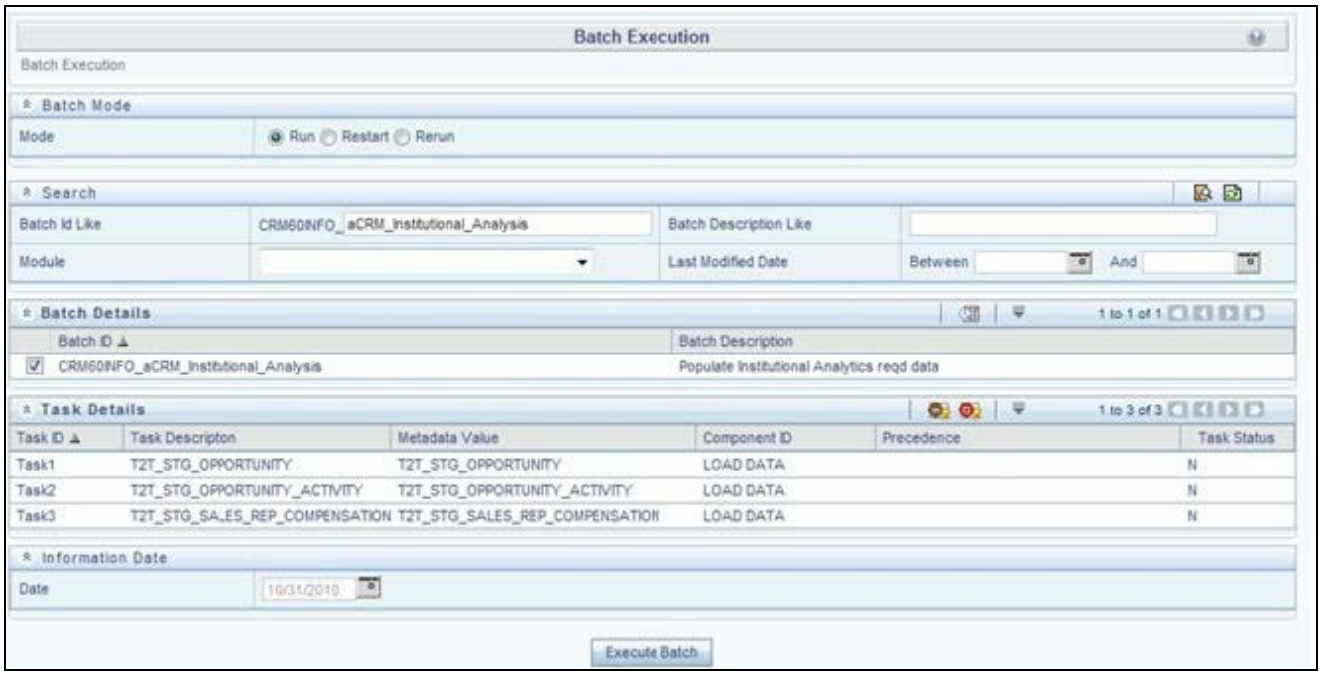


Figure 35. Execute Fact Opportunity Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name "T2T_STG_OPPORTUNITY" you want to process.
8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY]='USD'

   Here, 'USD' acts as reporting currency parameter to T2T
9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory:

`$FIC_DB_HOME/log/t2t.`

The file name will have the batch execution Id.

This table can be queried for errors: FCT_OPPORTUNITY$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact Opportunity Activity

Fact Opportunity Activity entity stores the fact data related to activities that are performed for each opportunity. It stores information like start & end dates, priority and severity of activity, cost of activity, and so on.

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 26. Fact Opportunity Activity definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_STG_OPPORTUNITY_ ACTIVITY	STG_OPPORTUNITY_ACTIVITY	FCT_OPPORTUNITY_ACTIVITY

To view the detailed structure of this table, refer to *Erwin Data Model*.

Prerequisites

Following are the lists of tables used in the population of Fact Opportunity Activity and these tables are required to be loaded prior to running the T2T.

- DIM_DATES
- DIM_OPPORTUNITY
- DIM_ACTIVITY_TYPE
- DIM_PRODUCT
- DIM_SALES_REPRESENTATIVE
- DIM_SALES_STAGE
- STG_OPPORTUNITY_ACTIVITY

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, refer to Chapter 3, "Dimension Loading Process,".

For details on populating DIM_DATES dimension table, refer to Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification*.

Executing the Fact Opportunity Activity Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_Institutional_Analysis – Task2** has to be executed for the required MIS Date.

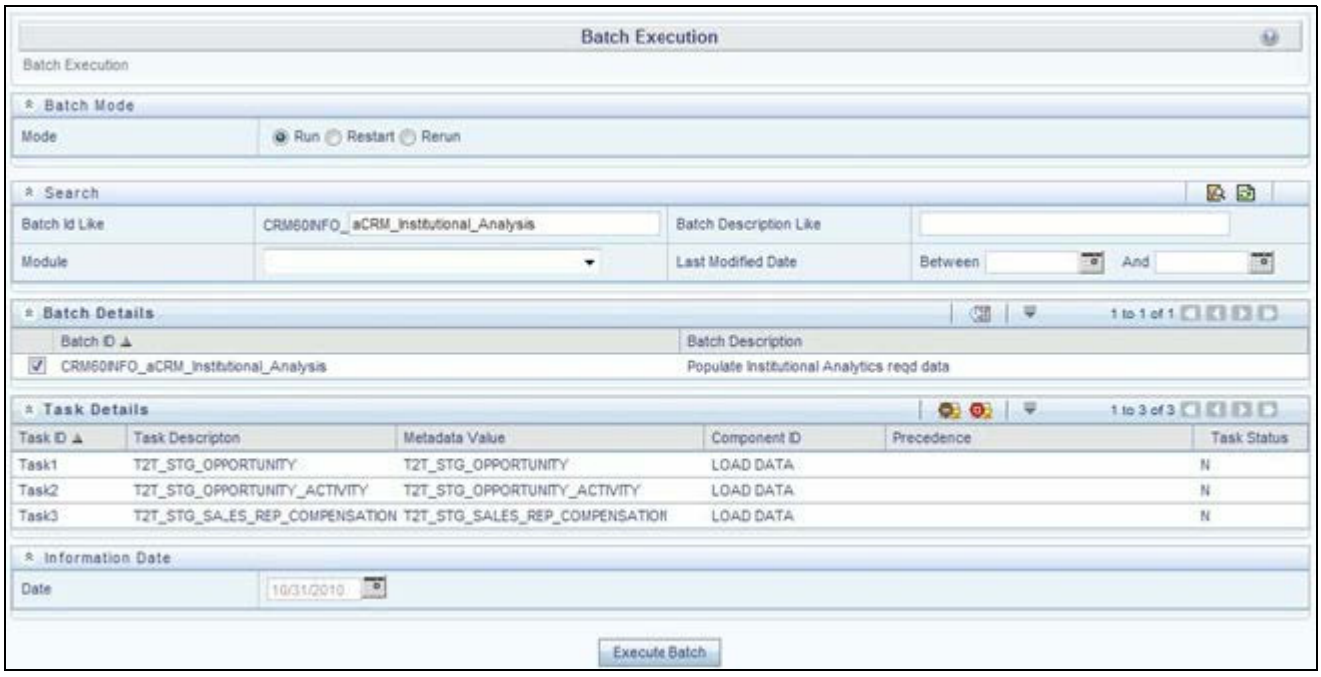


Figure 36. Execute Fact Opportunity Activity Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name 'T2T_STG_OPPORTUNITY_ACTIVITY' you want to process.
8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY]='USD'

   Here, 'USD' acts as reporting currency parameter to T2T

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: $FIC_DB_HOME/log/t2t.

The file name will have the batch execution Id.

The following tables can be queried for errors: FCT_OPPORTUNITY_ACTIVITY$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact Sales Representative Compensation

Fact Sales Representative Compensation entity stores the sales incentive compensation paid for a sales representative against a product.

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 27. Fact Sales Representative Compensation

T2T Definition Name	Source Staging Table	Destination Table
T2T_STG_SALES_REP_ COMPENSATION	STG_SALES_REP_ COMPENSATION	FCT_SALES_REP_ COMPENSATION

For more information, see Customer Insight Erwin Data Model to view the detailed structure of the earlier tables.

Prerequisites

Following are the lists of tables used in the population of Fact Sales Representative Compensation and these tables are required to be loaded prior to running the T2T.

- DIM_DATES
- DIM_PRODUCT
- DIM_SALES_REPRESENTATIVE
- STG_SALES_REP_COMPENSATION

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, see Dimension Tables Population.

For details on populating DIM_DATES dimension table, see Chapter 4, *Time Dimension Population*.

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Sales Representative Compensation Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen).

A seeded batch, **<Infodom>_aCRM_Institutional_Analysis – Task3** has to be executed for the required MIS Date.

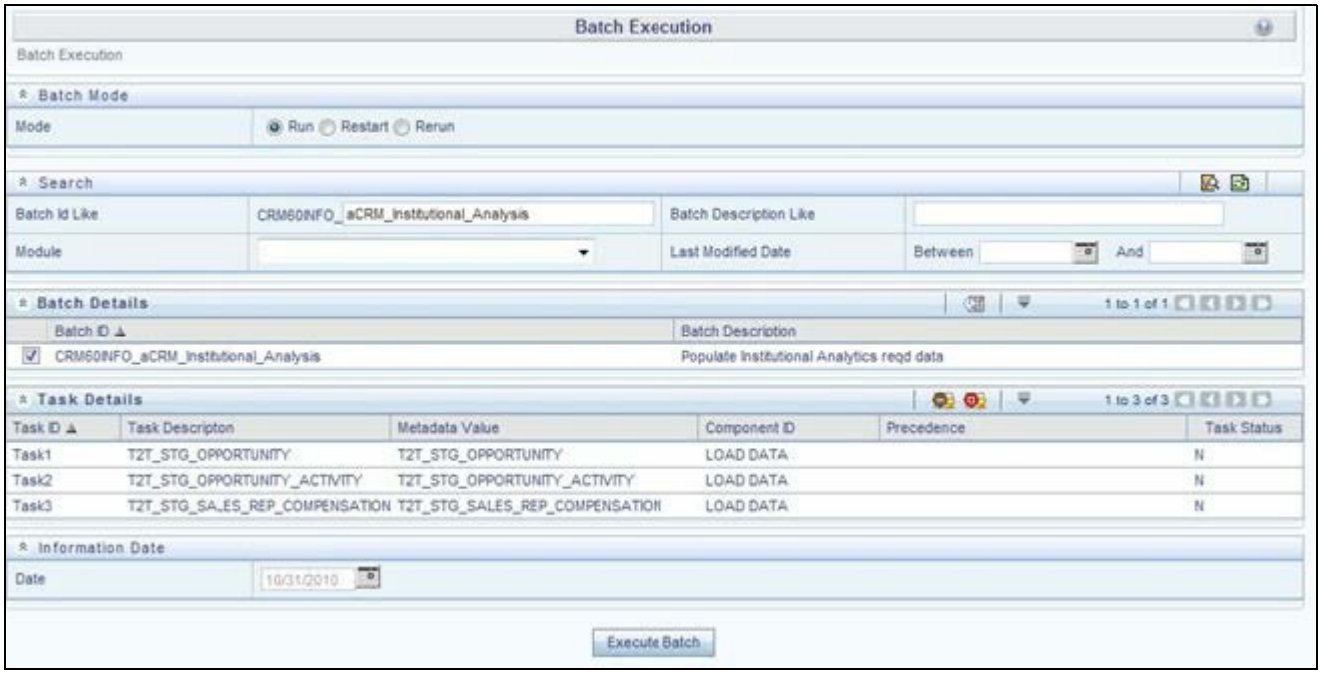


Figure 37. Execute Fact Sales Representative Compensation Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.**IP address** - Select the IP address from the list.

- **Load Mode** - Select **Table to Table** from the list.
- **Source Name** - Select **<T2T Source Name>** from the list.
- **File Name** - Select the T2T name 'T2T_STG_SALES_REP_COMPENSATION', you want to process.

8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY]='USD' Here, 'USD' acts as reporting currency parameter to T2T

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: $FIC_DB_HOME/log/t2t.

The file name will have the batch execution id.

The following tables can be queried for errors: FCT_SALES_REP_COMPENSATION$

Note: For more information on configuration and execution of a batch, see Oracle Financial Services Analytical Applications Infrastructure User Guide.

Fact Application

Fact Application entity stores the fact data of applications like application details, current stage, status, rejection reason, time-taken in each stage, and so on.

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 28. Fact Application definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_APPLICATION	STG_APPLICATION	FCT_APPLICATION

For more information and to view the detailed structure of the earlier tables, see *Customer Insight Erwin Data Model*.

Prerequisites

Following are the lists of tables used in the population of Fact Application. These tables are required to be loaded prior to running the T2T:

- DIM_DATES
- DIM_APPLICATION_TYPE
- DIM_PRODUCT
- DIM_CREDIT_OFFICER
- DIM_CUSTOMER
- DIM_CHANNEL
- DIM_CREDIT_CENTER
- DIM_DECISION_STATUS
- DIM_GEOGRAPHY
- DIM_INDUSTRY
- DIM_APPLICATION_REJECT_REASONS
- DIM_DEVIATION_REASONS
- DIM_SALES_REPRESENTATIVE
- DIM_ACCOUNT
- DIM_PROSPECT
- DIM_BANDS
- STG_APPLICATION

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, see Dimension Tables Population.

For details on populating DIM_DATES dimension table, see Chapter 4, "Time Dimension Population,".

For identifying fields required in Channel Transaction tables in staging for the purpose of Customer Insight Application(s), refer to *Download Specification.*

Executing the Fact Application Population T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through *Operations* module), a seeded batch, **<Infodom>_aCRM_CommCust_Appln – Task1** has to be executed for the required MIS Date.

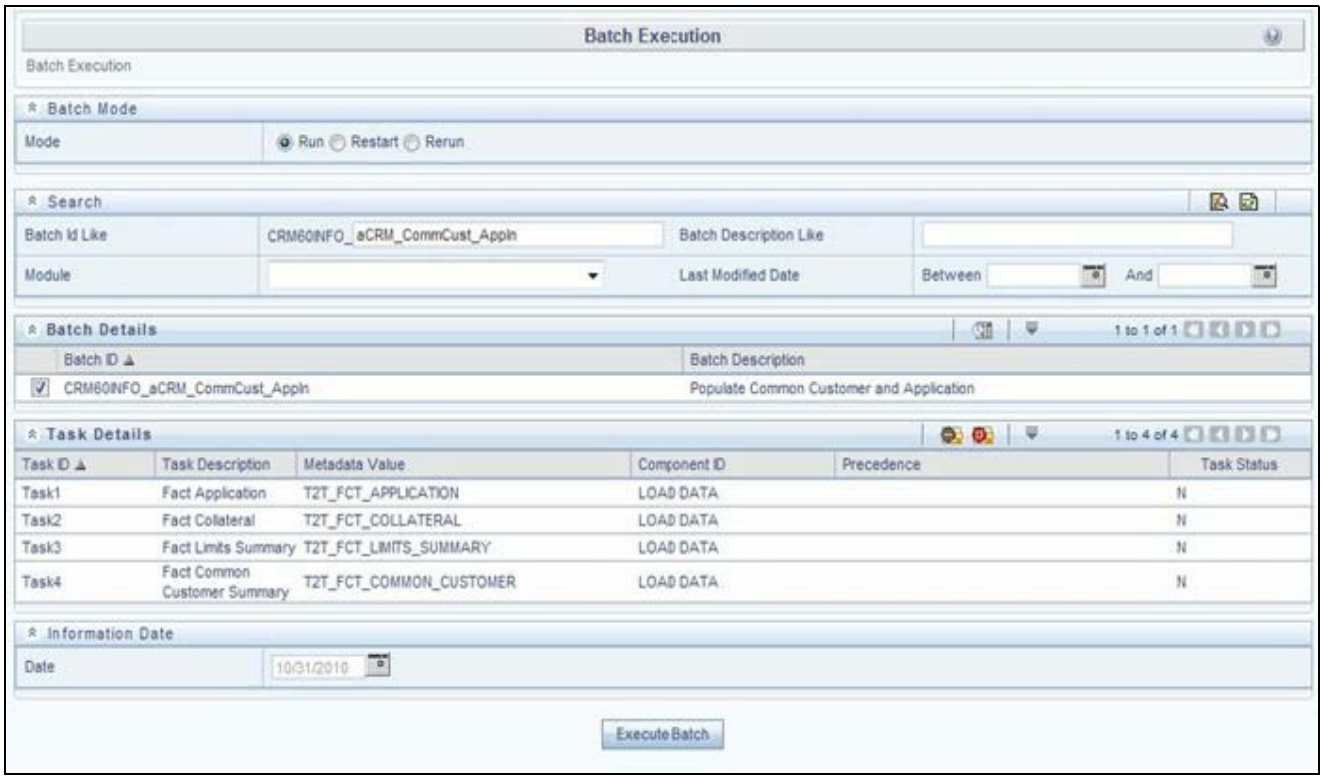


Figure 38. Execute Fact Application Population

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name 'T2T_FCT_APPLICATION', you want toprocess.
8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

For example, default value is [DRCY]='USD'

Here, 'USD' acts as reporting currency parameter to T2T.

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: $FIC_DB_HOME/log/t2t.

The file name will have the batch execution id.

The following tables can be queried for errors: FCT_APPLICATION$

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Account Manager Relation

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 29. Account Manager definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_ACCOUNT_MANAGERS_REL	STG_ACCOUNT_MGR_REL	FCT_ACCOUNT_MGR_REL

Note: For more information and to view the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

Prerequisites

The following are the lists of tables used in the population of Account Manager Relation. These tables are required to be loaded prior to running the T2T.

- Dim_account
- Dim_customer
- Dim_dates
- Dim_management
- Stg_account_mgr_rel

Executing the Account Manager Relation T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded batch, **<Infodom>_ACCOUNT_MANAGER_REL** has to be executed for the required MIS Date.

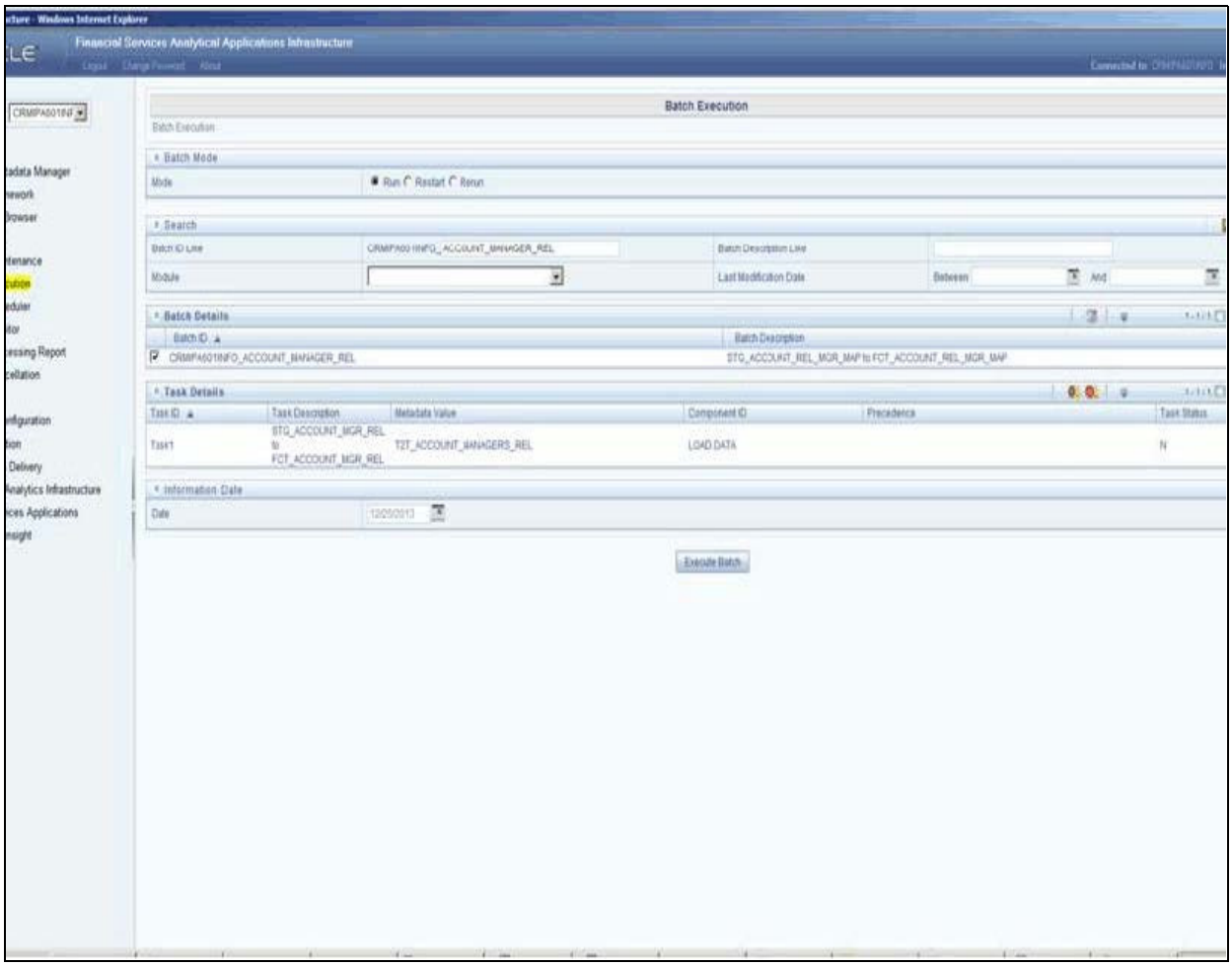


Figure 39. Execute Account Maneger Relation

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.

- **Datastore Type** - Select the appropriate datastore from the list.
- **Datastore Name** - Select the appropriate name from the list.
- **IP address** - Select the IP address from the list.
- **Load Mode** - Select **Table to Table** from the list.
- **Source Name** - Select **<T2T Source Name>** from the list.
- **File Name** - Select the T2T name 'T2T_ACCOUNT_MANAGERS_REL', you want to process.

8. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY] = 'USD'.

   Here, 'USD' acts as reporting currency parameter to T2T.

9. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: `$FIC_DB_HOME/log/t2t`.
The file name will have the batch execution Id.

Note: For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Management Forecast

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 30. Management Forecast definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_MANAGEMENT_FCAST	STG_MGMT_FORECAST	FCT_MGMT_FORECAST

Note: For more information and to view the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

Prerequisites

The following are the lists of tables used in the population of Account Manager Relation. These tables are required to be loaded prior to running the T2T.

- DIM_ORG_STRUCTURE
- DIM_DATES
- DIM_CUSTOMER
- DIM_LOB
- DIM_PRODUCT
- DIM_ORG_UNIT
- DIM_ACCOUNT
- STG_MGMT_FORECAST

Executing the Management Forecast T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded batch, **<Infodom>_MANAGEMENT_FCAST** has to be executed for the required MIS Date.

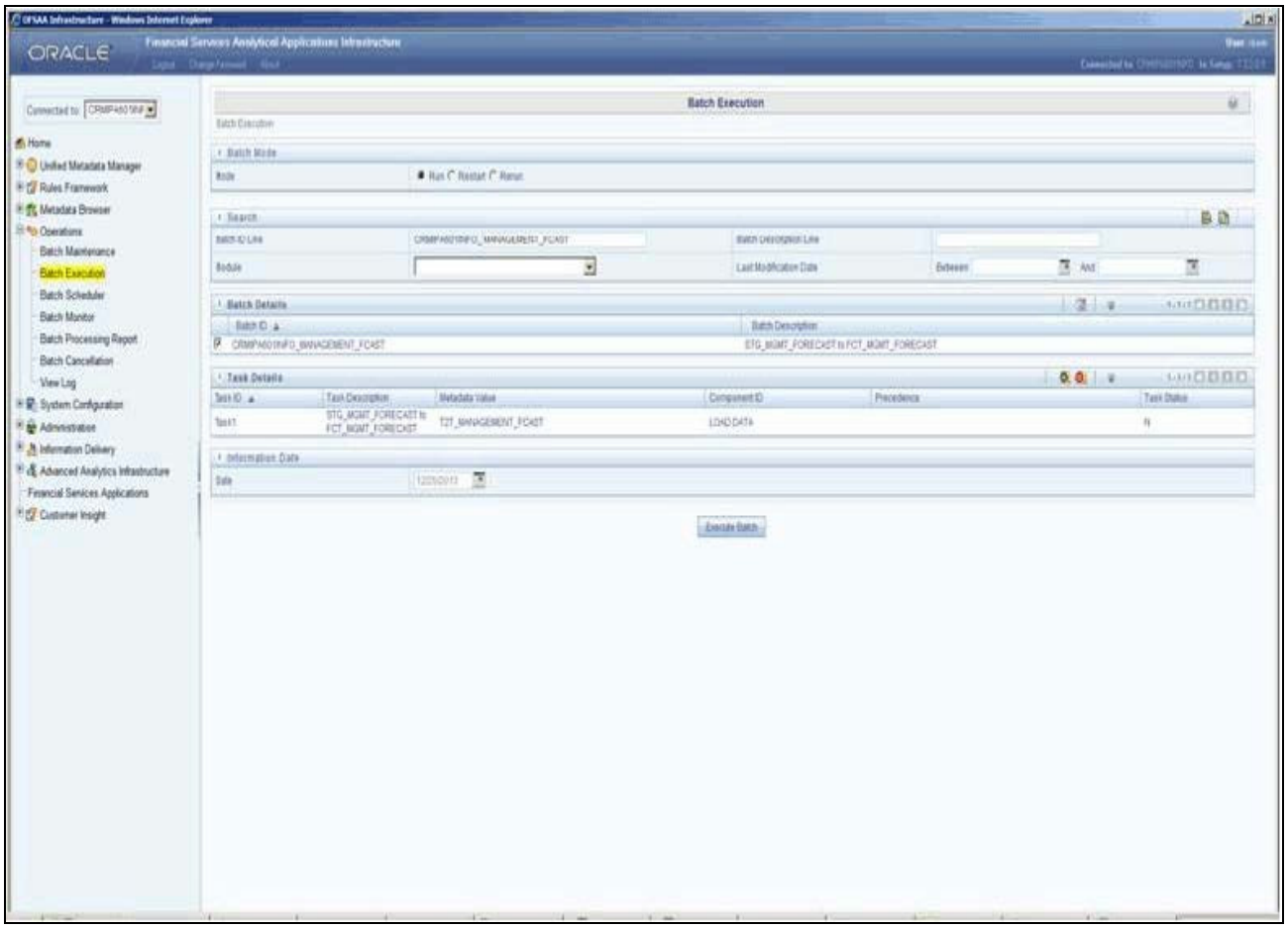


Figure 40. Execute Management Forecast

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.
2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
8. Select the following from the Dynamic Parameters List and click **Save**.
    - **Datastore Type** - Select the appropriate datastore from the list.
    - **Datastore Name** - Select the appropriate name from the list.
    - **IP address** - Select the IP address from the list.
    - **Load Mode** - Select **Table to Table** from the list.

- **Source Name** - Select **<T2T Source Name>** from the list.
- **File Name** - Select the T2T name 'T2T_MANAGEMENT_FCAST', you want to process.

9. Data file name will be blank for any Table to Table Load mode.

   Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

   For example, default value is [DRCY] = 'USD'.

   Here, 'USD' acts as reporting currency parameter to T2T.

10. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: `$FIC_DB_HOME/log/t2t`.

The file name will have the batch execution Id.

> **Note:** For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Fact Account Customer Relation

The following table lists the seeded T2T Definitions with related Source Table and Destination tables:

Table 31. Fact Account Customer Relation definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_ACCT_CUST_ RELATIONSHIP	STG_CUSTOMER_ RELATIONSHIP	FCT_ACCT_CUST_ RELATIONSHIP

Note: For more information and to view the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

Prerequisites

The following are the lists of tables used in the population of Account Manager Relation. These tables are required to be loaded prior to running the T2T.

- DIM_DATES

- DIM_CUSTOMER
- DIM_ACCOUNT
- DIM_GEOGRAPHY
- DIM_MANAGEMENT
- DIM_CHANNEL
- DIM_PRODUCT
- DIM_DATA_ORIGIN
- STG_CUSTOMER_RELATIONSHIP

Executing the Account Customer Relation T2T

To execute the T2T component from OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded batch, **<INFODOM>_ACCT_CUST_RELATIONSHIP** has to be executed for the required MIS Date.

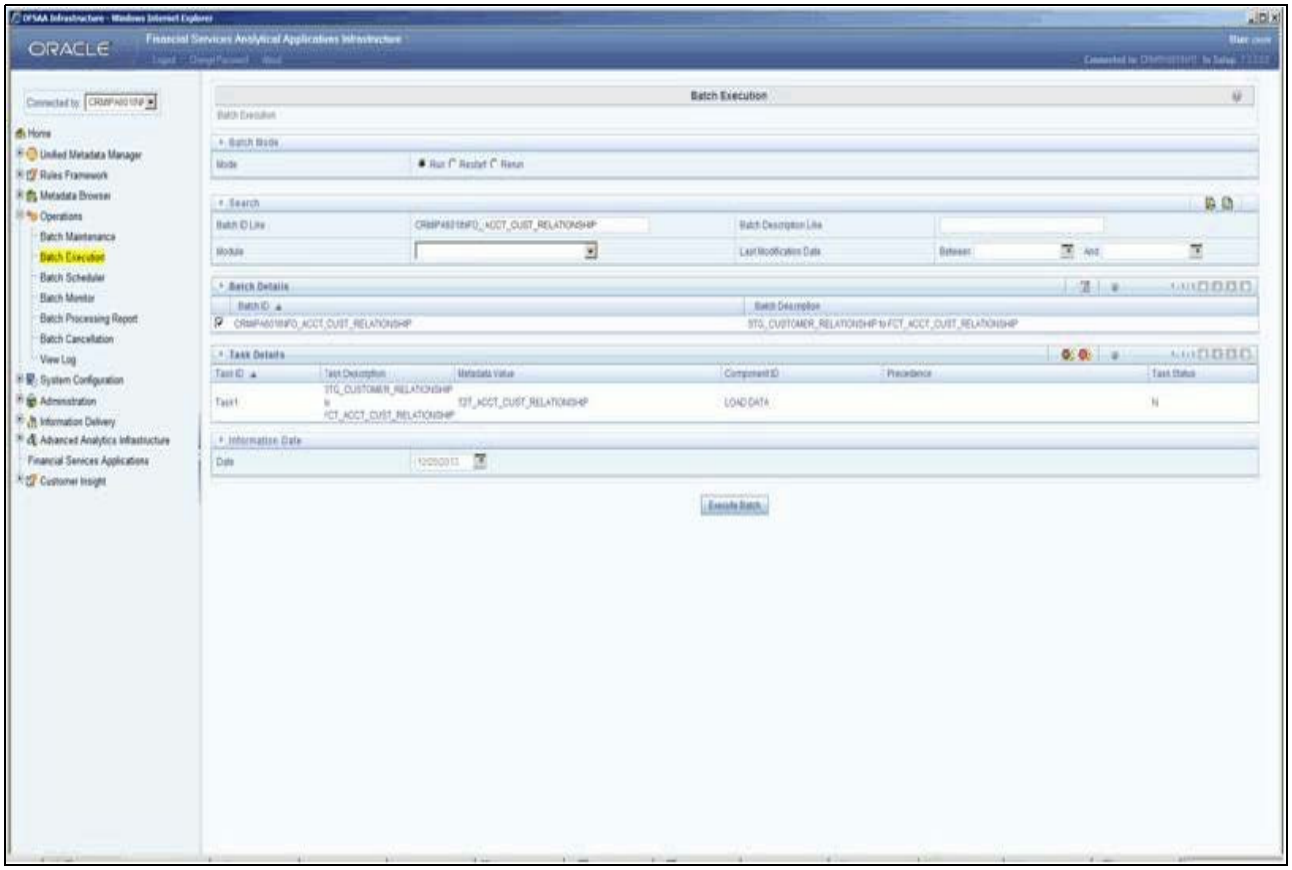


Figure 41. Execute Account Customer Relation

Alternatively, following steps will help you create a new batch:

1. From the **Home** menu, click **Operations** and select **Batch Maintenance**.

2. Click **New Batch** ('+' symbol in Batch Name container). Enter the **Batch Name** and **Description**.
3. Click **Save**.
4. Click the check box in the **Batch Name** container to select the **Batch**, you created in the earlier step.
5. Enter the **Task ID** and **Description**.
6. Select **Load Data** from the Components list.
7. Select the following from the Dynamic Parameters List and click **Save**.
   - **Datastore Type** - Select the appropriate datastore from the list.
   - **Datastore Name** - Select the appropriate name from the list.
   - **IP address** - Select the IP address from the list.
   - **Load Mode** - Select **Table to Table** from the list.
   - **Source Name** - Select **<T2T Source Name>** from the list.
   - **File Name** - Select the T2T name T2T_ACCT_CUST_MANAGERS_RELATIONSHIP', you want to process.

     Data file name will be blank for any Table to Table Load mode.

     Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, Default value has to be provided.

     For example, default value is [DRCY] = 'USD'.

     Here, 'USD' acts as reporting currency parameter to T2T.
8. Execute the batch created in the preceding steps.

For more information, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen.

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 following directory: `$FIC_DB_HOME/log/t2t`.

The file name will have the batch execution Id.

> **Note:** For more information on configuration and execution of a batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Fact Account Profitability

Fact Account Profitability entity stores fact data for reporting line items of revenue, costs, and expense related to each customer account. The data into this table is populated from other fact tables like FCT_COMMON_ACCOUNT_SUMMARY, FCT_PFT_ACCOUNT_SUMMARY, FCT_FTP_ACCOUNT_SUMMARY, FCT_REG_CAP_ACCOUNT_SUMMARY, and FCT_ECO_CAP_ACCOUNT_SUMMARY.

The following table lists the seeded Post Load Transformation Definition with related Source Table and Destination tables:

Table 32. Fact Account Profitability

DT Definition Name	Source Tables	Destination Table
FN_FCT_ACCOUNT_PFT	FCT_COMMON_ACCOUNT_SUMMARY	FCT_ACCOUNT_PROFITABILITY
	FCT_PFT_ACCOUNT_SUMMARY	
	FCT_FTP_ACCOUNT_SUMMARY	
	FCT_REG_CAP_ACCOUNT_SUMMARY	
	FCT_ECO_CAP_ACCOUNT_SUMMARY	
	FCT_PFT_CUSTOMER_SUMMARY	

Refer to *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *Erwin Data Model* to view the detailed structure of the tables.

Information from account summary fact tables are populated to Fact Account Profitability through a mapping process. Reporting line dimension is mapped to measures present in account summary. A PL/SQL procedure then populates the fact by reading the mapping definition.

Reporting line dimension is created/maintained from Attribute Member Hierarchy Maintenance (AMHM) component of OFSAAI. A Reporting line item represents a revenue, costs, or expenses. Rollup signage is set as an attribute for a reporting line. To know more about AMHM, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

The Account summary tables contain the revenue, costs, or expenses measures pertaining to an Account. Map Maintenance component of OFSAAI is used to map the measures of account summary tables (represented in a measure hierarchy) to reporting line hierarchy. A pre-defined mapping "Reporting Line Mapping" is seeded along with the application installer. Reporting Line Hierarchy and Reporting Line Measure Hierarchy are the two hierarchies which are used for the mapping. Reporting Line Hierarchy is a parent child hierarchy which is based on Reporting Line Dimension entity.

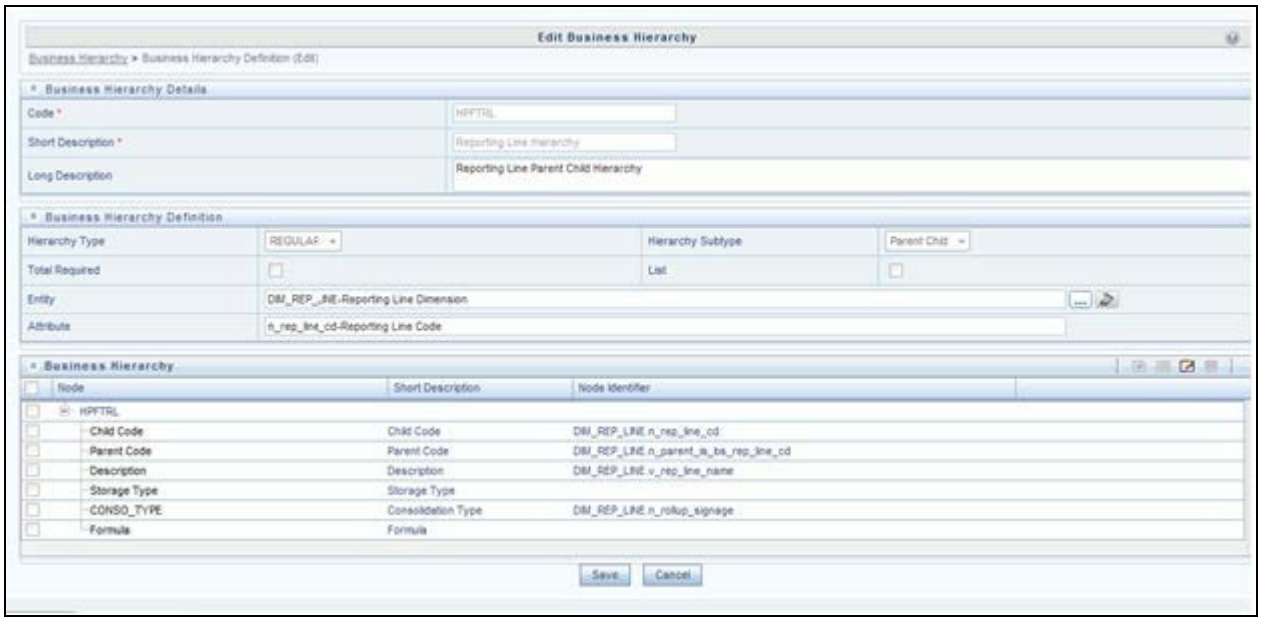


Figure 42. Reporting Line Hierarchy

Reporting Line Measure hierarchy is a Non Business Intelligence Enabled Hierarchy which is based on measures from the Account Summary tables.

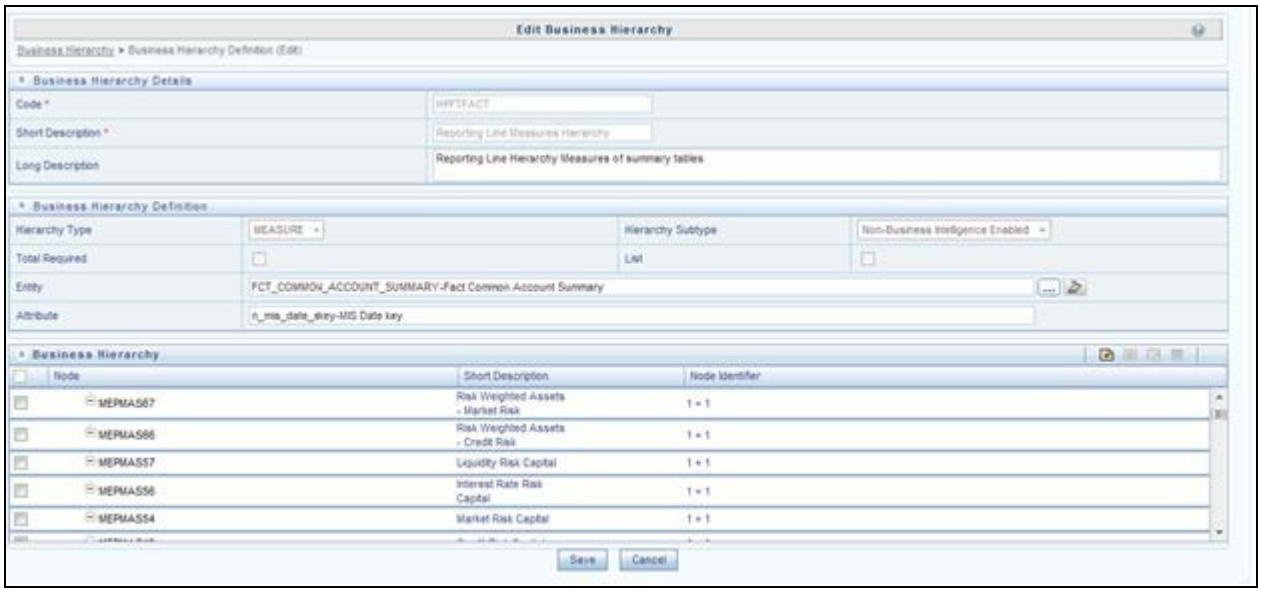


Figure 43. Reporting Line Hierarchy

A seeded map is configured between the Reporting Line Hierarchy and Reporting Line Measure Hierarchy from Map Maintenance of OFSAAI.

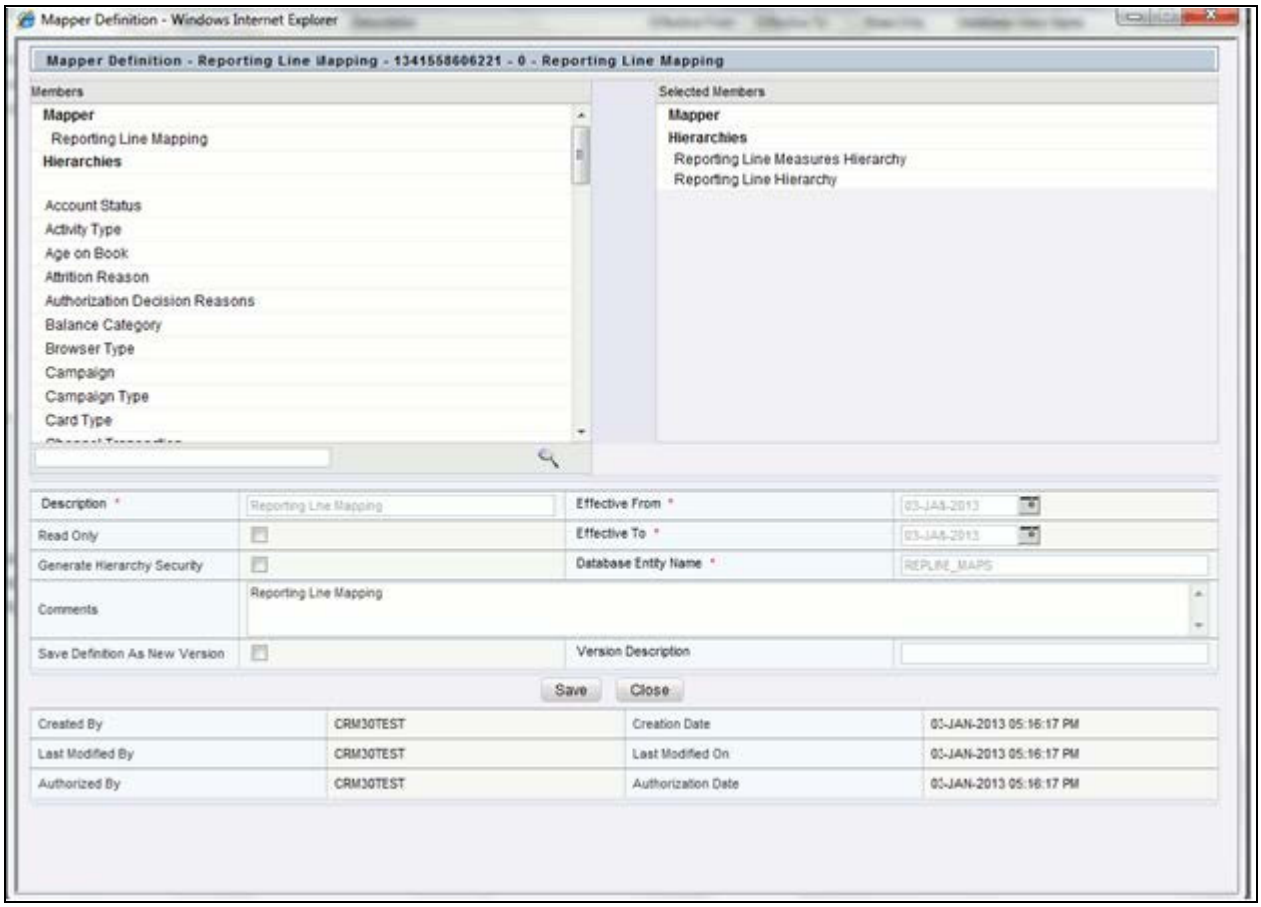


Figure 44. Mapper Definition

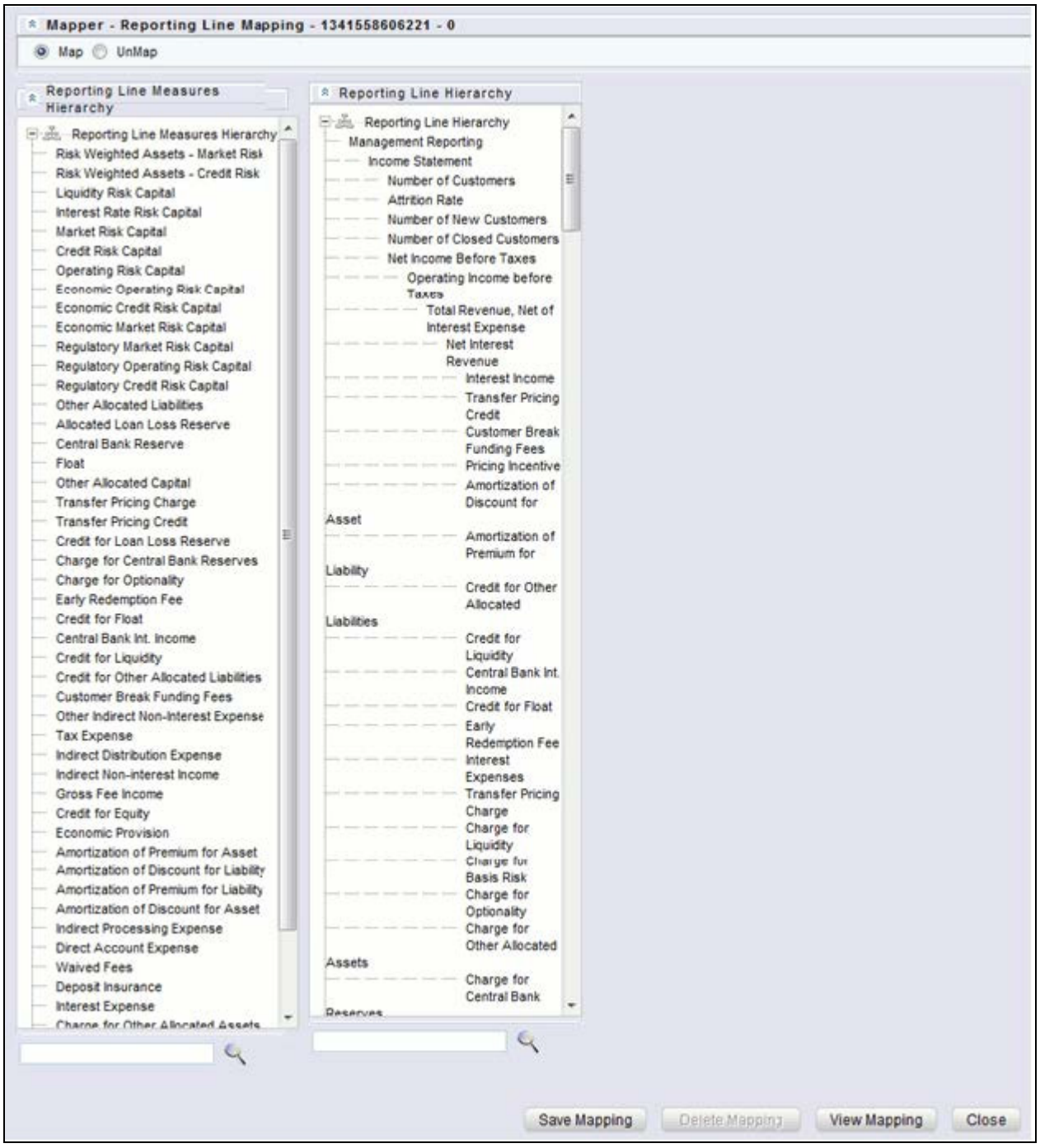


Figure 45. Mapper Definition - Reporting Line Hierarchy

For more information on defining/maintaining Mapper, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Steps to Define Mapping for Custom Reporting Line Items

Follow the below steps to define mapping for Custom Reporting Line items:

1. Add Custom Reporting Line or Modify existing Reporting Line.
2. Add Custom Reporting Line Hierarchy or modify existing seeded reporting line hierarchy.
3. Execute the seeded batch **<INFODOM>_ Repline_Dimension_Update** specifying the Reporting line hierarchy as parameter to batch.
4. Modify the seeded Business Metadata.
5. Map Maintenance.

Add Custom Reporting Line or Modify existing Reporting Line

Custom Reporting Lines can be added or modified from AMHM.

Following are the seeded attributes of Reporting Line Dimension:

- Financial Element Code
- GL Account Code
- Rollup Signage

Figure 46. Attributes

A Reporting line can be added or modified from the *Members* screen as shown below. To modify the existing reporting line, select the member by selecting the adjacent check box and select the **Edit** button on the menu bar.

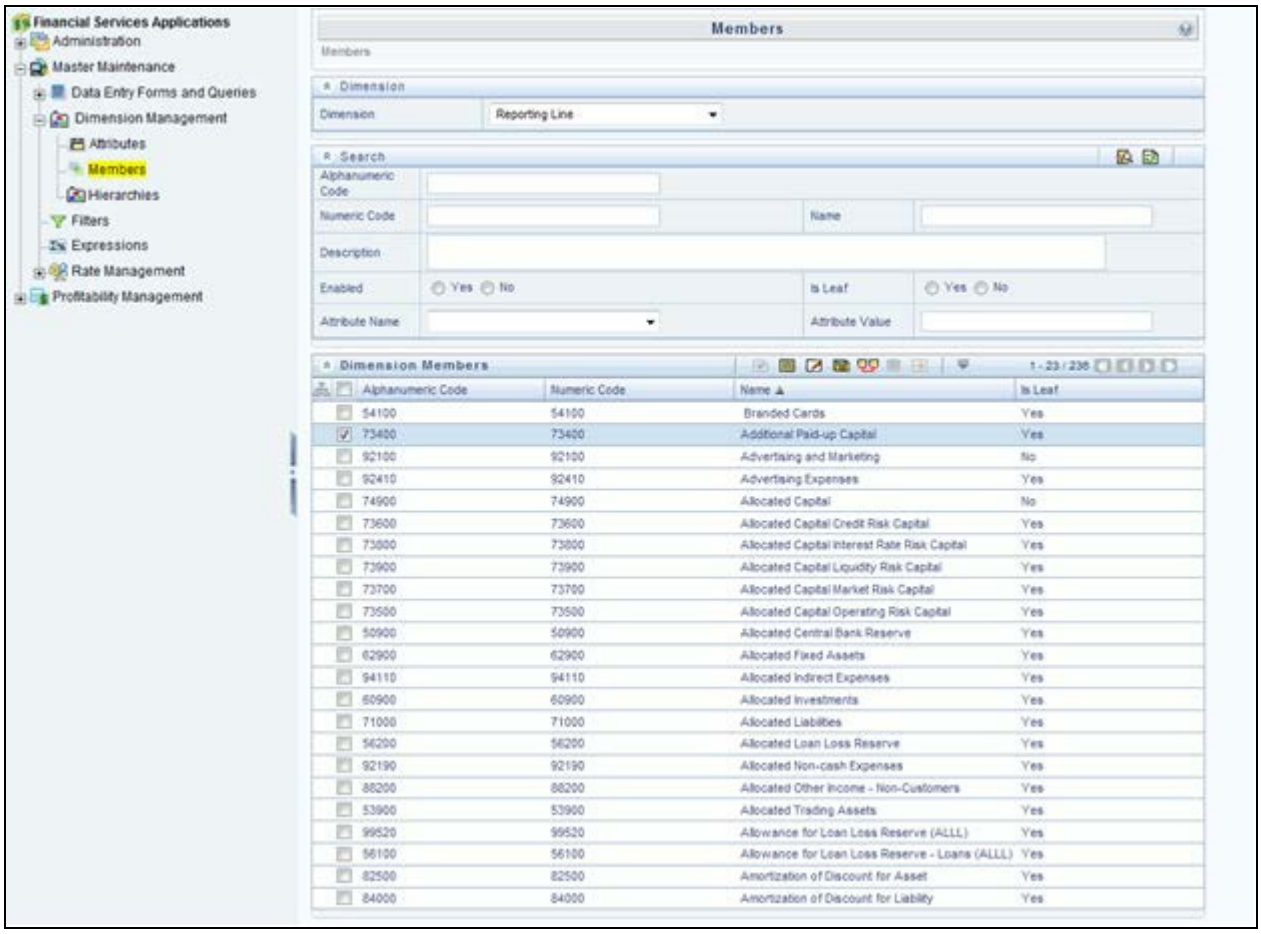


Figure 47. Members

To add a new reporting line, follow these steps:

1. Select **Add** button from the *Members* screen.

   The *Member Definition (New Mode)* screen is displayed.

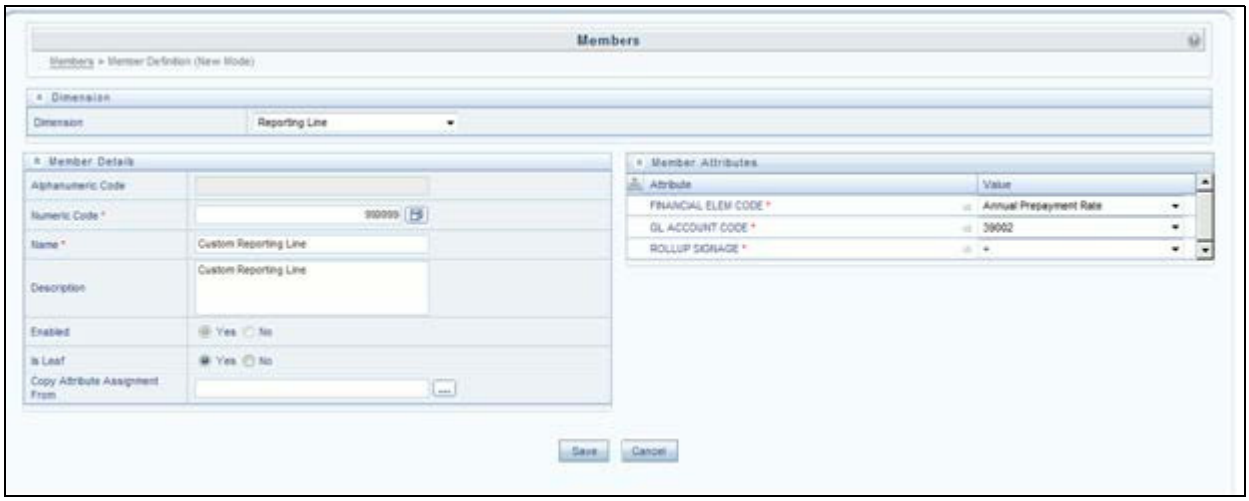


Figure 48. Member Definition (New Mode)

2. In the *Member Definition (New Mode)* screen:
   - Enter **Numeric Code**.
   - Enter the **Name** of the custom reporting line.
   - Enter the **Description** of the custom reporting line.
   - Select Yes, if the custom reporting line has to be **Enabled** or not.
   - Select Yes, if the custom reporting line **Is Leaf** or not.
   - Select the Attributes for the reporting line member.
   - Save the Member definition.

To modify a reporting line, follow these steps:

1. Click **Edit** button from the *Members* screen.

   The *Member Definition (Edit Mode)*screen is displayed.

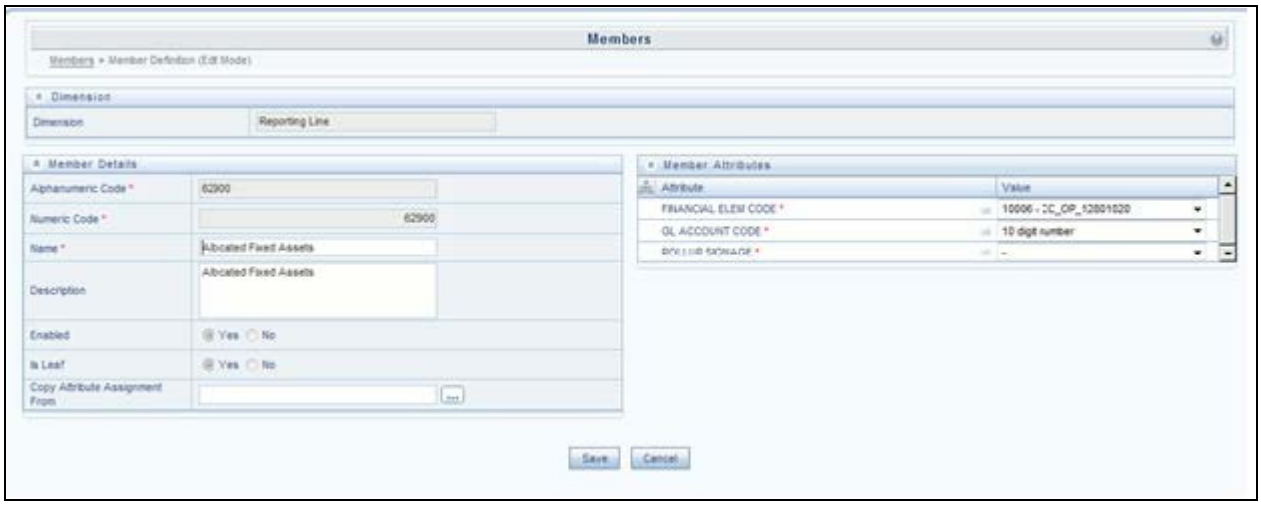


Figure 49. Member Definition (Edit Mode)

2. In the *Member Definition (New Mode)* screen:
   - Enter **Numeric Code**.
   - Enter the **Name** of the custom reporting line.
   - Enter the **Description** of the custom reporting line.
   - Select Yes, if the custom reporting line has to be **Enabled** or not.
   - Select Yes, if the custom reporting line **Is Leaf** or not.
   - Select the Attributes for the reporting line member.
   - Save the Member definition.

To modify a reporting line, follow these steps:

In the *Member Definition (Edit Mode)* screen, perform the following as required:

1. Modify the **Name** of the custom reporting line.
2. Modify the **Description** of the custom reporting line.
3. Modify the selection of the radio button in the **Enabled** field.
4. Modify the selection of the radio button in the **Is Leaf** field.
5. Modify the Attributes for the reporting line member.
6. Save the Member definition.

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

Add Custom Reporting Line Hierarchy or Modify Existing Seeded Reporting Line Hierarchy

To create a new Reporting Line Hierarchy, follow these steps:

1. Click **Add** button from the menu. The *Hierarchy Definition (New Mode)* screen is displayed.

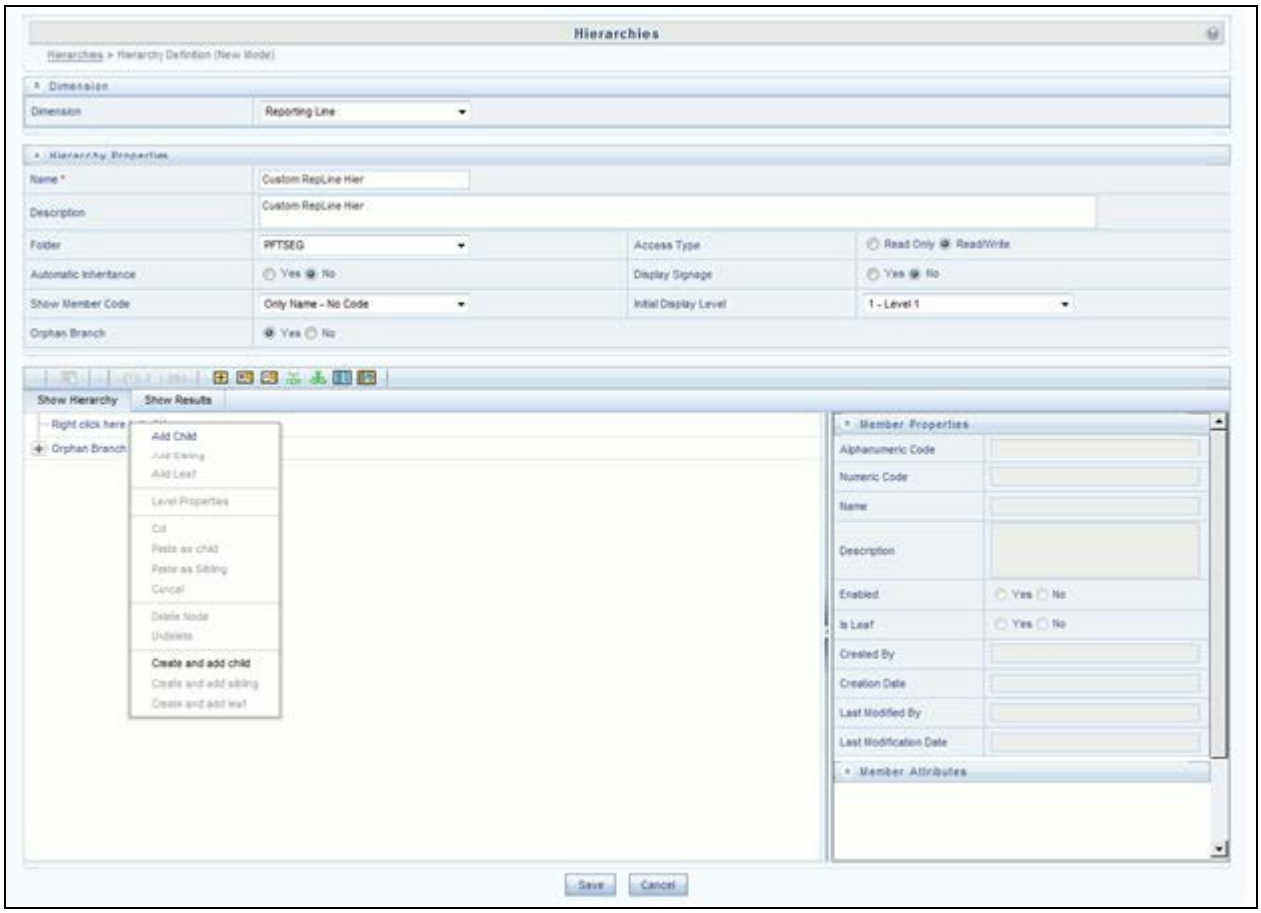


Figure 50. Hierarchy Definition (New Mode)

2. Enter the details in the required fields, and click **Save**.

   Note: Alternatively, insert scripts and update scripts can be prepared into tables DIM_REPORTING_LINE_B, DIM_REPORTING_LINE_TL, DIM_REPORTING_LINE_ATTR, and DIM_REPORTING_LINE_HIER for adding any new custom reporting lines or modifying an existing reporting line.

Execute the seeded batch **<Infodom>_ Repline_Dimension_Update** specifying the Reporting line hierarchy as parameter to batch. It populates data into DIM_REP_LINE table. This batch invokes the DT fn_rep_line_parent_child.

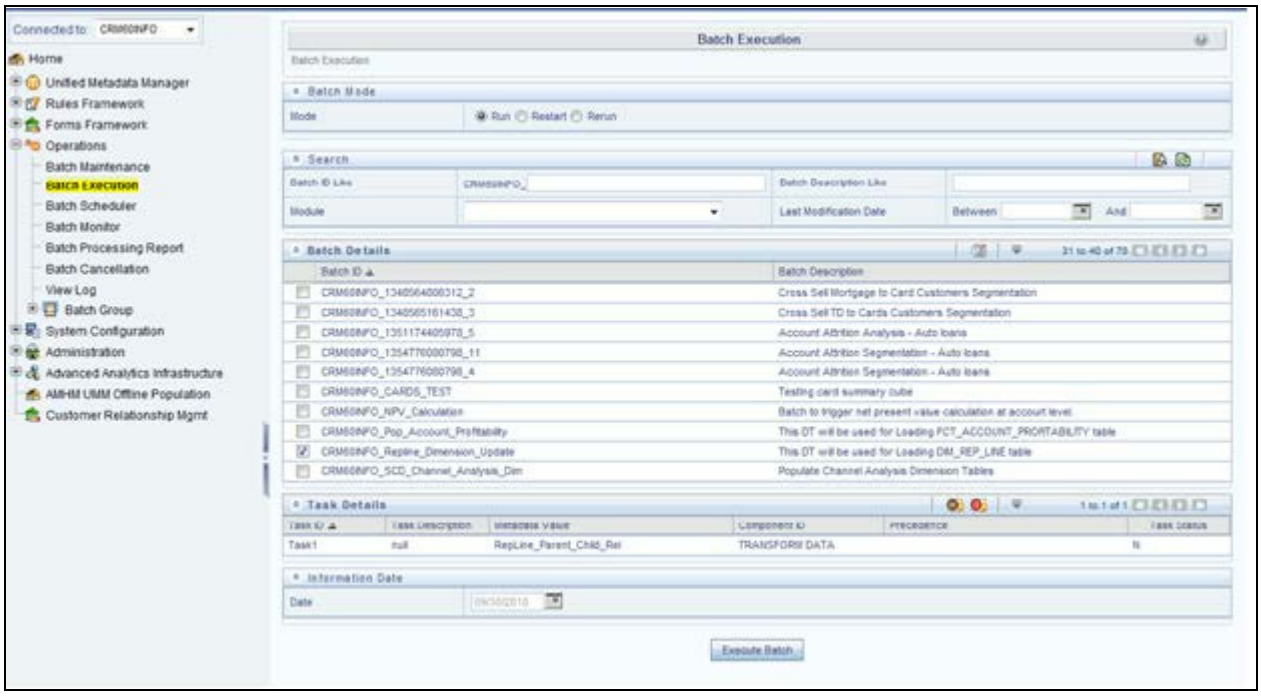


Figure 51. Rep Line batch execution

To modify existing seeded Reporting Line Hierarchy:

1. Select the check box adjacent to the Reporting Line Hierarchy to be modified.
2. Click **Edit** button from the menu.
3. Modify the Hierarchy as required and click **Save**.

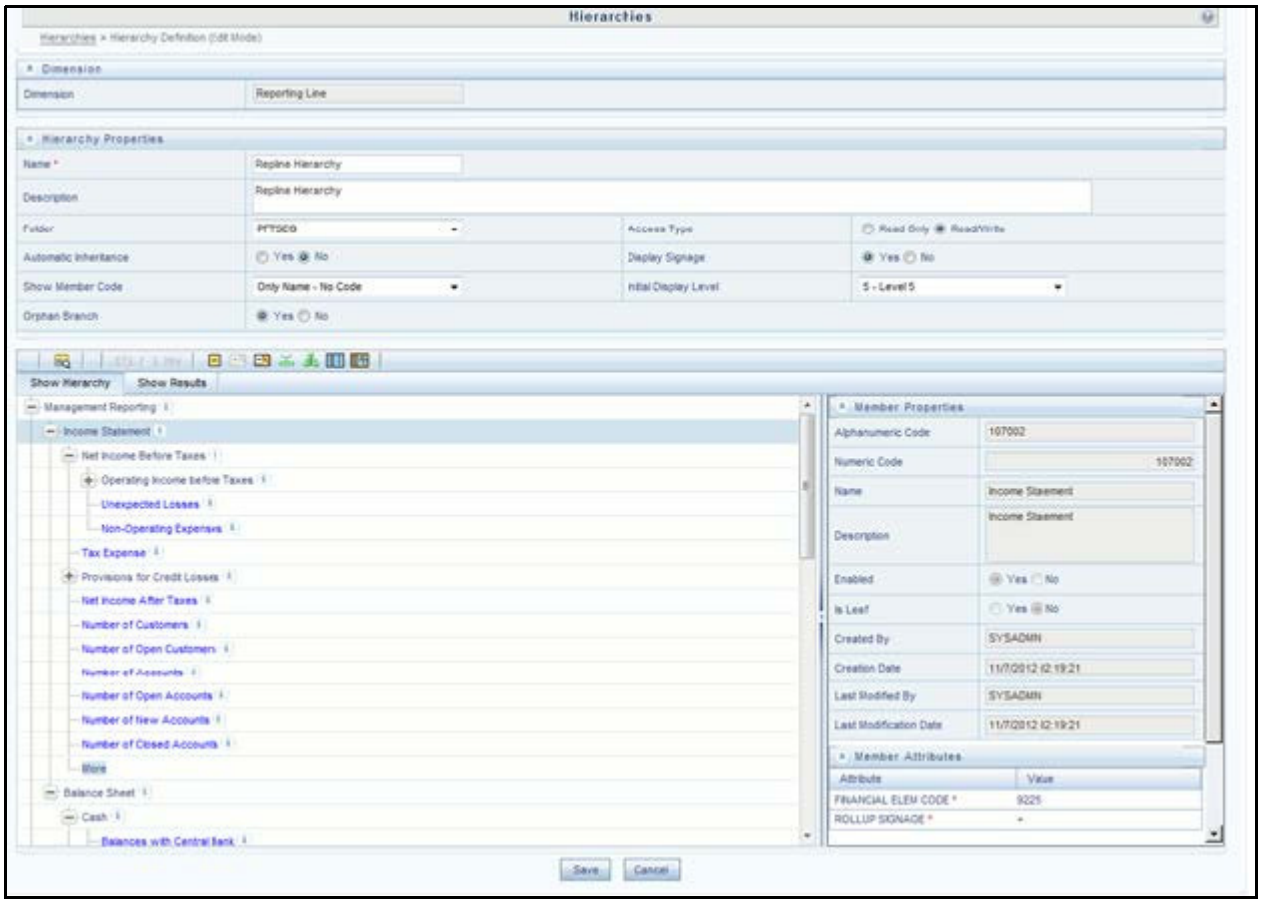


Figure 52. Hierarchies

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

Modify the Seeded Business Metadata

1. Resave the Seeded Business Metadata parent child hierarchy, "Reporting Line Hierarchy" (HPFTRL), so that the changes done are consolidated in the hierarchy as well.

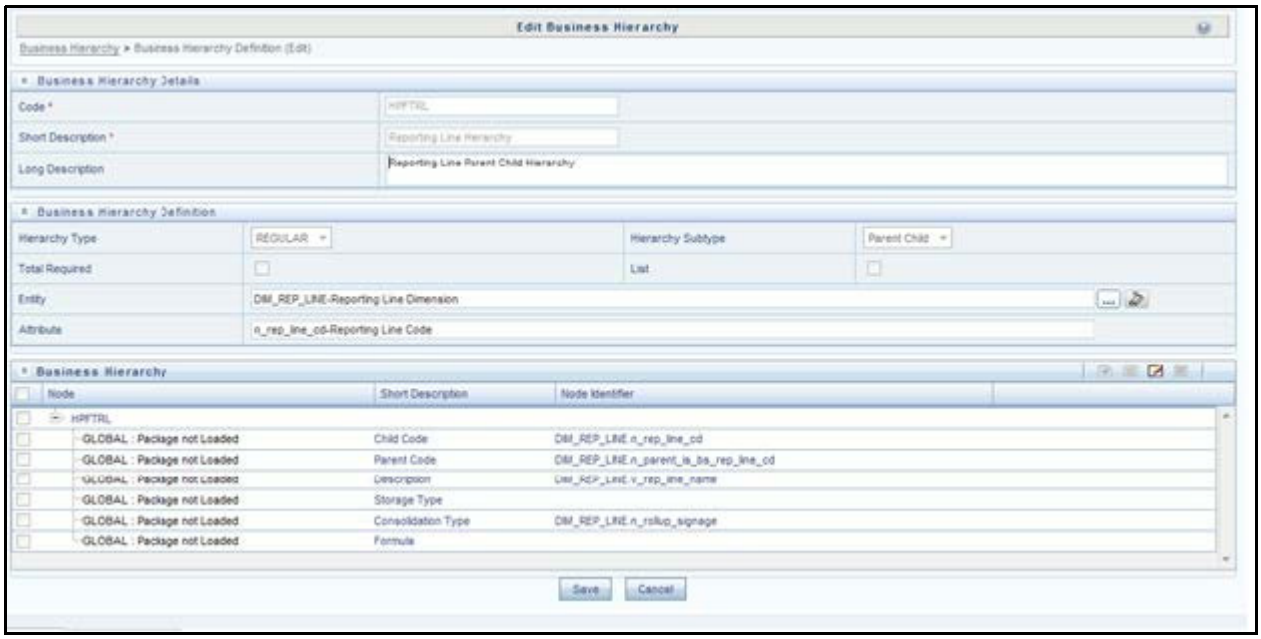


Figure 53. Business Hierarchy

2. (Optional) Create the Business Measures for the newly added reporting lines.
3. Attach and Save the defined Business Measures to the hierarchy "Reporting Line Measures".
4. Save the metadata.

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

Map Maintenance

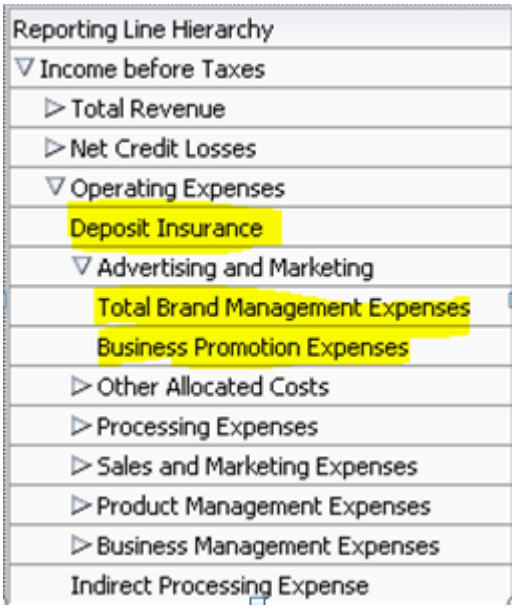
Once all the above steps are done, the seeded map configured between the Reporting Line Hierarchy and Reporting Line Measure Hierarchy has to be modified if required from Map Maintenance of OFSAAI.

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

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 a subtraction to the corresponding Parent Reporting Line. The reporting line values that are loaded to the Fact tables like FCT_ACCOUNT_PROFITABILITY or FCT_MGMT_REPORTING are leaf level reporting lines.

For example, consider the following hierarchy:



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 values will be expected as positive, and some will be expected as negative. For example:

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 rolls up into Operating Expenses, it is considered a subtraction. This rollup into the immediate parent is called Rollup Signage.

However, when rolling up further, (in this case, Income before Taxes), the signage of Deposit Insurance will be 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 rolls up into Income before Taxes,

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

Hence, 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.

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.

Prerequisites

Following are the lists of tables used in the population of Fact Account Profitability and these tables are required to be loaded prior to running the DT.

- DIM_DATES - Mandatory
- DIM_REP_LINE - Mandatory
- FCT_COMMON_ACCOUNT_SUMMARY
- FCT_PFT_ACCOUNT_SUMMARY
- FCT_FTP_ACCOUNT_SUMMARY
- FCT_REG_CAP_ACCOUNT_SUMMARY
- FCT_ECO_CAP_ACCOUNT_SUMMARY

For more information on SCDs, refer to Chapter 3, "Dimension Loading Process,".

Executing the Fact Account Profitability Population DT

To execute the DT component from OFSAAI ICC framework (accessed through *Operations* module), a seeded batch, **<Infodom>_Pop_Account_Profitability** has to be executed for the required MIS Date.

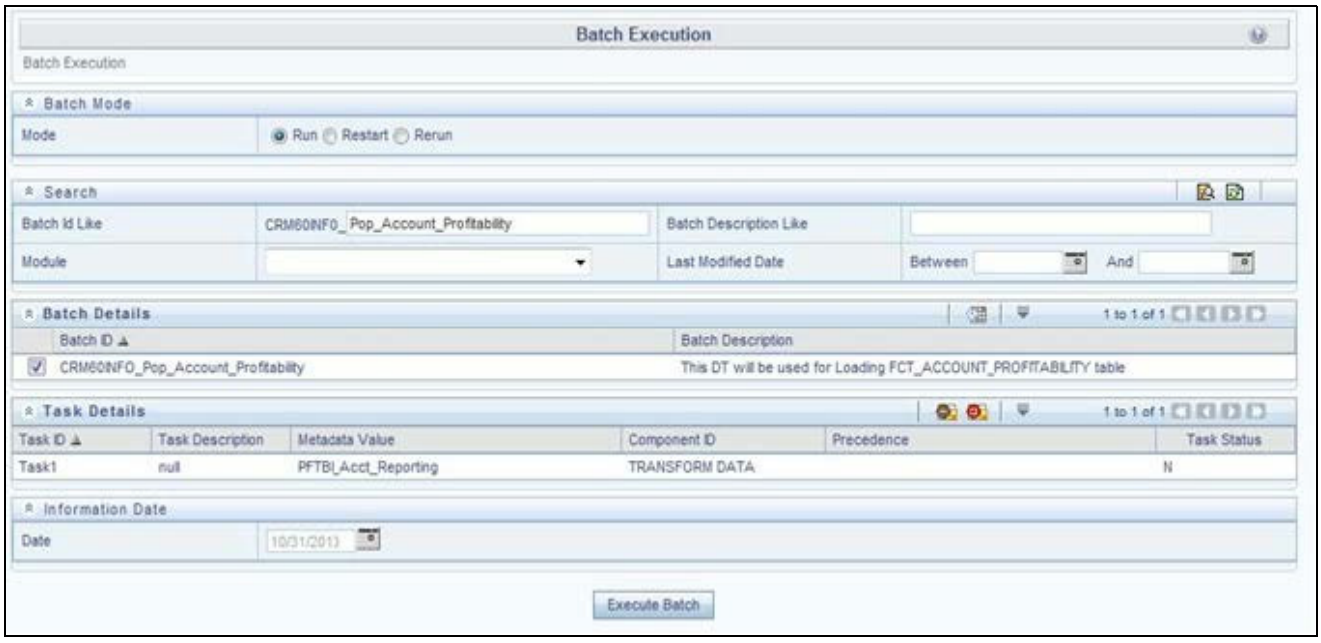


Figure 54. Execute Fact Account Profitability Population

Alternatively, you can create a new Task for an existing Batch from the *Batch Maintenance* screen, as mentioned below:

1. Select the check box adjacent to a Batch Name in the *Batch Maintenance* screen.
2. Click **Add** (+) button from the *Task Details* grid.

   The *Task Definition* screen is displayed.
3. Enter the **Task ID** and **Description**.
4. Select the **TRANSFORM DATA** component from the **Components** drop down list.
5. In the Dynamic Parameters List, select the appropriate **Datastore Type** from the drop down list.
6. Select the appropriate **Datastore Name** from the drop down list. Usually it is the Information Domain name.
7. Select the **IP Address** from the drop-down list.
8. Select the Rule Name **FCT_ACCT_TRANSFORMATION** from the drop down list.
9. Enter the Parameter List details as mentioned below:
   - Reload Account Profitability table for the given MIS Date flag – can be Y or N within single quotes.
   - Reporting Currency code – This has to be enclosed within single quotes.

   For Example, if reporting currency is in US Dollar, then **'USD'** has to be specified.

Note: Batch run ID and **As Of Date** are passed internally by the batch to the Data Transformation task.

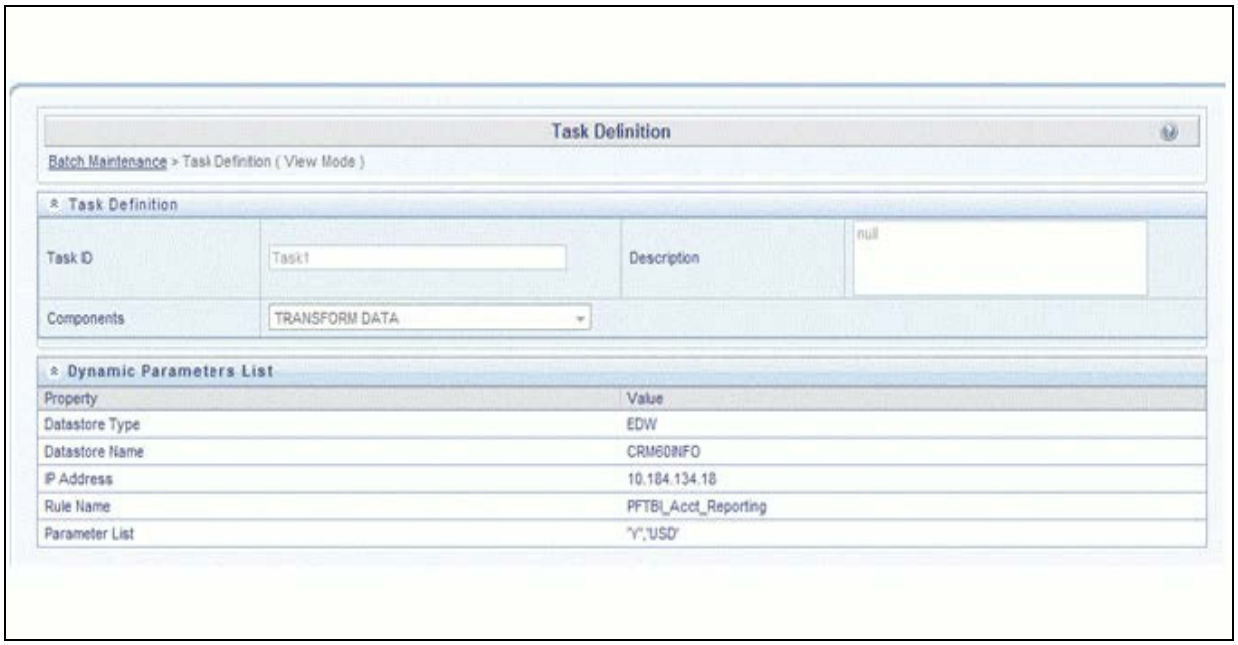


Figure 55. Task Definition

10. Execute the batch for which the Task has been created.

    Note: The batches "<INFODOM>_POP_ACCOUNT_PROFITABILITY" and "<INFODOM>_aCRM_CRM_ACC_SUMM" populate a row with "Run skey & Reporting Currency Code" combo into the table RUN_EXE_PARAMETERS.

    If the user wants to run both the batches or if the user wants to re-execute one of these batches for the same "Run skey & Reporting Currency Code" combo, then the previous entry made in the table RUN_EXE_PARAMETERS have to removed manually before executing the batch for this value combo. Failing to do this will lead to the error while executing the batch.

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

Checking the Execution Status

The status of batch execution can be monitored from the *Batch Monitor* screen.

Note: For a more comprehensive coverage of configuration and execution of a batch, refer to *Operations* chapter in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

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 following directory:

`$FIC_DB_HOME/log/date`. The file name will have the batch execution id.

Executing the Seeded Run Rule Framework

The CRM account summary T2Ts and the Fact Account Profitability DTs are now compatible with the OFSAAI Run Rule Framework. On executing these items from the RRF, the summary tables will be automatically populated with new Run Skey values. This section helps with brief information on executing the seeded RRF process, to populate the CRM account summary and Fact Account Profitability tables.

The CRM account summary T2Ts and the Fact Account Profitability DTs are packaged with the conventional ICC batches as well as with OFSAAI Run Rule Framework. It is recommended to use the OFSAAI Run Rule Framework to execute these items.

Consider the following points before deciding the execution path.

- On executing these items through the Run Rule Framework, the run_skey value is automatically generated by the system and the same is populated in FCT_CRM_ACCOUNT_SUMMARY and FCT_ACCOUNT_PROFITABILITY tables.
- If the items are to be executed through ICC batch:
  - The user have to manually pass the run_skey value to be used while populating the records.
  - If the tables FCT_CRM_ACCOUNT_SUMMARY and FCT_ACCOUNT_PROFITABILITY already have the records for the run_skey being passed, the user have to manually delete these records from the tables before executing.
- Consider executing these items through ICC batch only if a repopulation for the same run_skey is to be performed.
- For a fresh run, it is always advised to use the Run Rule Framework.

1. Select the seeded process by name "CRMAS_ACCT_PFTY" available in the *Process* screen.

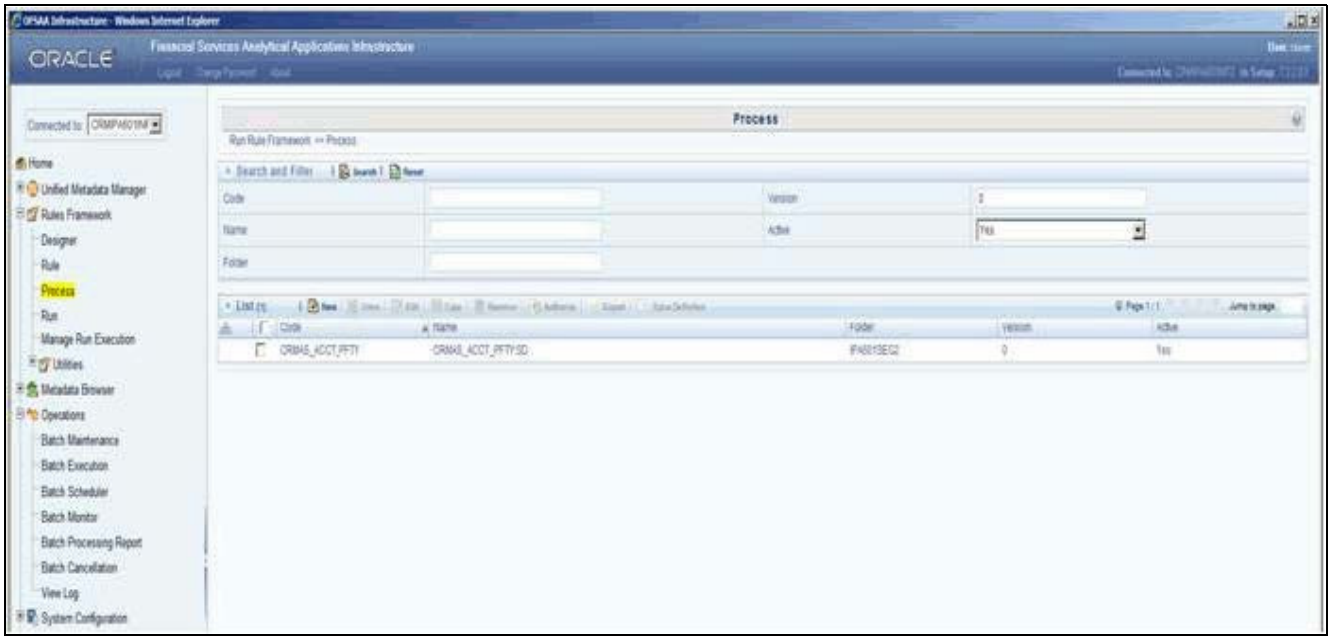
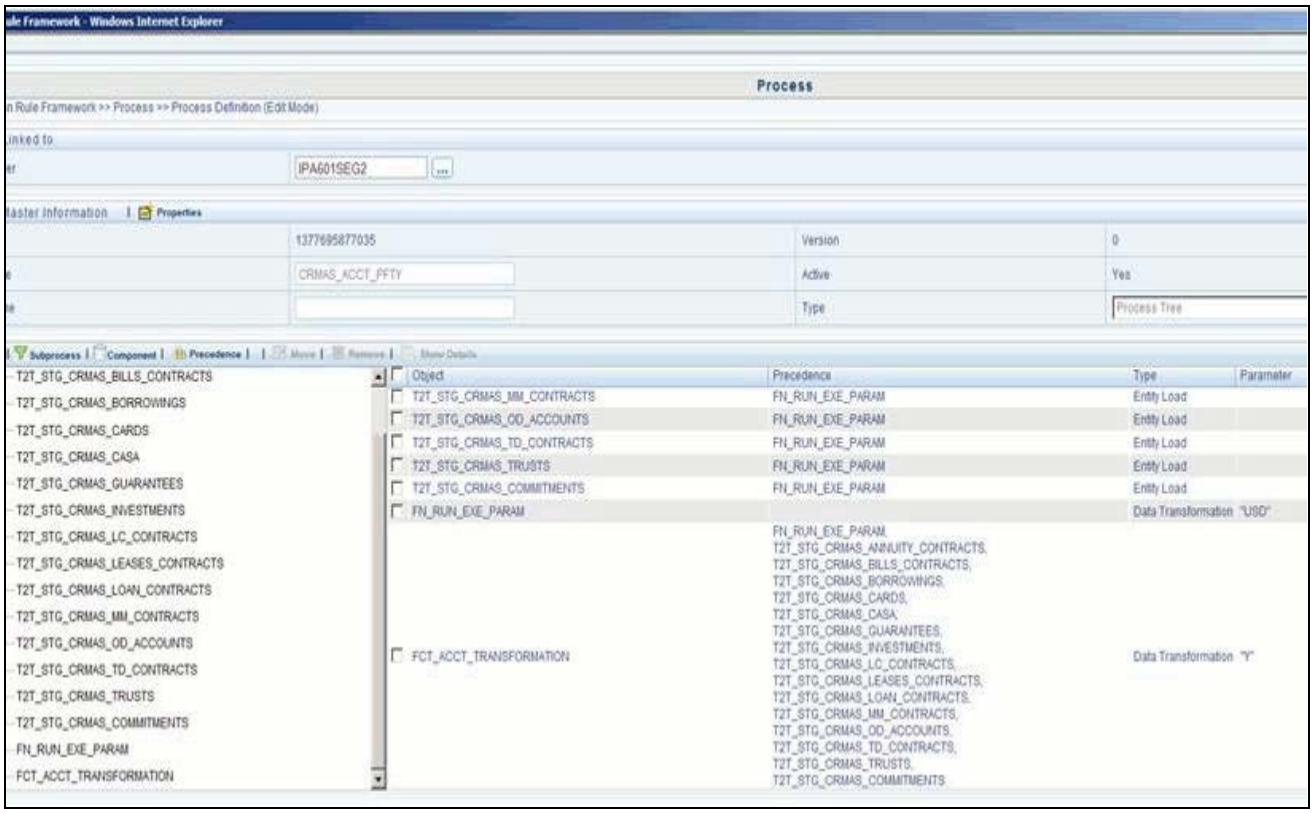
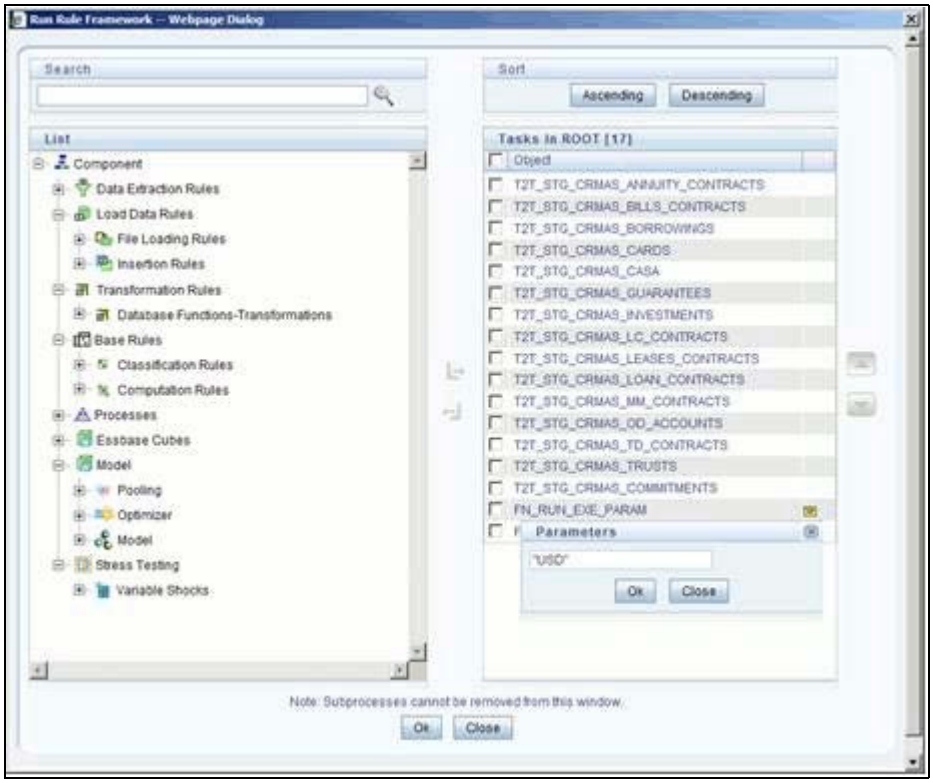


Figure 56. Seeded Run Rule Framework

2. Edit the process and click the "component" option.



3. From the list of tasks available in the right pane, click the arrow present near the " FN_RUN_EXE_PARAM" task.
4. Feed in the currency code of the Reporting Currency.



5. From the list of tasks available in the right pane, click the arrow present near the task by name "FCT_ACCT_TRANSFORMATION".
6. Feed the values for the below parameters as comma separated values enclosed individually in double quotes.
   - Re Run Flag
   - Regulator Capital flag (optional)
   - Economic Capital flag (optional)
7. Save the Process.
8. Select the seeded "Run" by name "CRMAS_ACCT_PFTY_RUN" and click **Fire Run**.
9. In the batch execution tab , select "Create & Execute" option from the **Batch** menu.
10. Select the desired MIS Date from the calendar and click **OK**.
11. The execution log can be accessed on the application server in the following directory: `$FIC_DB_HOME/log/date & $FIC_DB_HOME/log/t2t`. The file name will have the batch execution id.

    Note: For more information on configuration and execution of a Run rule, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

CHAPTER 12

Cube Build Process

This chapter discusses the following topics:

Introduction

Reports of OFSIPA application 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 about the cube and for building the cubes. Cubes are optional source of data for Institutional Performance application.

The chapter contains the following sections:

- List of cubes seeded within the application
- Process for building cubes

Overview of Cubes

OFSIPA application has the following seeded cubes:

- Institutional Analysis
  - Purpose

    The purpose of this cube is to provide analysis of various Account related measures across dimensions like Product, Line of Business, Vintage, and so on.

  - Dataset

    This cube is based on the FCT_COMMON_ACCOUNT_SUMMARY, FCT_CRM_ACCOUNT_SUMMARY, FCT_COMMON_CUSTOMER_SUMMARY, and FCT_CRM_CUSTOMER_SUMMARY fact tables.
- RM P and L Cube
  - Purpose

The purpose of this cube is to provide details of Profit and Loss statement of a Relationship Manager across dimensions like Line of Business, Product, Organizational Unit, and so on.

- Dataset

  This cube is based on the FCT_ACCOUNT_PROFITAIBILTY and FCT_ACCOUNT_MGR_REL fact tables.

In case there is an error with the Relationship Manager cube saving or execution for the first time after the installation the parent child hierarchies must be saved by editing the hierarchies individually and re-saving them after which the cube needs to be saved successfully and re-executed.

Creating Configuration Files

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. Configuration files for seeded cubes are available as part of the installer. However, if there are any changes to cube definition then configuration files are recreated during saving of the cube definition.

Follow these steps:

1. On the LHS menu of OFSAAI, go to **Home > Unified Metadata Manager > Business Metadata Management > Cubes**.
2. Click Search and check if you can see the cubes in the pop up window that opens.
3. Click on the cube that needs to be built and click **OK** to return to the Cube Definition Screen.
4. Click **Save** to save the cube. A pop up appears saying 'Operation Successful'.

   Note: Cube definition will be 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

The Cube build process in OFSAA Infrastructure contains the following steps:

1. Generating 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.
2. Creating the cube outline on Essbase server. This is performed by the **CREATE CUBE** component task within the batch definition.
3. Loading the data to the cube. This is performed by the **CREATE CUBE** task within the batch definition

This section covers the following topics:

- Prerequisites
- Tables used by the Cube build component
- Executing the Cube build task

- Checking the execution status

Prerequisites

Following are the prerequisites for creating a cube:

1. All the post install steps mentioned in the OFSAA Infrastructure installation guide and Solution installation manual have been completed successfully.
2. Parentage files need to be created for BI hierarchies after dimension data is loaded. 'Resave Metadata' process is used to create the parentage files.
3. OFSAAI application user needs to have the required functions mapped to the user for doing Resave Metadata and accessing the Home> Unified Metadata Manager > Business Metadata Management screens and executing a batch from Application batch operations screen
4. Execute Save Metadata by navigating to the following screen on the OFSAAI framework LHS Menu.
5. Go to **Home>Administration>Save Metadata**.
6. Choose all the available metadata under Hierarchy and move it to the right by using the '>>' button.
7. Click **Save** and might take a few minutes for the saving to complete.
8. Click **Show Details** to view the log for the Save operation.
9. Ensure that the following services are running on the application server before doing a cube build:
    - Iccserver
    - Router
    - AM
    - Messageserver
    - Olapdataserver
10. Batches need to be created for executing, which is explained in the Executing the Cube build section.
11. All the required tables for dataset need to be populated before you execute the cube batches, such as Dimension Population, Time Dimension population, Account Summary Population and Fact Ledger Population.
12. 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.**
- Click any dataset in the pop up which opens 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 below button to define an expression using the Expression screen.
- Click **OK** to return to the data set screen.

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

14. Frame a 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 filterparts>
```

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

To execute the cube build process from OFSAAI ICC framework (accessed through the application Batch Operations screen), create a new Batch with two tasks – one for performing Data crunching (component is Aggregate Data) operations and another for building cube (component is Build Cube). The above batch needs to be created for each of the cubes.

- Aggregate Data Task

1. From the Home menu, select **Operations** and then select **Batch Maintenance.**
2. Click New Batch ('+' symbol in Batch Name container) and enter the Batch Name and Description.
3. Click **Save.**
4. Select the Batch you created in the earlier step by clicking on the check box in the Batch Name container.
5. Click New Task ('+' symbol in Task Details container).
6. Enter the Task ID and Description.
7. In the Component drop down, choose **Aggregate Data**.
8. Select the following from the Dynamic Parameters List and then click Save:
    - Datastore Type - Select the appropriate datastore from the list.
    - Datastore Name - Select the appropriate 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.

- Create Cube Task

1. In the batch created in Aggregate Data task above, click New Task ('+' symbol in Task Details container).
2. Enter the Task ID and Description.
3. In the Component drop down, choose Create Cube.
4. Select the following from the Dynamic Parameters List and then click Save:

- Datastore Type - Select the appropriate datastore from the list.
- Datastore Name - Select the appropriate 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.

5. Execute the batch created in the above step.

Note: A common issue in the Aggregate task is Data Set not having records for which the steps mentioned in the prerequisites have to followed or the SQL query in Data Cruncher log file has to be checked on the database (Location of log file mentioned in the 'Checking the Execution Status' section below). 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 try appending a string to the Hierarchy member code so that it is unique across the cube or changing the hierarchy data to make the node unique across the cube.

Seeded batches are provided along with the IPA application installer. The following describes the OFSIPA seeded batches:

- Institutional Analysis

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

- RM P and L Cube

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

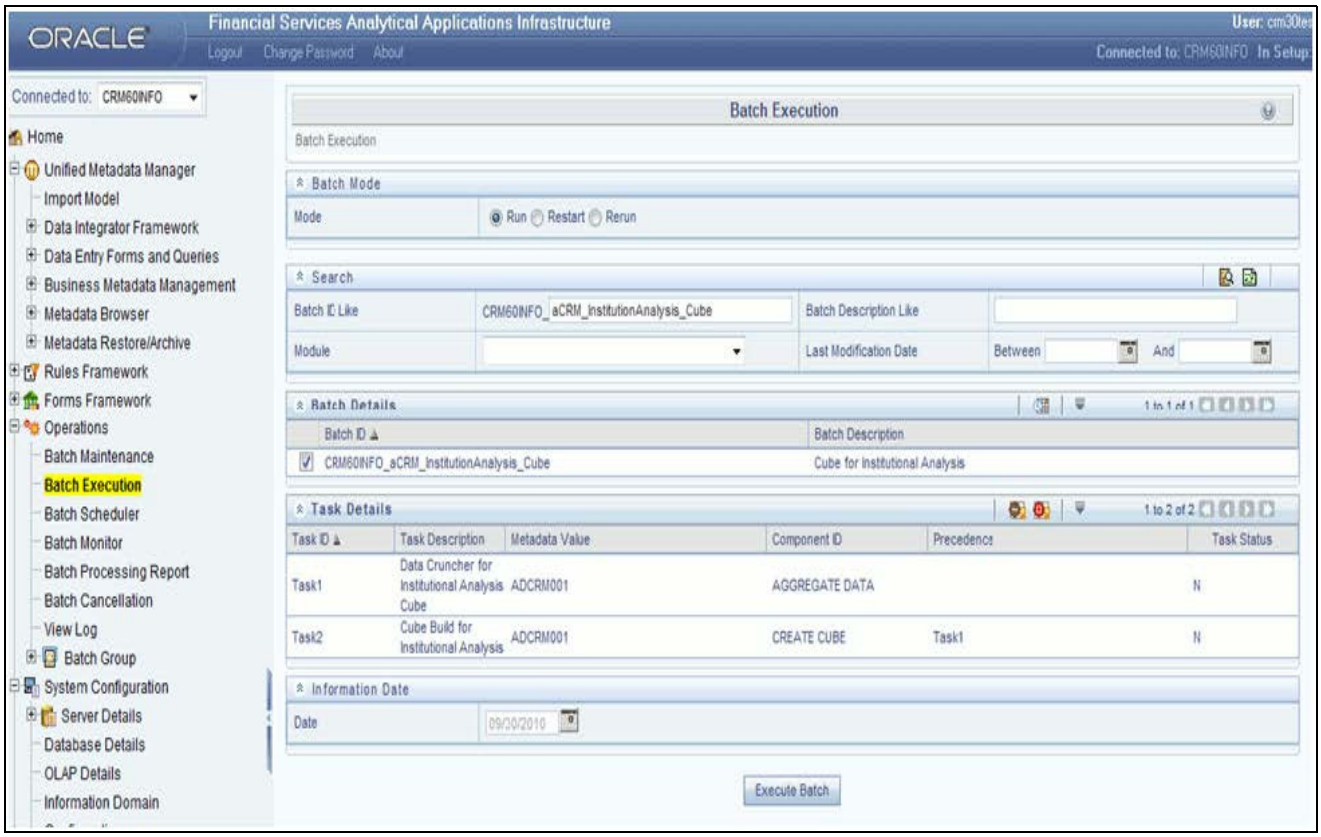


Figure 57. Batch execution

Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen. This you can access by navigating to the following screen on the LHS menu screen: **Home >Operations >Batch Monitor**.

Note: For a more comprehensive coverage of configuration and execution of a batch, refer to *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

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 following 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 How to Develop a New Cube, page C-1 on how to add a New cube or modifying existing ones. For any new cube added using the OFSAAI framework Cube screen , the tasks for execution are the same as mentioned above.

CHAPTER 13

Time Series Forecasting

This chapter discusses the following topics:

Introduction

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.

Note: Projected data is generated through statistical modeling. ARIMA/ARIMAX modeling is used to create the projected data up to a period of 5 years. Historical data for last 2 years is used for creating the projections. The projections is made at an account level. When making the projections for accounts based on the life of the accounts following rules need to be followed:

- 1 to 12 MOB – Use segment information of the account to create projections
- More than 12 MOB – The projections should be solely based on historical data of the account.

Guidelines

Following are the guidelines associated with respect to the execution R code:

- Data should be generated for at least one group for more than 12 continuous mis dates. 12 is the parameter n. Consider 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 is 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, q, and d. 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
  - [http://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
- [http://http://stat.ethz.ch/R-manual/R-patched/library/stats/html/arima.html]

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

Files Used

Two files are required for R script execution. Both the files are present at `$FIC_DB_HOME/bin` folder and need execute permissions. Following are the files used:

- RExec executable
- ARIMA_AVF.r

Errors

Following are the errors:

- Subscript out of bounds usually means that sufficient data has not gone in. Model is trying to apply an algorithm on a dataset that is returning null chunk.
- 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

CHAPTER 14

Segmentation

This chapter discusses the following topics:

Introduction

Segmentation is the procedure of grouping together a set of customers based on certain similar features. These customers grouped together are known to have similar behavior and hence, the future behavior of accounts within a segment can be predicted to follow the similar behavioral patterns as observed for other accounts. Thus, by predicting the behavior of an account, it can be segmented with a set of similar accounts and its future projections can be created. These future projections provide the value of net income that can be expected from an account or customer.

Segmentation is done based on a certain set of dimensions wherein accounts which exhibit a particular combination of dimension members are grouped together. Based on the characteristic around which segmentation is to be created, the dimensions used for segmentation can vary.

Following are the several segmentation types that are being supported:

- Corporate Tracker Segmentation
- Profitability Segmentation
- Risk Based Segmentation
- Behavioral Segmentation

Note: The segmentation models within CI are also used to provide an output to OFS Price Creation and Discovery application (OFS PCD). The segments within CI calculate the average values of profitability components which are then used in PCD to analyze the future behavior of an account belonging to that segment and predict its profitability.

The objective of segmentation is to define a framework that will score accounts at MIS Date and Run level and correspondingly create clusters based on the scores.

Segmentation is done using the following dimensions:

- Year of incorporation
- Status of listing
- Income
- Industry
- Country of incorporation
- Group asset size

Following is the list of Product Types used in IPA segmentation:

- CASA for Current and Saving Accounts
- TD for Term Deposits
- LOAN for Loan Contracts
- CARDS for Credit Cards.

Similar Product Type which is used in Price Creation and Discovery are being referred as CARDS for Credit Cards and Term Deposits.

When Price Creation and Discovery is integrated and is installed with IPA, user is required to update column V_PRODUCT_TYPE in FSI_SEG_REP_LINE_MAP table accordingly to the match the product type used in the Price Creation and Discovery Application.

Table 33. FCT_ACCOUNT_SEGMENT_SCORE

Column Name	Logical Name
N_ACCT_SKEY	Account Key
D_ACCT_START_DATE	Account Start Date
N_ACCT_SEGMENT	Account Segment
N_ACCT_SEGMENT_SCORE	Account Segment Score
N_RUN_SKEY	Run Key
N_MIS_DATE_SKEY	Date Key

The above table will act as the input for another table that stores facts of Account Profitability (Movement, Average Balance, and so on) at the level of Month on Book, Account Segment, Run, and Reporting Line. Following is the structure of this table.

Table 34. FCT_ACCT_SEGMENT_MOB_SUMMARY

Column Name	Logical Name
AVG_BAL_RCY	AccountAverageBalanceReportingCurrency
N_END_BAL	AccountEndingBalance
AVG_BAL	AccountAverageBalance
END_BAL_RC	AccountEndingBalanceReportingCurrency
MOVEMENT	Movement
MOVEMENT_RCY	MovementReportingCurrency
REP_LINE_CD	ReportingLineCode
RUN_SKEY	Run Key
MONTH_ON_BOOK	Month on Book
ACCT_SEGMENT	Account Segment

DIM_SEGMENT_TABLE would be populated using SCD Process (Map reference number 267) and source would be a view DIM_SEGMENT_V for which data would be from DIM_SEGMENT_B/TL and FSI_SEGMENT_TYPE_CD/MLS tables.

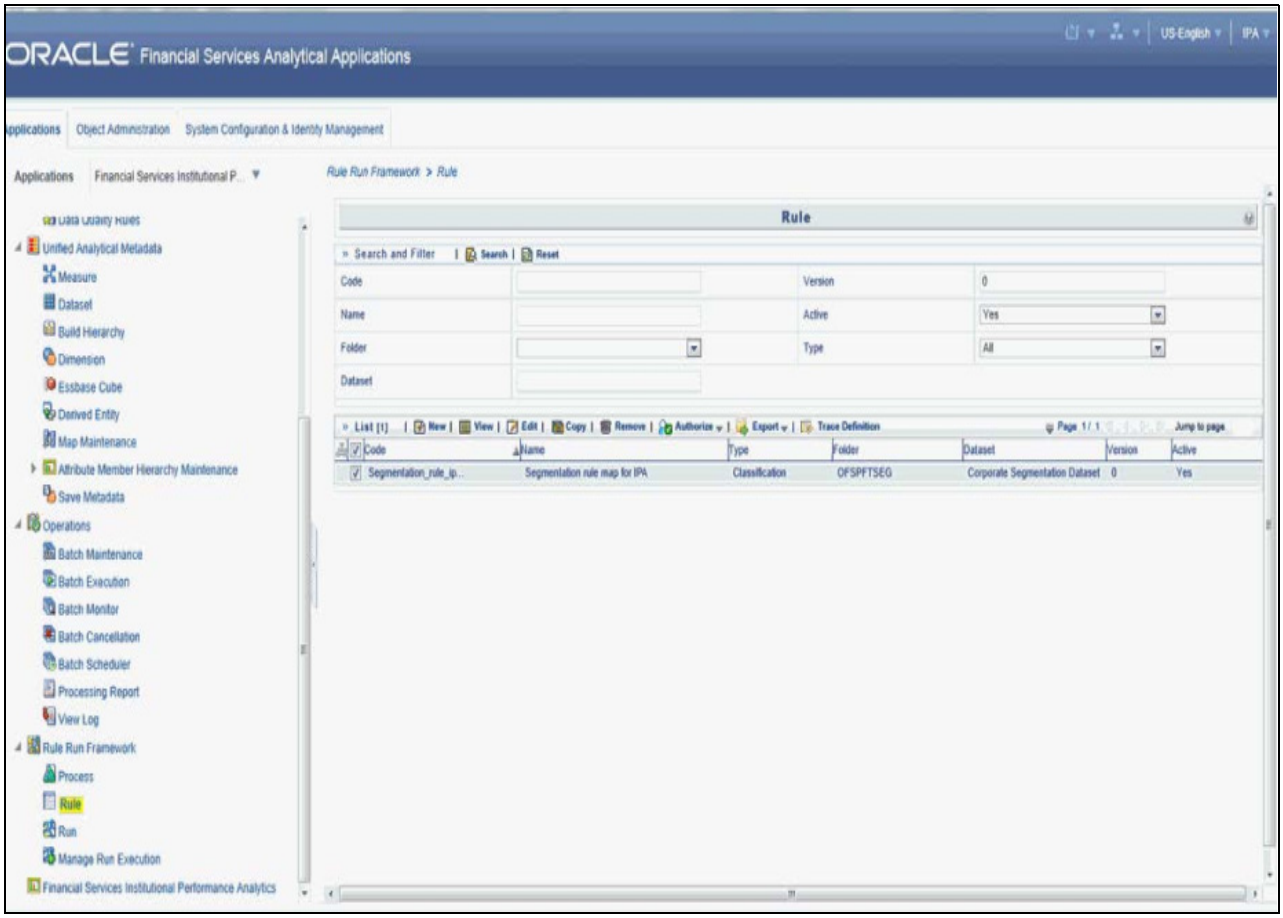
FSI_SEGMENT_TYPE_CD/MLS table stores list of Segment Types used in IPA. For example, Corporate Tracker, Behavioral, Profitability segments, and so on.

The Segment score will not be updated. The rule will update only the n_segment.

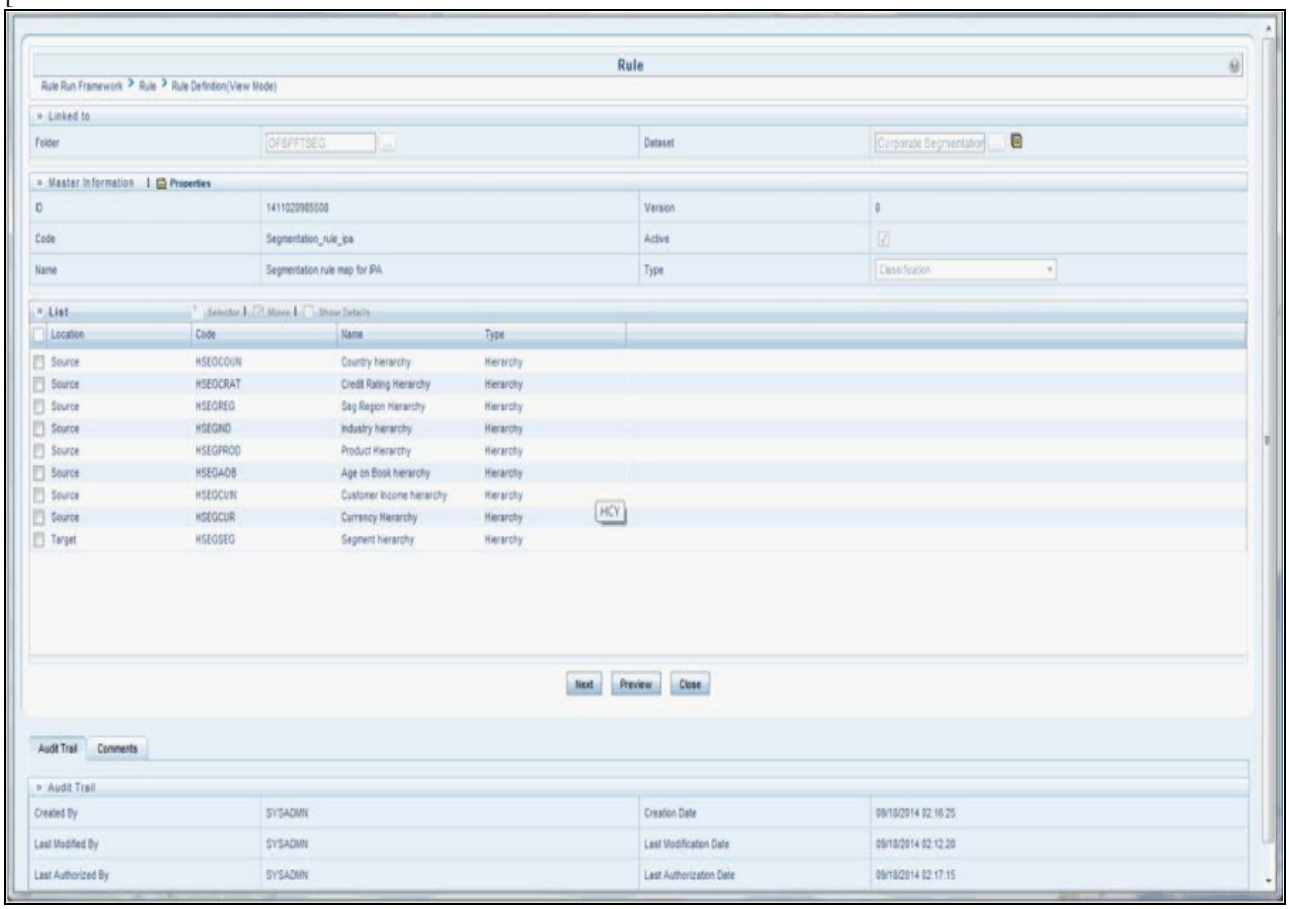
Creating a rule

To define a rule, follow these steps:

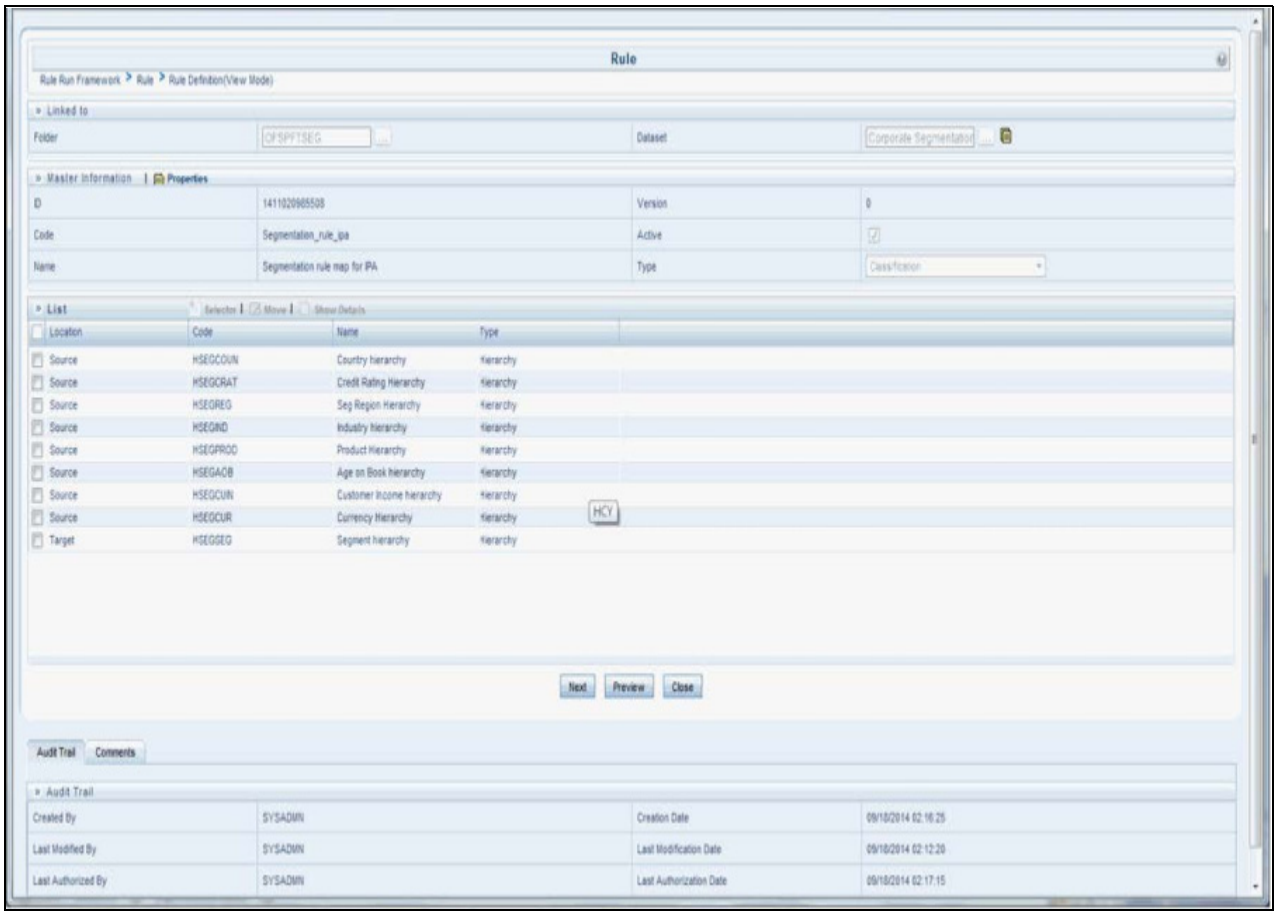
1. Click **Rule** and the following rule appears.



2. Select the rule and click **View**. The following screen appears.



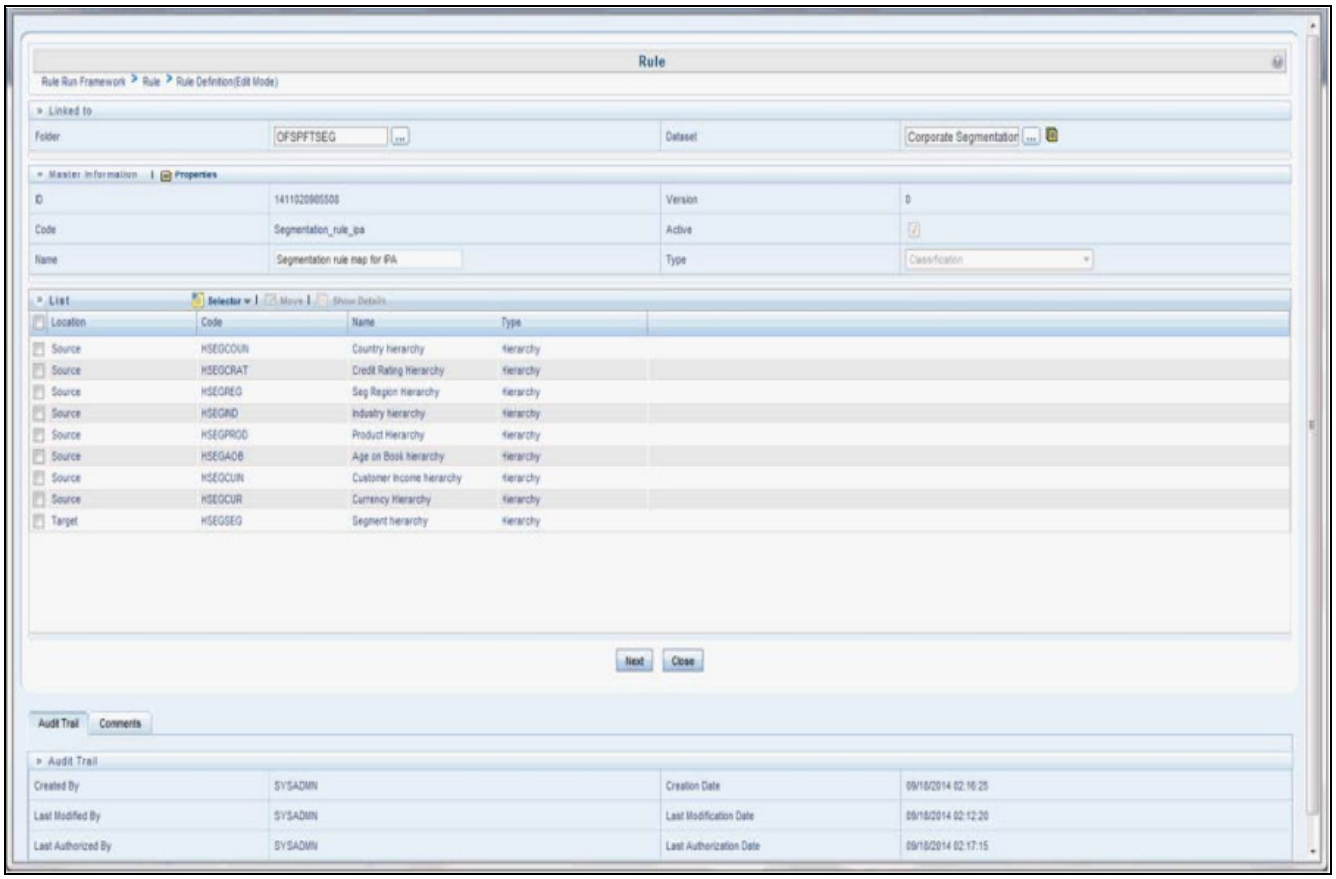
3. On clicking **Next**, the rule defined comes up. For the first time when the rule is not defined, only default seeded node rule should appear as shown in the following screen:



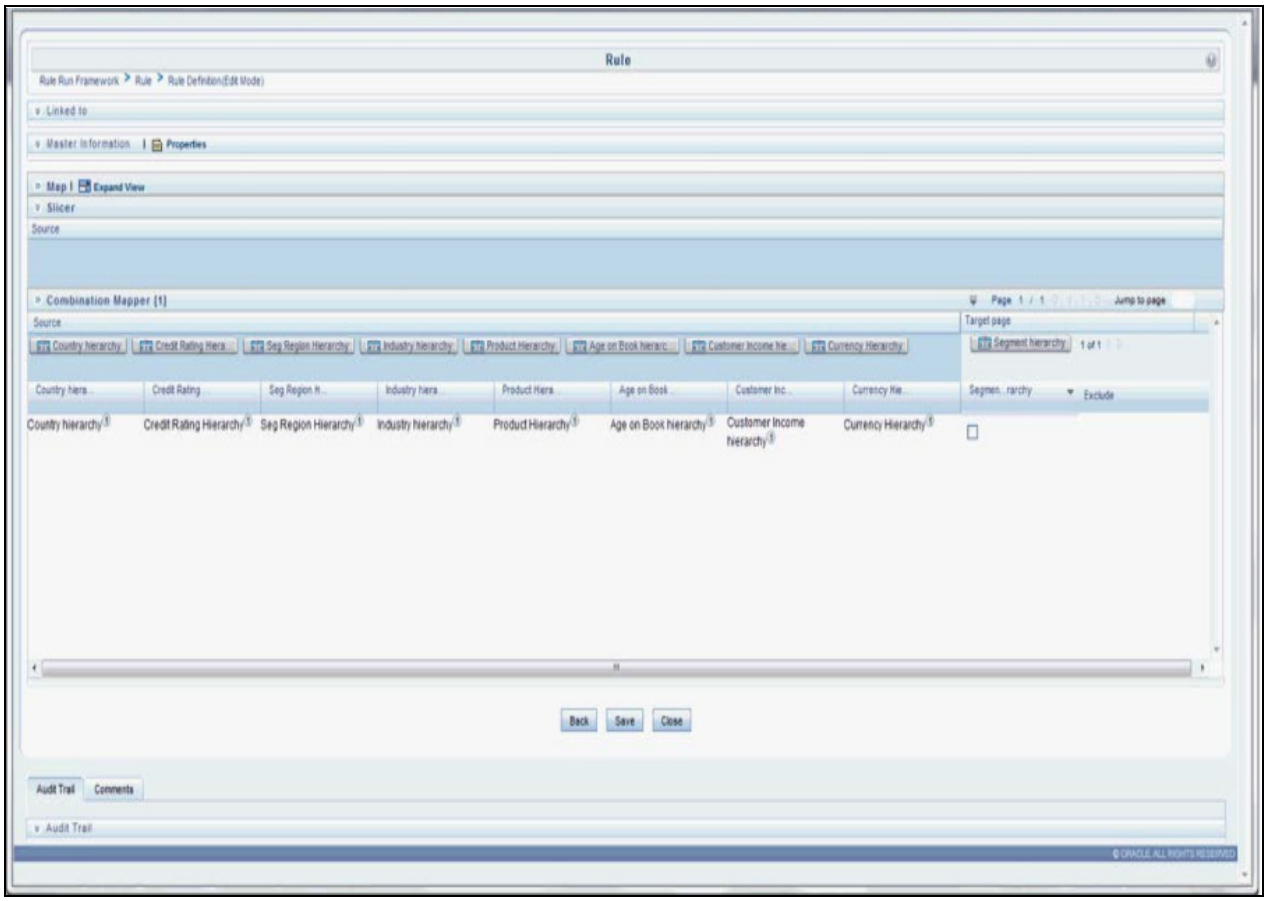
Editing a rule

To edit a rule, follow these steps:

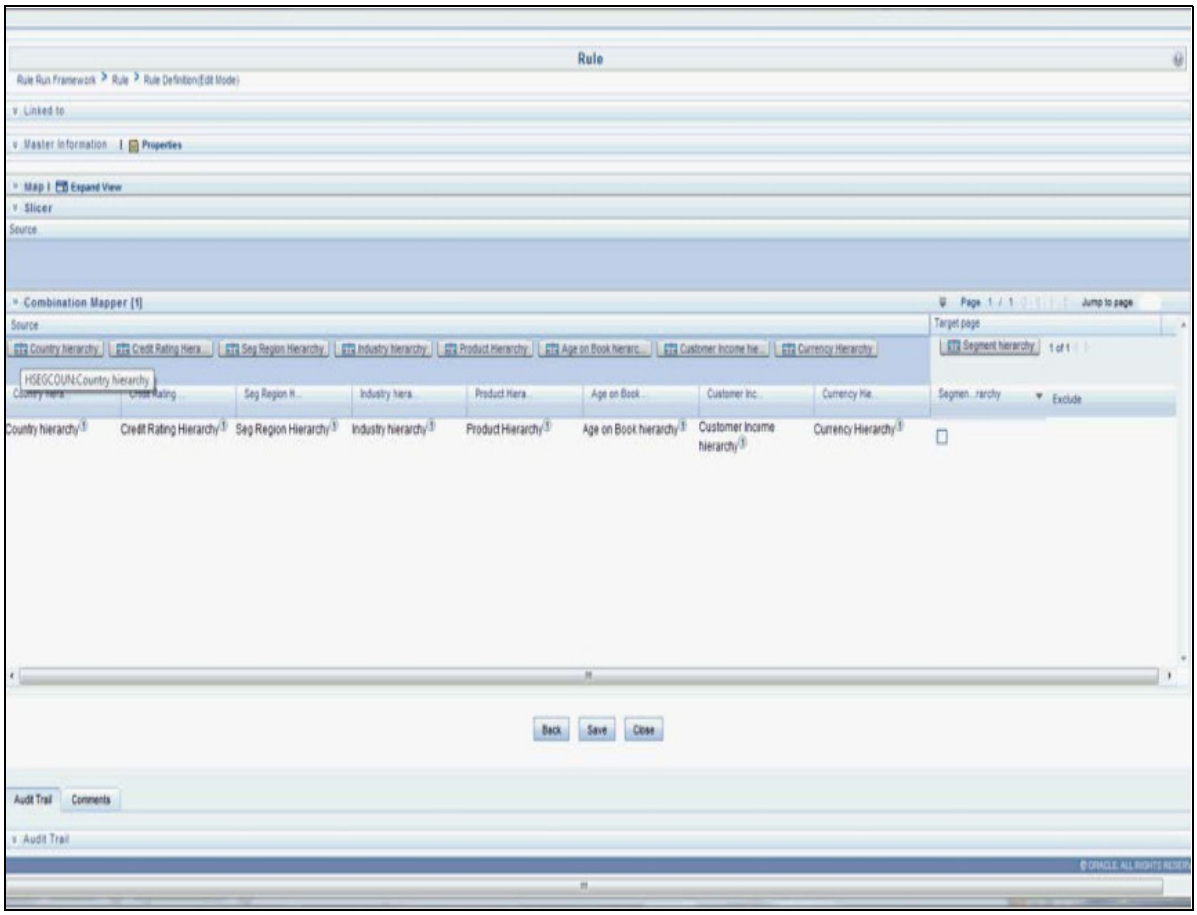
1. Select the rule and click **Edit**. The following screen appears:

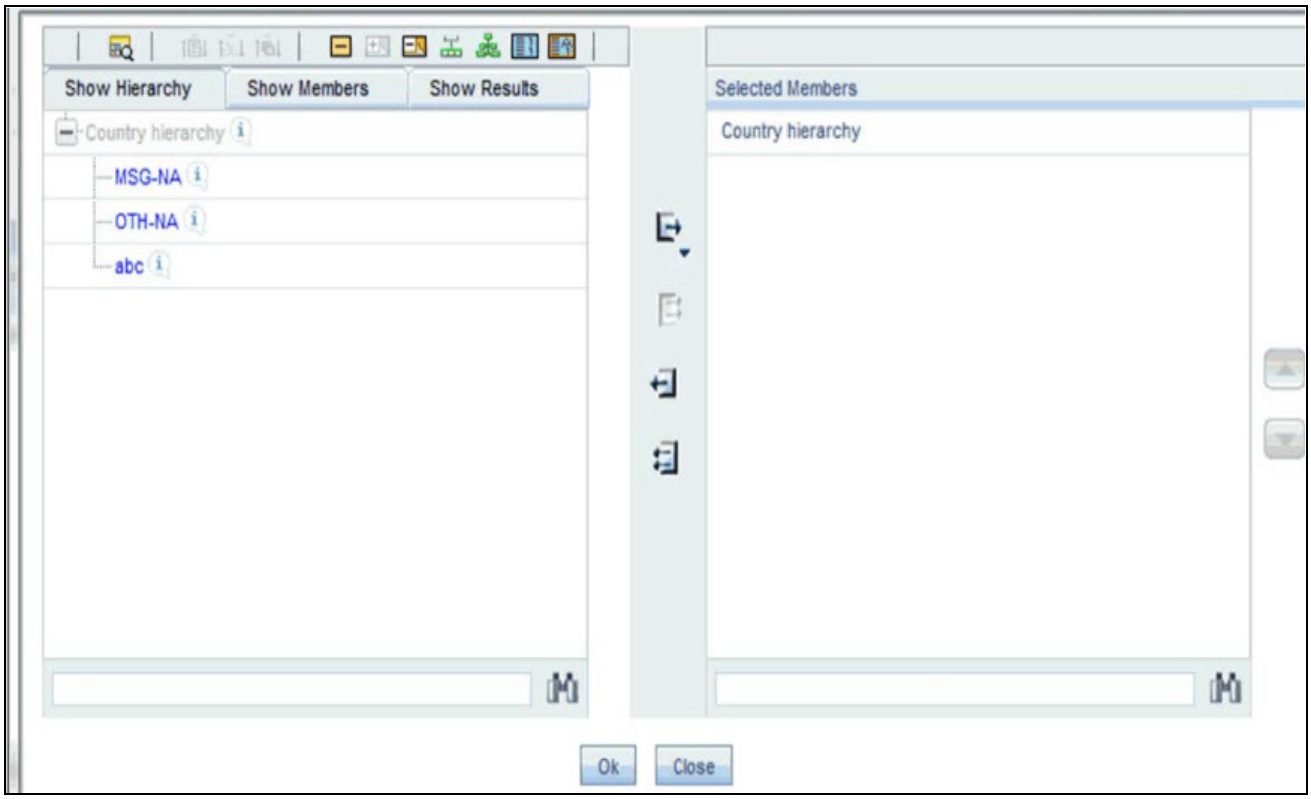


2. Click **Next**. First time default node defined as rule will show up in the following figure:

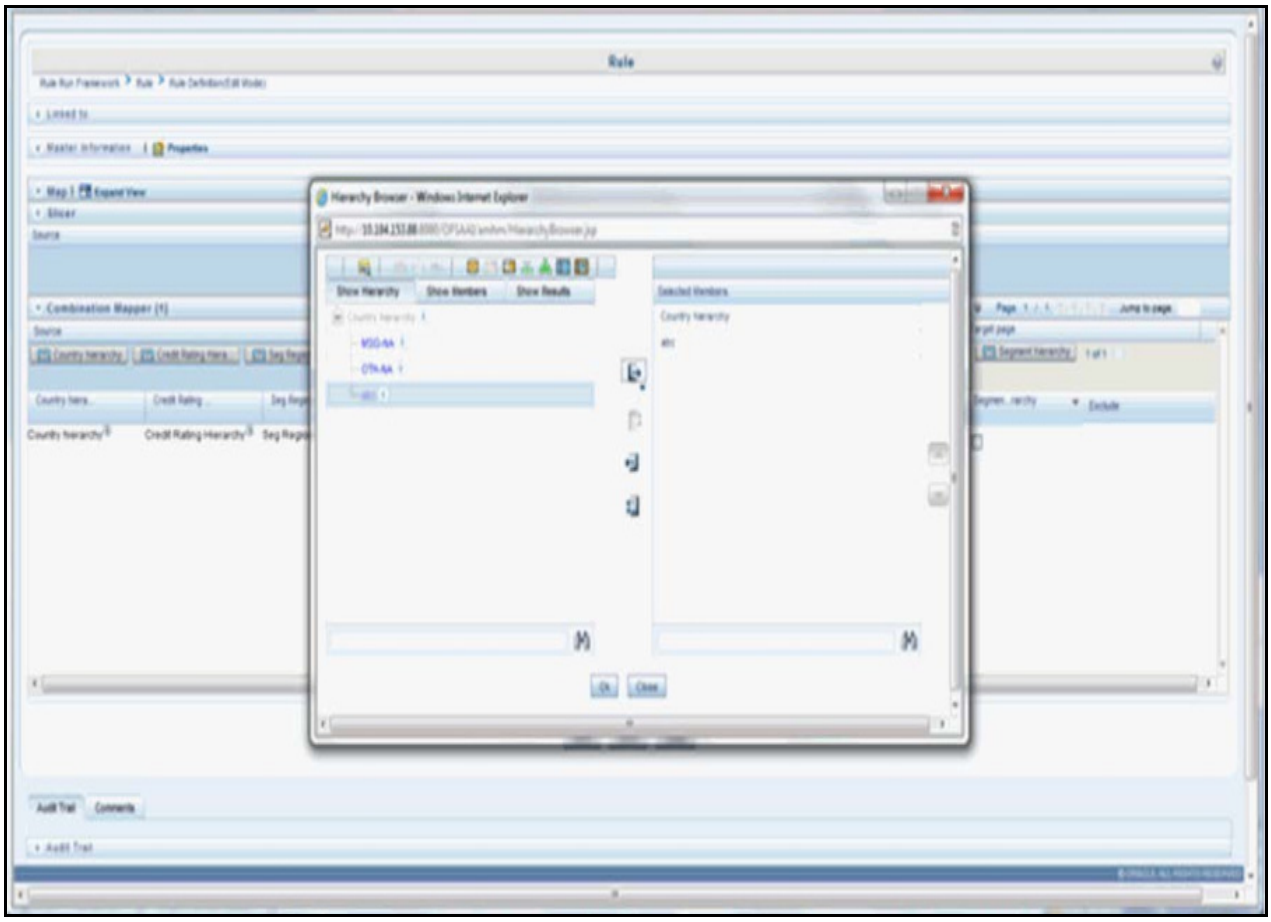


3. Click **Hierarchy** as shown in the following figure and the hierarchy screen opens up.

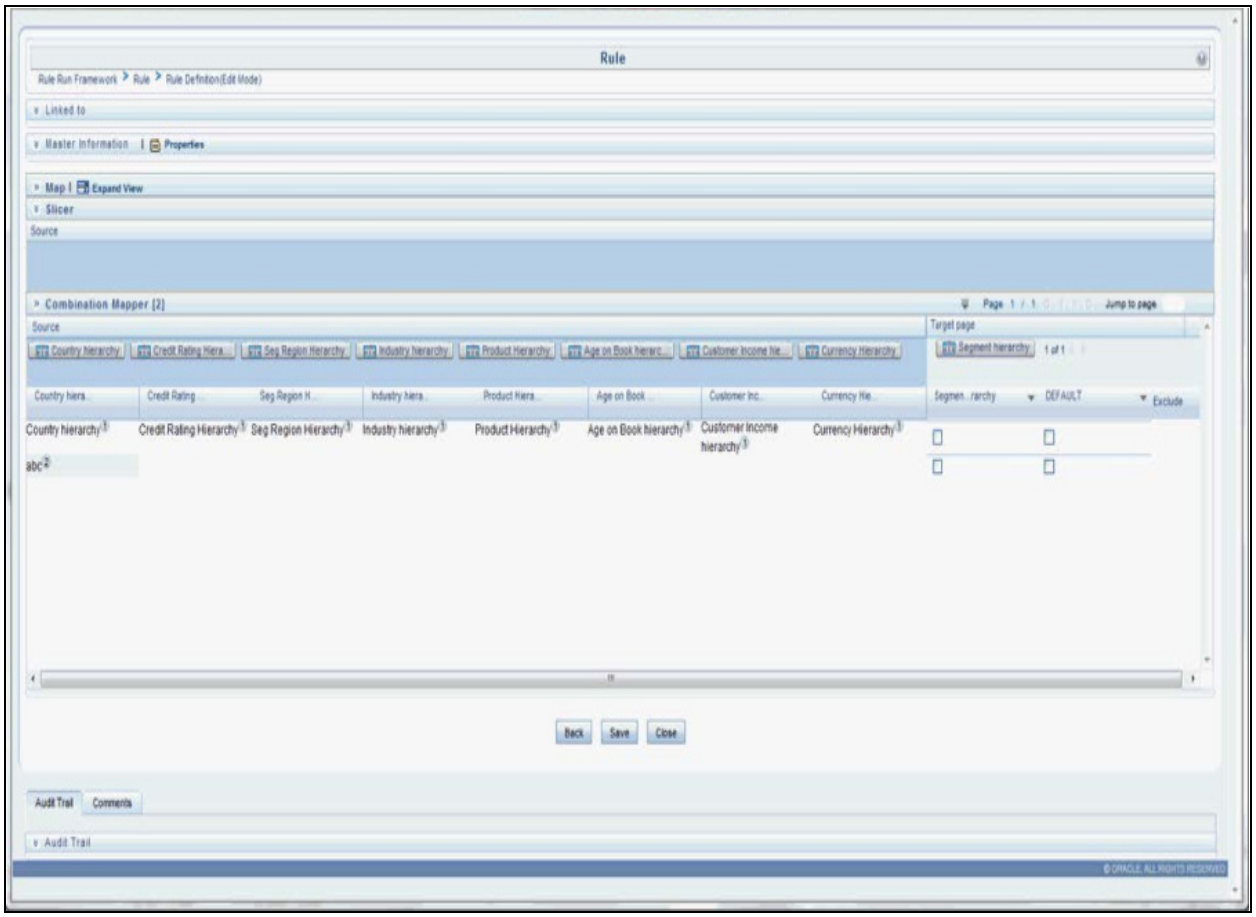

Rule
Rule Run Framework > Rule > Rule Definition(Edit Mode)
Linked to
Master Information | Properties
Map | Expand View
Slicer
Source
Combination Mapper [1]
Source
Country hierarchy
Credit Rating Hiera...
Seg Region Hierarchy
Industry hierarchy
Product Hierarchy
Age on Book hierarc...
Customer Income hie...
Currency Hierarchy
HSEGCOUN:Country hierarchy
Country hierarchy
Credit Rating Hierarchy
Seg Region Hierarchy
Industry hierarchy
Product Hierarchy
Age on Book hierarchy
Customer Income hierarchy
Currency Hierarchy
Target page
Segment hierarchy
1 of 1
Segmen...rarchy
Exclude
Back
Save
Close
Audit Trail
Comments
Audit Trail



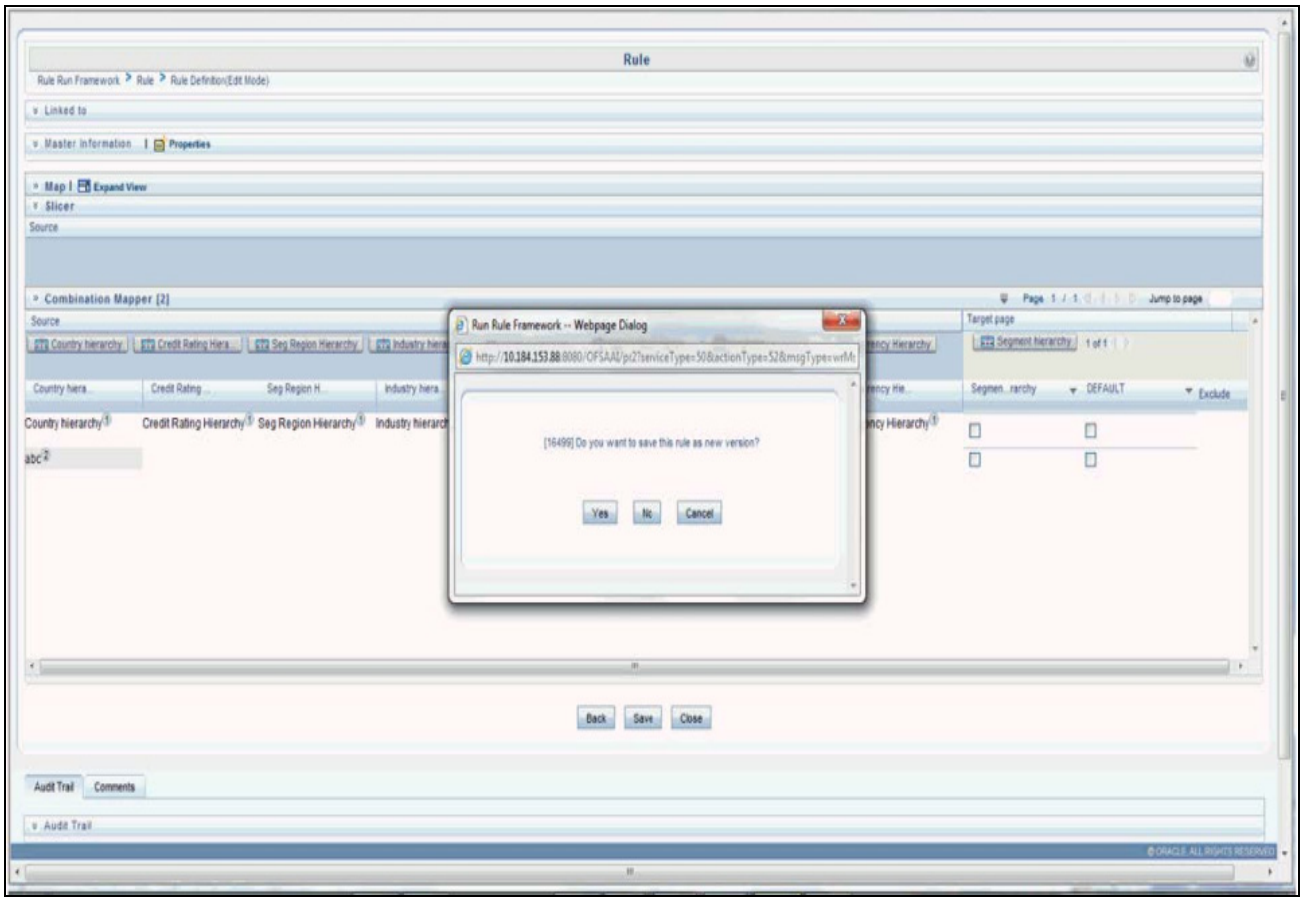
4. Select the hierarchy and click **OK**.



5. The selected node appears in the rule.

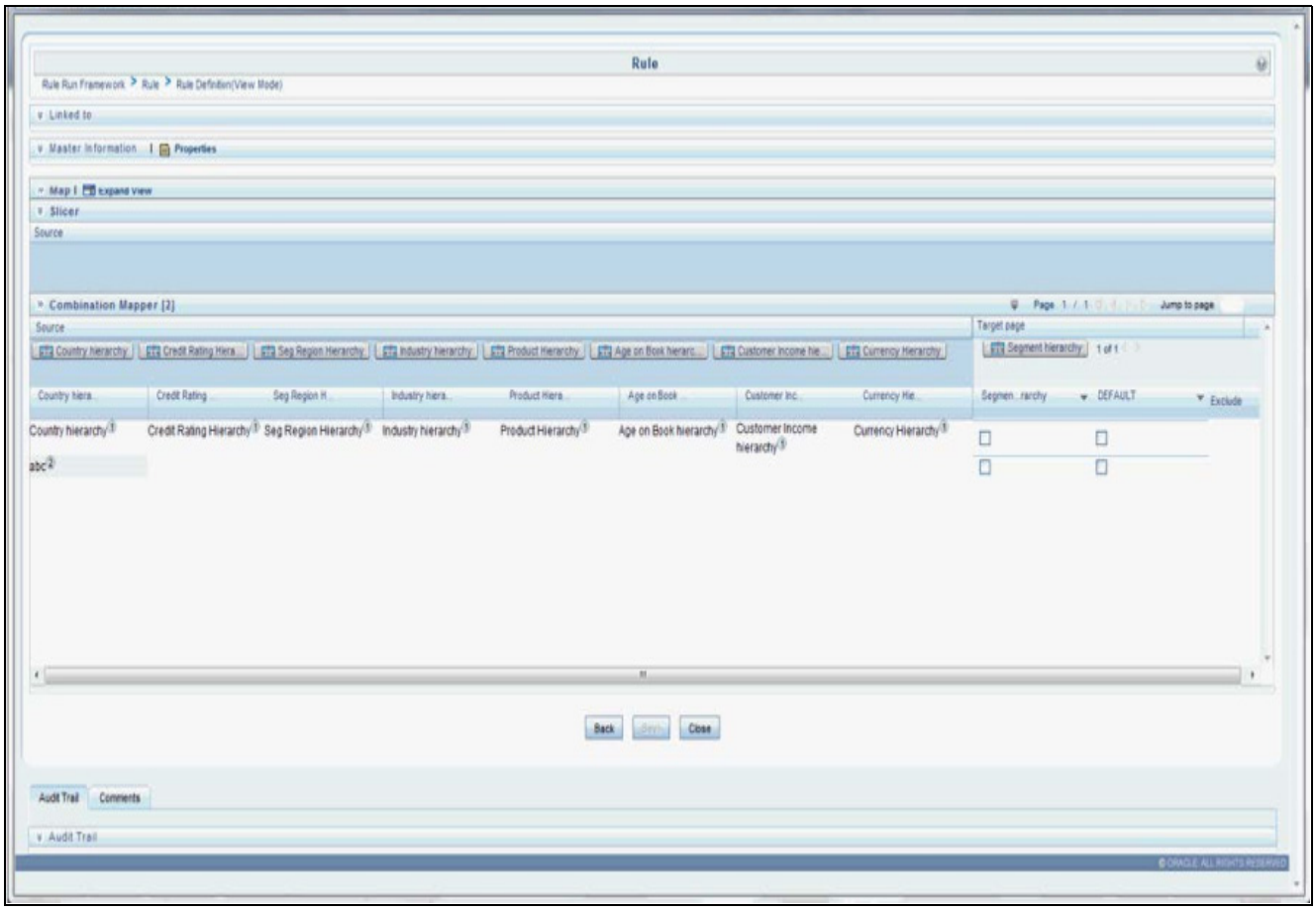


6. Similarly, select all the nodes that need to be considered for the rule and assign it to the target hierarchy. Click **Save**. A confirmation message appears as shown in the following.



7. On clicking **Yes**, the following message appears: *Save action with authorize was successful on following definitions Segmentation_rule_ipa.*

8. Navigate back to the main screen and click the view rule. The rule saved is shown.



CHAPTER 15

Overview of OFSIPA Reports

Oracle Financial Services Institutional Performance Analytics (OFSIPA) offers dashboards to users that organize different kinds of reports by subject area.

These reports present:

- Behavioral and Engagement trends of its target segments - exposures, commitments, line utilization, assets/liabilities, deposits, withdrawals, fees, income, recent transactions and so on.
- Performance of the business and underlying customers.
- Product holdings and across the organization (that is Corporate client and any of its sub-divisions or subsidiaries)
- Efficiency of the sales force in terms of ongoing customer revenue generation, cross-sell and up-sell, product usage and pipeline.
- Efficiency of investments (like marketing, partner development).
- Time hierarchy prompted reports are all drill enabled on time hierarchy. On first load, the values are visible for a year, and on subsequent drills, we obtain values for quarter and month. These are not drill through reports.

Introduction to Dashboards

Oracle Financial Services Institutional Performance Analytics (OFSIPA) offers the following dashboards for various reports:

- Line of Business Analysis Dashboard
- Customer Analysis Dashboard
- Manager Analysis Dashboard
- Product Analysis Dashboard

Line of Business Analysis Dashboard

The Line of Business Analysis dashboard has the following tabs:

- Profit and Loss
- Profit and Loss (Scenario)
- Customer Summary
- Revenue Summary
- Performance Summary
- Top N Summary
- Customer Distribution

- Product Penetration
- Cross Sell Summary

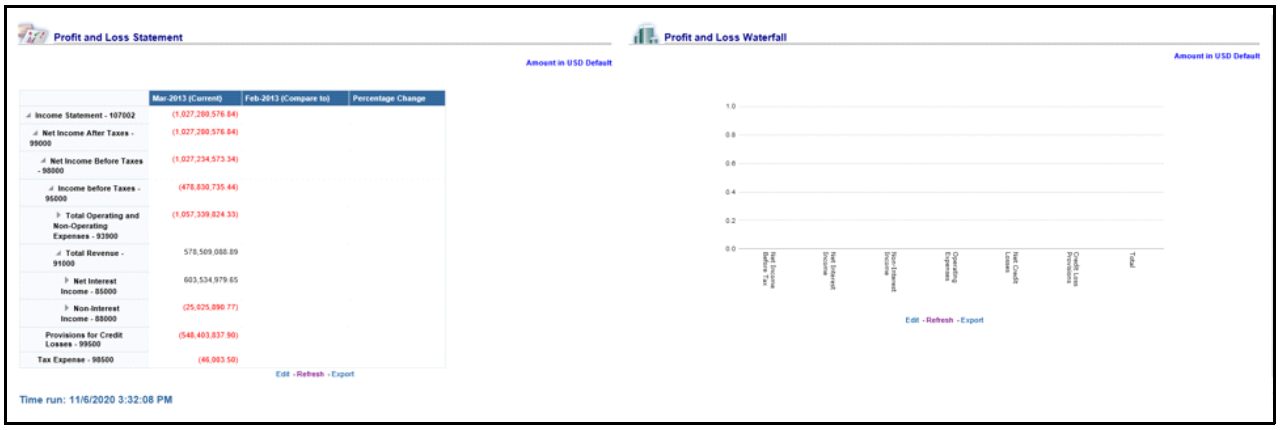
Profit and Loss Tab

The Profit and Loss tab contains two Reports:

- Profit and Loss Statement
- Profit and Loss Summary graph report

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
    - Legal Entity (Drop Down Filter)
    - Line of Business (Drop Down Filter)
    - As of date - Calendar Date Selection
    - Currency (Drop Down Filter)
    - Amount - Denomination (Drop Down Filter)
- Page Level Filters
    - Period level (Radio Button Selection)
    - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)



Profit and Loss (Scenario) Tab

The Profit and Loss (Scenario) tab compares profit and loss actual values with different scenarios like, Plan, Budget, Forecast Values etc, to monitor and track the profit level situations. The comparison can be done between any scenarios.

The report is a tabular one with three columns.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
    - Legal Entity (Drop Down Filter)
    - Line of Business (Drop Down Filter)

- As of date - Calendar Date Selection
- Currency (Drop Down Filter)
- Amount - Denomination (Drop Down Filter)

- Page Level Filters
  - Scenario Selection: Scenario (Selected) Period Compared to Scenario (selected for comparison)- (Drop Down Filter)

Customer Summary Tab

The Customer Summary tab contains the following customers details reports:

- Open Customer Over Time
- Customer Summary by Month

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
- Report Level Filter(s)
  - LOB Filter: For Report Open Customer Over Time- Graphical Report and Customers by Month Graph Report.

Open Customer Over Time

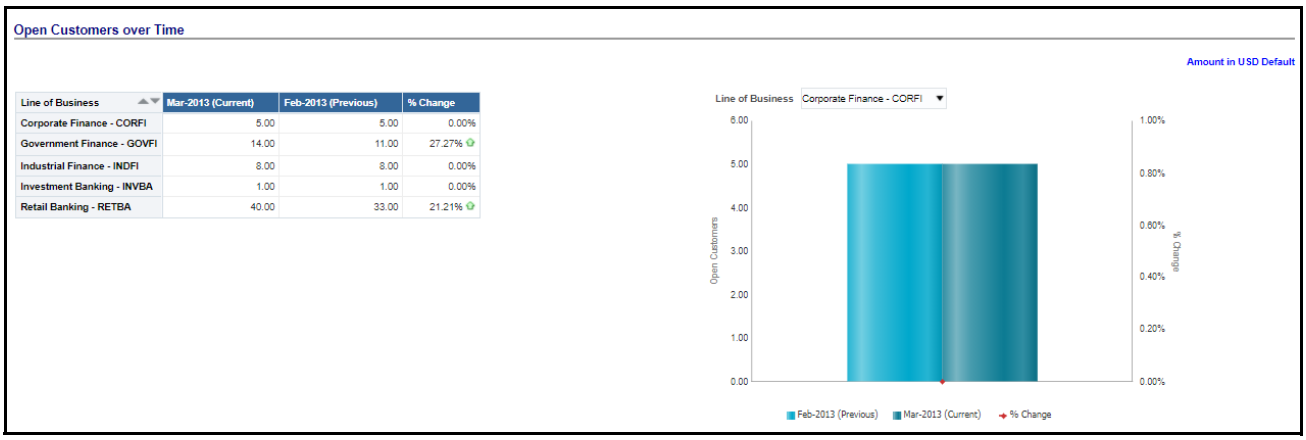
This report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and with percentage Change. The period shown in Table report will be in accordance with the period selected at the "As of Date" filter and Period Level Selection radio button.

For example:

- Selecting Year Radio Button: Would show data for 12 months of the period Selected at "As-Of_Date"
- Selecting Quarter Radio Button: Would show data for the current quarter of the period Selected at "As-Of_Date"
- Selecting Month Radio Button: Would show data for 1 month of the period Selected at "As-Of_Date"

This repost has two parts:

- **Open Customers Over Time- Tabular Report**: This Tabular report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and percentage Change. This report shows for the LOBs selected at Dashboard level LOB filter.
- **Open Customers Over Time -Graph Report**: This report shows average no of open customers by LOB with percentage change through graph. The Graph uses a Report Level LOB Filter where single LOB can be selected to see the trend.



Customers Summary by Month

This report shows average no of open customers, New Customers and Closed Customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and with percentage Change. The period shown in Table report will be in accordance with the period selected at the "As of Date" filter and Period Level Selection radio button.

For example:

- Selecting Year Radio Button - Would show data for 12 months of the period Selected at "As-Of_Date"
- Selecting Quarter Radio Button - Would show data for the current quarter of the period Selected at "As-Of_Date"
- Selecting Mnth Radio Button - Would show data for 1 month of the period Selected at "As-Of_Date"

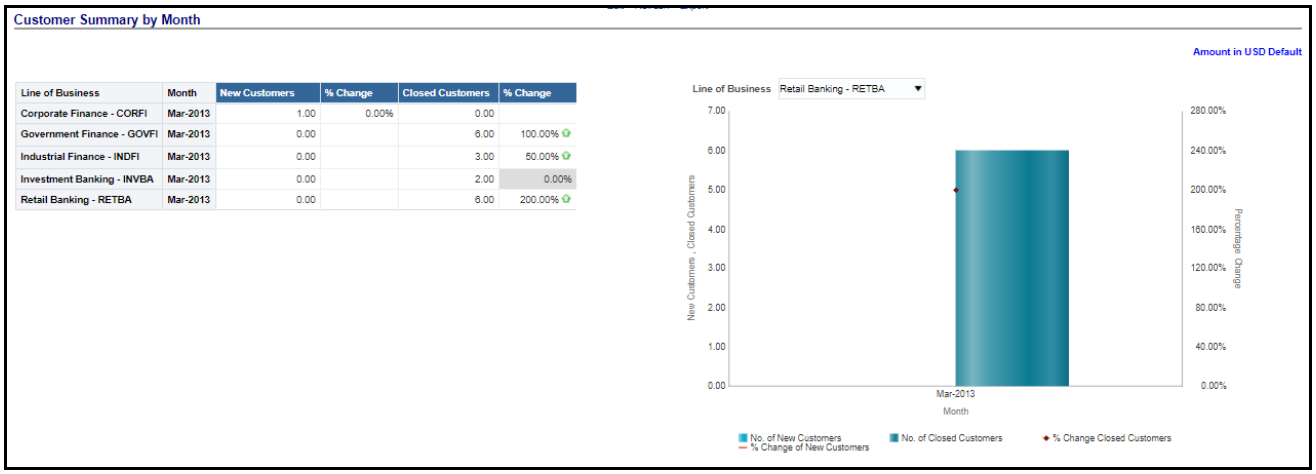
This repost has two parts:

Customer Summary by Month- Tabular Report

This Tabular report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and percentage Change. This report shows for the LOBs selected at Dashboard level LOB filter.

Customer Summary by Month-Graph Report

This report shows average no of open customers by LOB with percentage change through graph. The Graph uses a Report Level LOB Filter where single LOB can be selected to see the trend.



Revenue Summary Tab

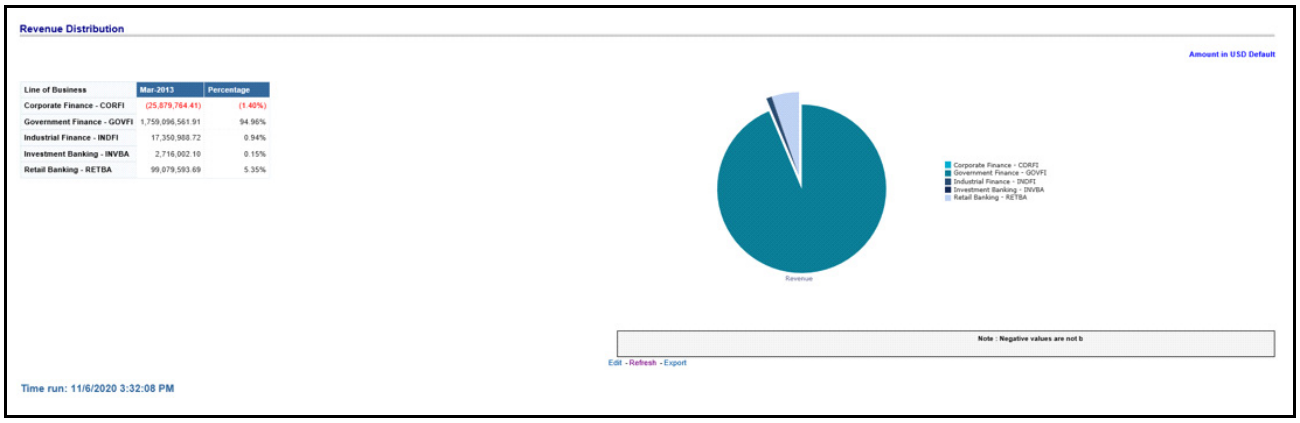
The report in this page shows Revenue Distribution of Business

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

This report has two sections:

- Revenue Distribution by LOB with Percentage of Revenue by each LOB Selected from LOB Dropdown - Tabular Report
- Revenue Distribution by LOB Selected from LOB Dropdown - Pie Chart, where each Pie Slice Represent each LOB.



Performance Summary Tab

This Page contains reports that contain:

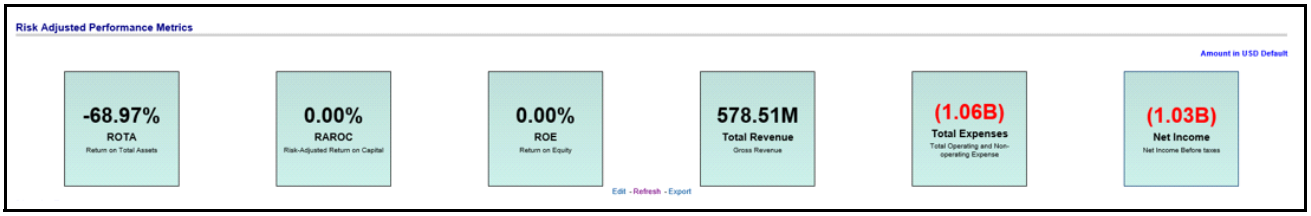
- Performance Summary Report-RAPM (Risk Adjusted Performance Metric)
- Margin Reports

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection

Risk Adjusted Performance Metric

Shows Key Performance Indicators like Return on Total Assets, Risk Adjusted return on Capital (RAROC), Return on Equity (ROE), Total Revenue, Total Expenses, Net Income, and so on.

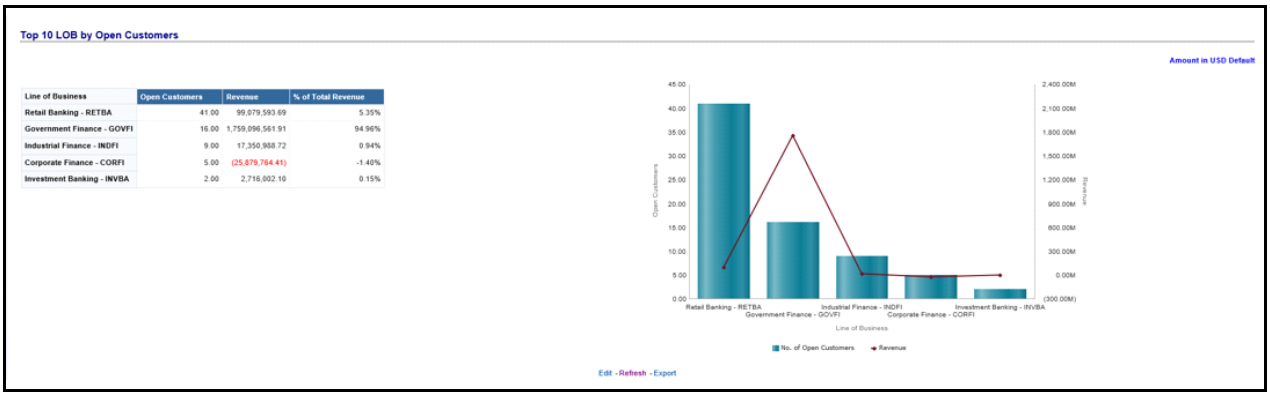


Margin Reports

Captures margin on various financial parameters.

Top N Summary Tab

This tab contains the Top Rank Tables Reports. The Report displayed is TOP N Summary.



Customer Distribution Tab

This tab contains Customer Distribution Reports which has two parts as follows:

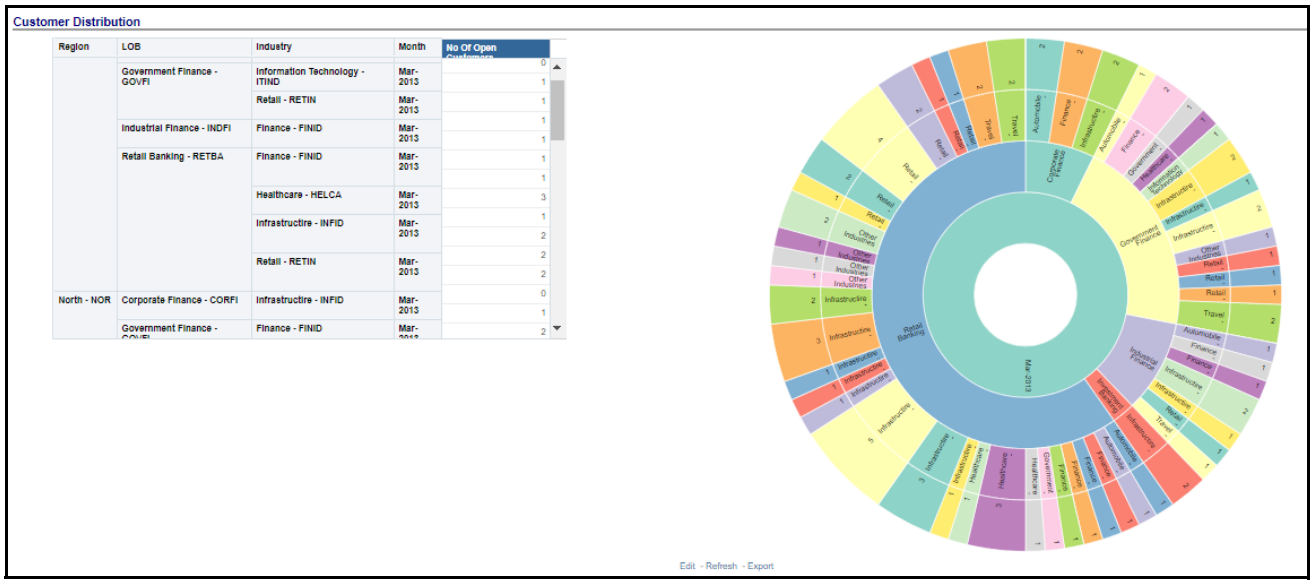
- Tabular Report
- Pie Wheel Report where Customer is distributed across LOBs and Industry as Pie Slices

This report shows Distribution of Customers across LOBs. This Report can b generated over Region, Legal Entity, Industry, As of Date, porting Currency, Amount and Period Level (Year, Month, LOB).

The Values in Month and No. Of Open Customer Columns would show Monthwise actual data

For example, the month column would show; when Period Level shows as follows:

- Period Level Select >Month - Month Column would show the current Month, No of Open Customers would show actual number of customers
- Period Level Select >Quarter - Month Column would show the months of the Quarter, No of Open Customers would show actual number of customers per corresponding month of the Quarter
- Period Level Select >Year- Month Column would show the months of the Year, No of Open Customers would show actual number of customers per corresponding month of the Year



Cross Sell Summary Tab

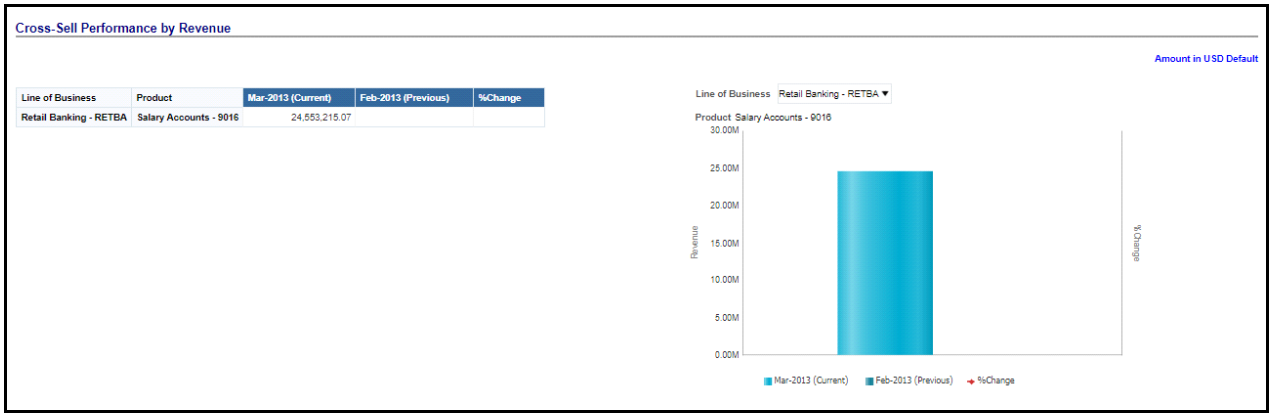
This tab contains the following reports:

- Cross-Sell Performance by Revenue
- Cross-Sell Performance by Income
- Cross-Sell Performance by Open Customers

The reports in this page can be generated at following granularity:

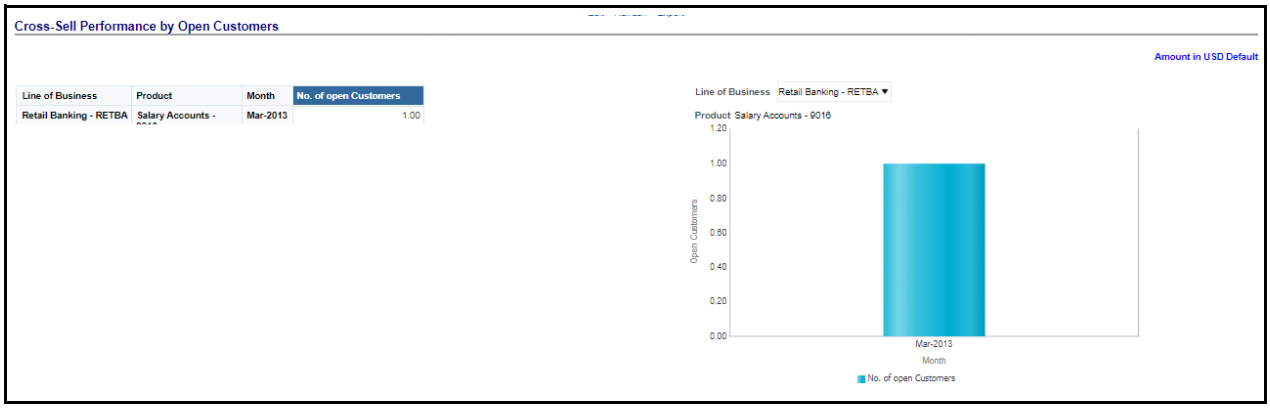
- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

Cross-Sell Performance by Revenue



Cross-Sell Performance by Income

Cross-Sell Performance by Open Customers



Product Penetration Tab

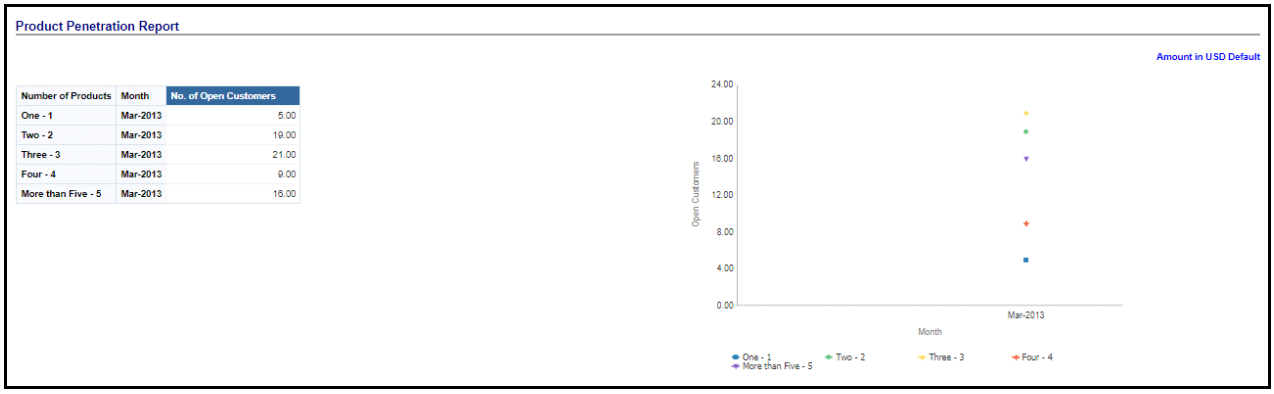
The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

This report shows number of Customers by Product and Time Period Selected. This Report can be generated over Region, Legal Entity, Industry, As of Date, porting Currency, Amount and Period Level (Year, Month, LOB).

The Values in Month and No. Of OPen Customer Columns would show Monthwise actual data, for example, the month column would show; when Period Level shows as follows:

- Period Level Select >Month - Month Column would show the current Month, No of OPen Customers would show actual number of customers
- Period Level Select >Quarter - Month Column would show the months of the Quarter, No of OPen Customers would show actual number of customers per corresponding month of the Quarter
- Period Level Select >Year- Month Column would show the months of the Year, No of OPen Customers would show actual number of customers per corresponding month of the Year



Customer Analysis Dashboard

This dashboard displays the following tabs:

- Profit and Loss Tab
- Profit and Loss (Scenario) Tab
- Performance Summary Tab
- Top N Summary Tab
- Customer 360 Tab
- Customer Group Tab
- Revenue Summary Tab

Profit and Loss Tab

This Tab/page contains two Reports

- Profit and Loss Statement
- Profit and Loss Waterfall graph report

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Customer Name/ID - Key Word Search
  - Customer Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
  - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)

Profit and Loss (Scenario) Tab

This page compares profit and loss actual values with different scenarios like, Plan, Budget, Forecast Values etc, to monitor and track the profit level situations. The comparison can be done between any scenarios. The report is a tabular one with three columns.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Customer Name/ID - Key Word Search
  - Customer Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
  - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)
- Report Level Filters
  - Scenario Selection: Scenario (Selected) Period Compared to Scenario (selected for comparison)- (Drop Down Filter)

Performance Summary Tab

This Page contains reports on containing:

- Performance Summary Report-RAPM (Risk Adjusted Performance Metric)- Shows Key Performance Indicators like Return on Total Assets, Risk Adjusted return on Capital (RAROC), Return on Equity (ROE), Total Revenue, Total Expenses, Net Income Etc
- Margin Reports - Captures margin on various financial parameters.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Customer Name/ID - Key Word Search
  - Customer Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)

Top N Summary Tab

This Page contains, Top Rank Tables Reports.

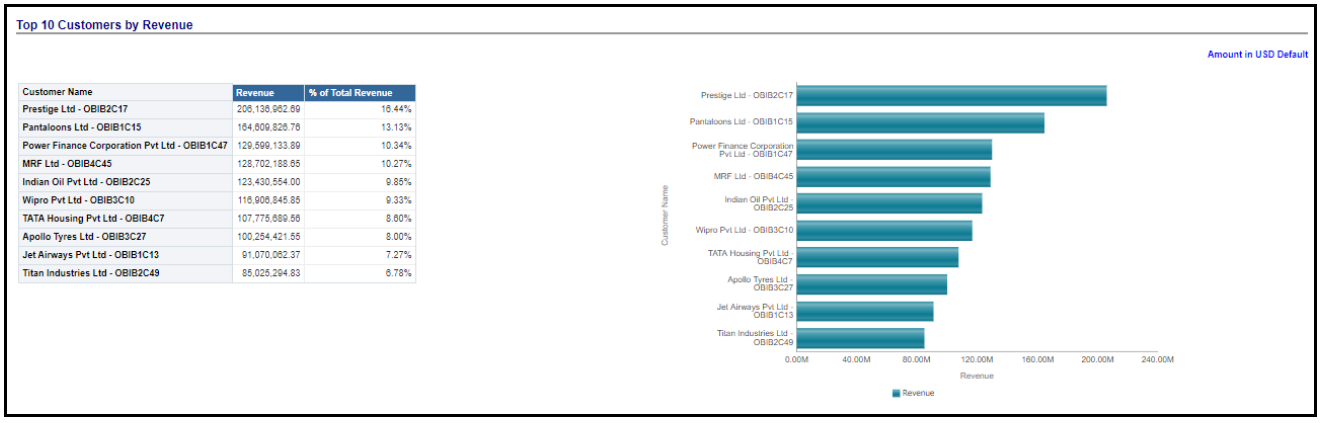
The reports in this page can be generated at following granularity:

- Dashboard Level Filters

- Search by Customer Name/ID - Key Word Search
- Customer Name (Drop Down Filter)
- Legal Entity (Drop Down Filter)
- Line of Business (Drop Down Filter)
- As of date - Calendar Date Selection
- Currency (Drop Down Filter)
- Amount - Denomination (Drop Down Filter)

- Page Level Filters
  - Period level (Radio Button Selection)

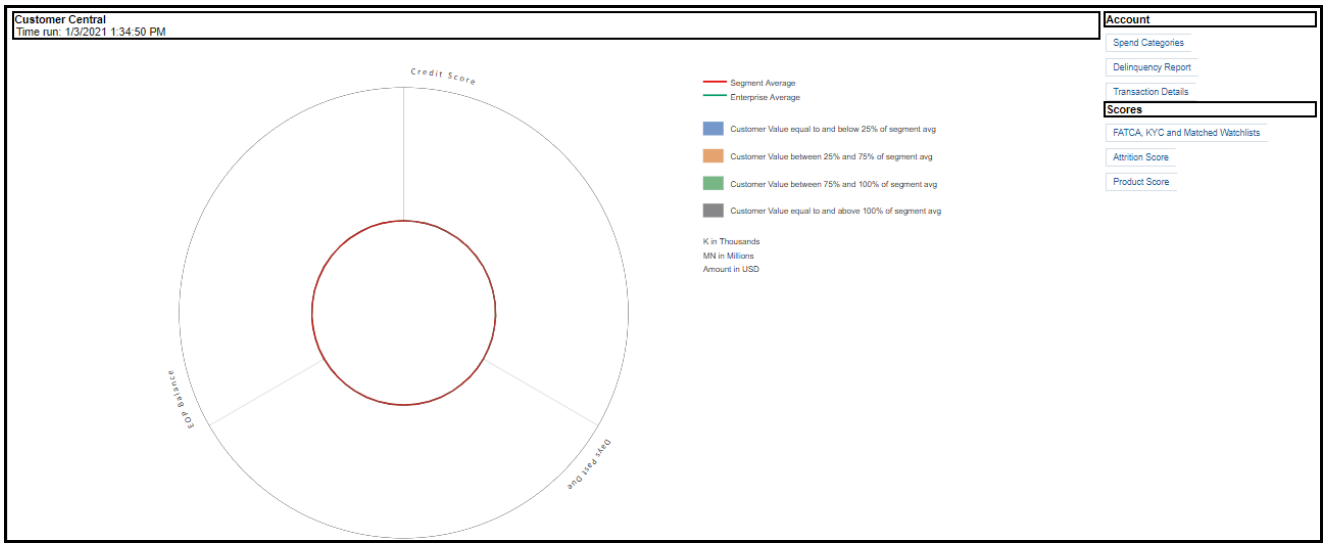
The Values shown here are average values. Please refer to the examples worksheet with Computation Logic. See the Computation Logic File for details.



Customer 360 Tab

This tab shows the following reports:

- Account Profile- Analyze By Account- Account Profile Report
- Customer 360>> Analyze By Customer>> Customer Credit Rating Jump off Report



Customer Group Tab

This tab shows the following reports:

- Customer Group Details
- Customer Group Structure

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Customer Name/ID - Key Word Search
  - Customer Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)

Revenue Summary Tab

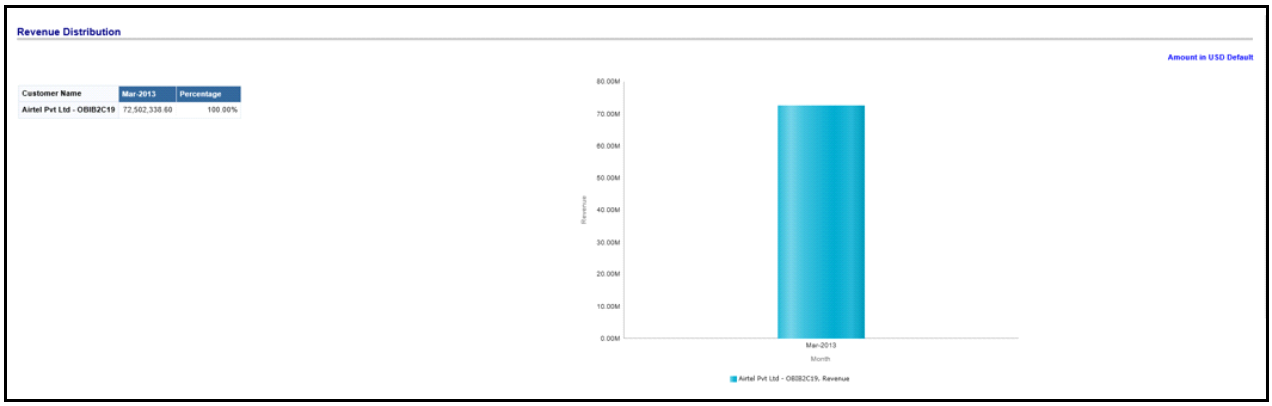
The report in this page shows Revenue Distribution of Business.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Customer Name/ID - Key Word Search
  - Customer Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

The Revenue Summary report has two sections:

- Revenue Distribution by LOB with Percentage of Revenue by each LOB Selected from LOB Dropdown - Tabular Report
- Revenue Distribution by LOB Selected from LOB Dropdown -Pie Chart, where each Pie Slice Represent each LOB.



Manager Analysis Dashboard

This dashboard has the following tabs:

- Profit and Loss Tab
- Profit and Loss (Scenario) Tab
- Cross Sell Summary Tab
- Revenue Summary Tab
- Relationship Manager Performance

Profit and Loss Tab

This tab contains two Reports:

- Profit and Loss Statement
- Profit and Loss Waterfall graph report

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Manager (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
  - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)

Profit and Loss (Scenario) Tab

This page compares profit and loss actual values with different scenarios like, Plan, Budget, Forecast Values, and so on, to monitor and track the profit level situations. The comparison can be done between any scenarios.

The report is a tabular one with 3 columns.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Manager (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
  - Scenario Selection: Scenario (Selected) Period Compared to Scenario (selected for comparison)- (Drop Down Filter)

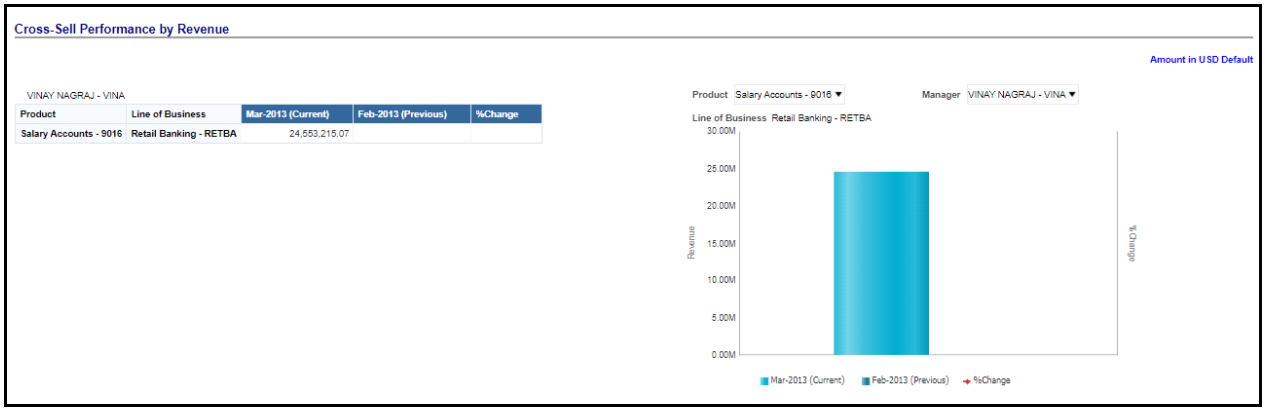
Cross Sell Summary Tab

This tab displays the following reports:

- Cross-Sell Performance by Revenue
- Cross-Sell Performance by Income
- Cross-Sell Performance by Open Customers

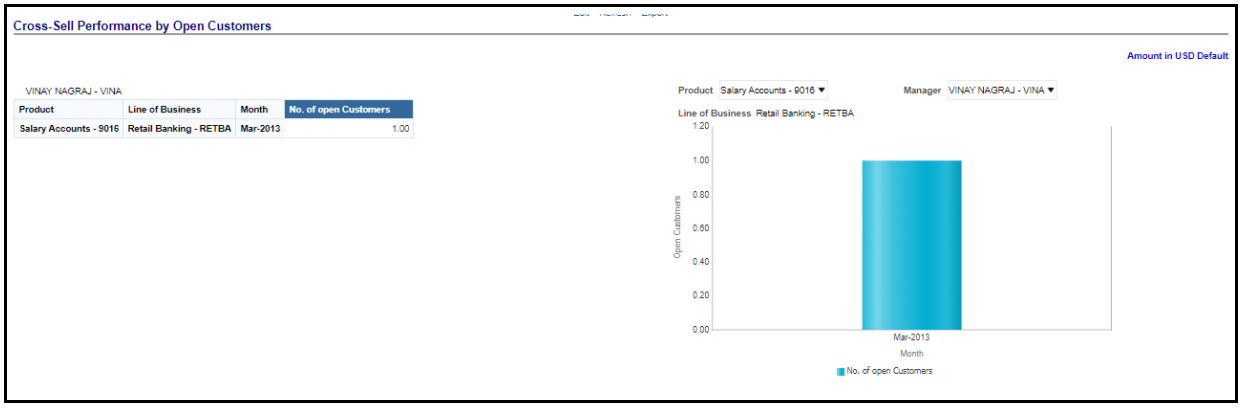
This report can be generated for Relationship Manager to measure their cross-sell efficiency and opportunity. Business that is acquired through Cross Sell in Analysis.

Cross-Sell Performance by Revenue



Cross-Sell Performance by Income

Cross-Sell Performance by Open Customers



Revenue Summary Tab

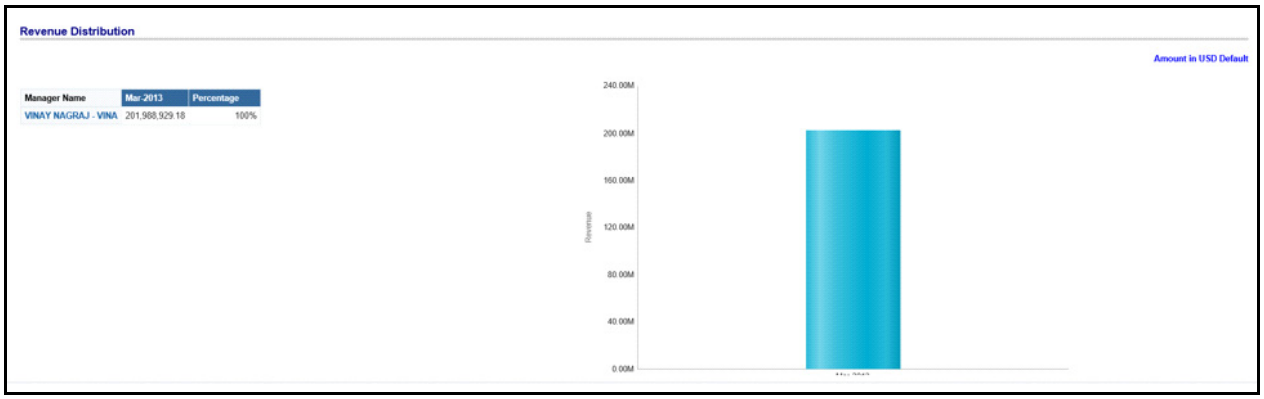
The report in this page shows Revenue Distribution of Business.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Search by Manager Name/ID - Key Word Search
  - Manager Name (Drop Down Filter)
  - Legal Entity (Drop Down Filter)
  - Line of Business (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

Th e Revenue Summary report has two sections:

- Revenue Distribution by LOB with Percentage of Revenue by each LOB Selected from LOB Dropdown - Tabular Report
- Revenue Distribution by LOB Selected from LOB Dropdown - Pie Chart, where each Pie Slice Represent each LOB.



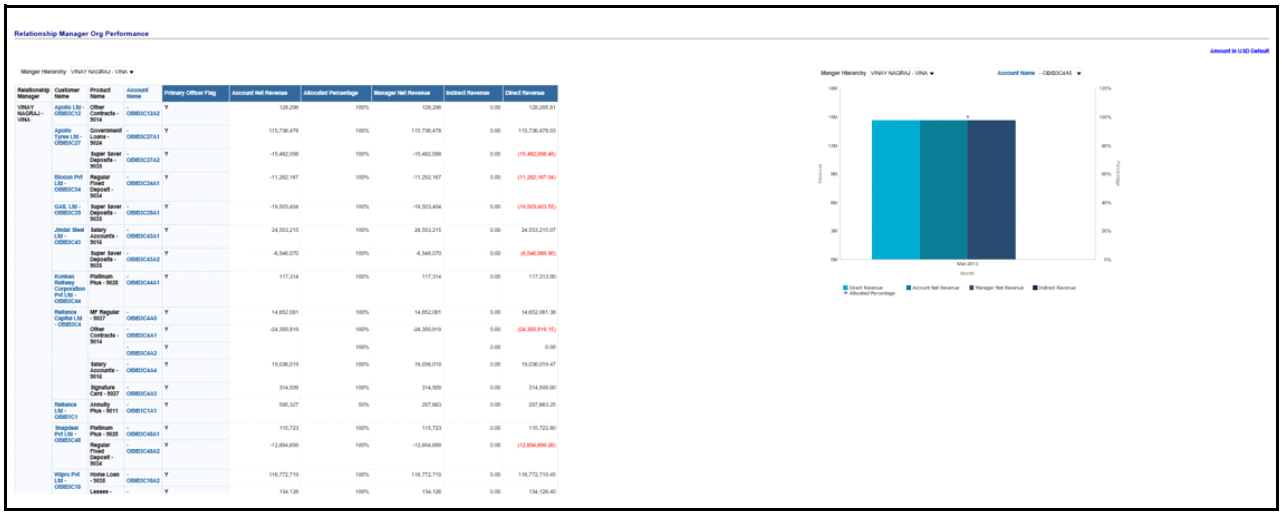
Relationship Manager Org Performance

This Report has been linked under Manager Analysis, Revenue summary Report as a Jump off/Linked report. By clicking on the bar Graph on Revenue Summary Report, the Report "Relationship Manager Org Performance Report" opens.

This report can be viewed by Customer Name, Product Name and Account.

This Page contains two Reports:

- Relationship Manager Org Performance - Table Report
- Relationship Manager Org Performance - Bar Graph



Product Analysis Dashboard

The Product Analysis Dashboard displays the following tabs:

- Profit and Loss Tab
- Profit and Loss (Scenario) Tab
- Customer Summary Tab
- Revenue Summary Tab

- Performance Summary Tab
- Top N Summary Tab
- Cross Sell Summary Tab

Profit and Loss Tab

This Tab/page contains two Reports

- Profit and Loss Statement
- Profit and Loss Summary graph report

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
  - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)

Profit and Loss (Scenario) Tab

This page compares profit and loss actual values with different scenarios like, Plan, Budget, Forecast Values etc, to monitor and track the profit level situations. The comparison can be done between any scenarios.

The report is a tabular one with 3 columns.

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)

- Product Type (Drop Down Filter)
- Product Type (Drop Down Filter)
- As of date - Calendar Date Selection
- Currency (Drop Down Filter)
- Amount - Denomination (Drop Down Filter)

- Page Level Filters
  - Period level (Radio Button Selection)
  - Period: Current (Selected) Period Compared to Previous Period - (Drop Down Filter)

Customer Summary Tab

The Customer Summary page contains the reports customers details

- Open Customer Over Time
- Customer Summary by Month

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)
- Report Level Filter(s)
  - Product Filter- for Report Open Customer Over Time- Graphical Report and Customers by Month Graph Report.

Open Customer Over Time

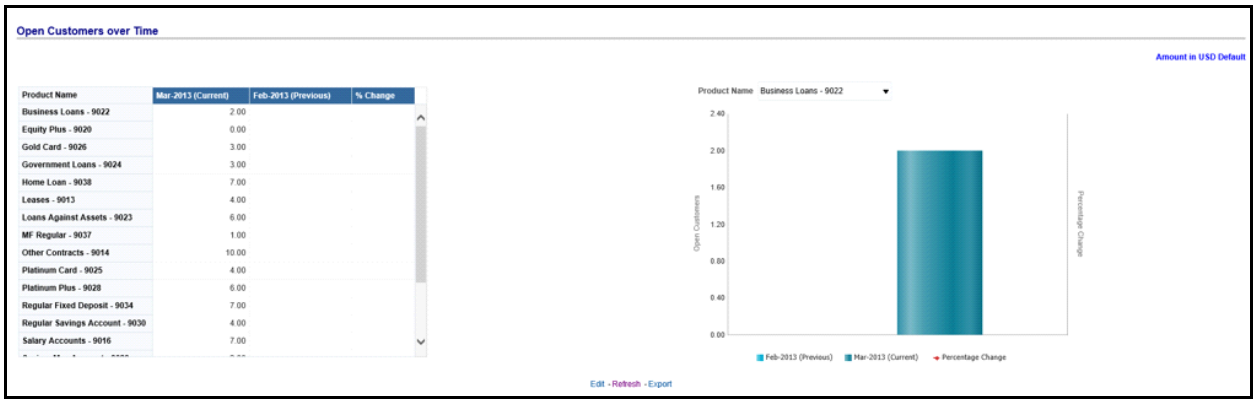
This report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and with percentage Change. The period shown in Table report will be in accordance with the period selected at the "As of Date" filter and Period Level Selection radio button.

For example:

- Selecting Year Radio Button - Would show data for 12 months of the period Selected at "As-Of_Date"
- Selecting Quarter Radio Button - Would show data for the current quarter of the period Selected at "As-Of_Date"
- Selecting Mnth Radio Button - Would show data for 1 month of the period Selected at "As-Of_Date"

This repost has two parts:

- **Open Customers Over Time- Tabular Report**: This Tabular report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would compare with previous Month and percentage Change. This report shows for the LOBs selected at Dashboard level LOB filter.
- **Open Customers Over Time -Graph Report**: This report shows average no of open customers by LOB with percentage change through graph. The Graph uses a Report Level LOB Filter where single LOB can be selected to see the trend.



Customers Summary by Month

This report shows average no of open customers, New Customers and Closed Customers by LOB, in comparison with Previous year same period and for Month, would compare with previous Month and with percentage Change.The period shown in Table report will be in accordance with the period selected at the "As of Date" filter and Period Level Selection radio button.

For example:

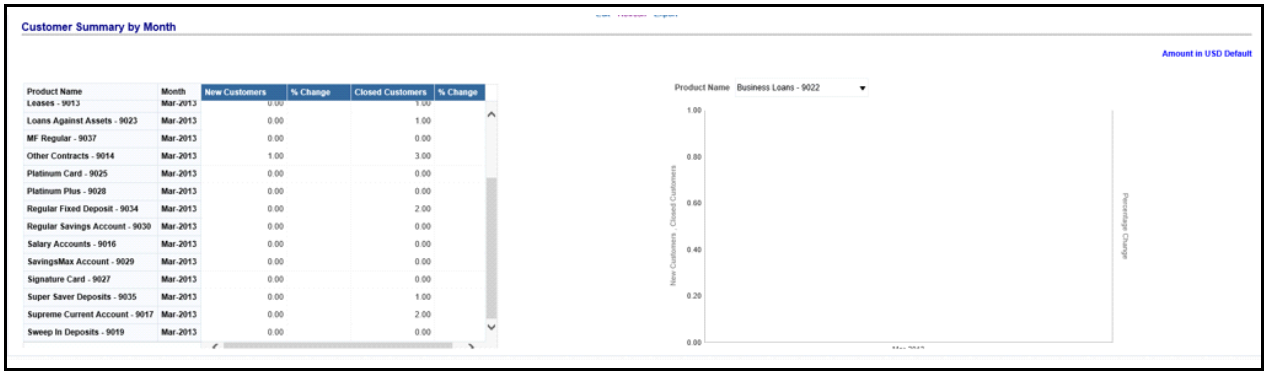
- Selecting Year Radio Button - Would show data for 12 months of the period Selected at "As-Of_Date"
- Selecting Quarter Radio Button - Would show data for the current quarter of the period Selected at "As-Of_Date"
- Selecting Mnth Radio Button - Would show data for 1 month of the period Selected at "As-Of_Date"

This repost has two parts:

- **Customer Summary by month- By New Customers and Closed Customers**: This Tabular report shows average no of open customers by LOB, in comparison with Pervious year same period and for Month, would

compare with previous Month and percentage Change. This report shows for the LOBs selected at Dashboard level LOB filter.

- **Customer Summary By Month - By Open Customers**: This report shows average no of open customers by LOB with percentage change through graph. The Graph uses a Report Level LOB Filter where single LOB can be selected to see the trend.



Revenue Summary Tab

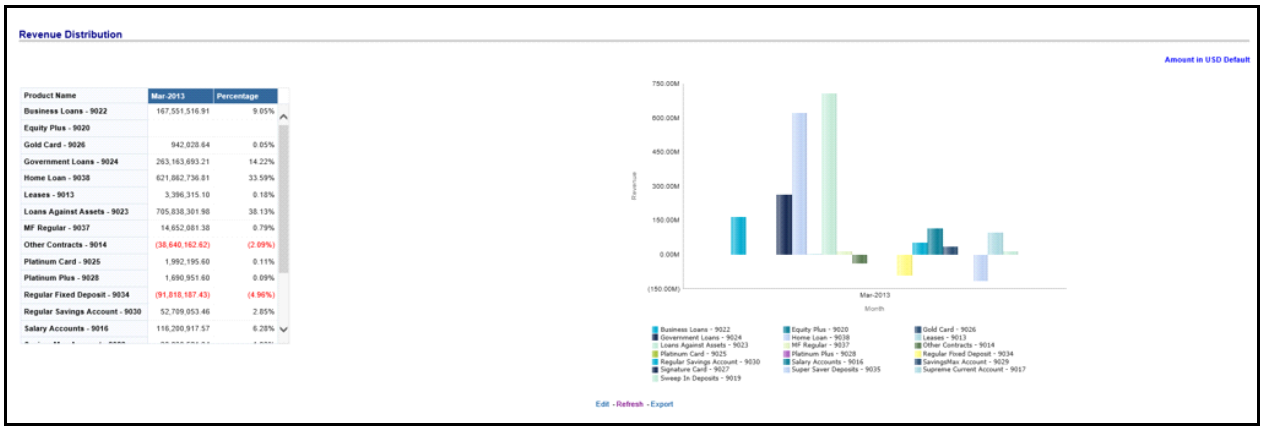
The report in this page shows Revenue Distribution of Business

The reports in this page can be generated at following granularity:

- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - Product (Drop Down Filter)
  - As of date - Calendar Date Selection
  - Currency (Drop Down Filter)
  - Amount - Denomination (Drop Down Filter)
- Page Level Filters
  - Period level (Radio Button Selection)

This report has two section:

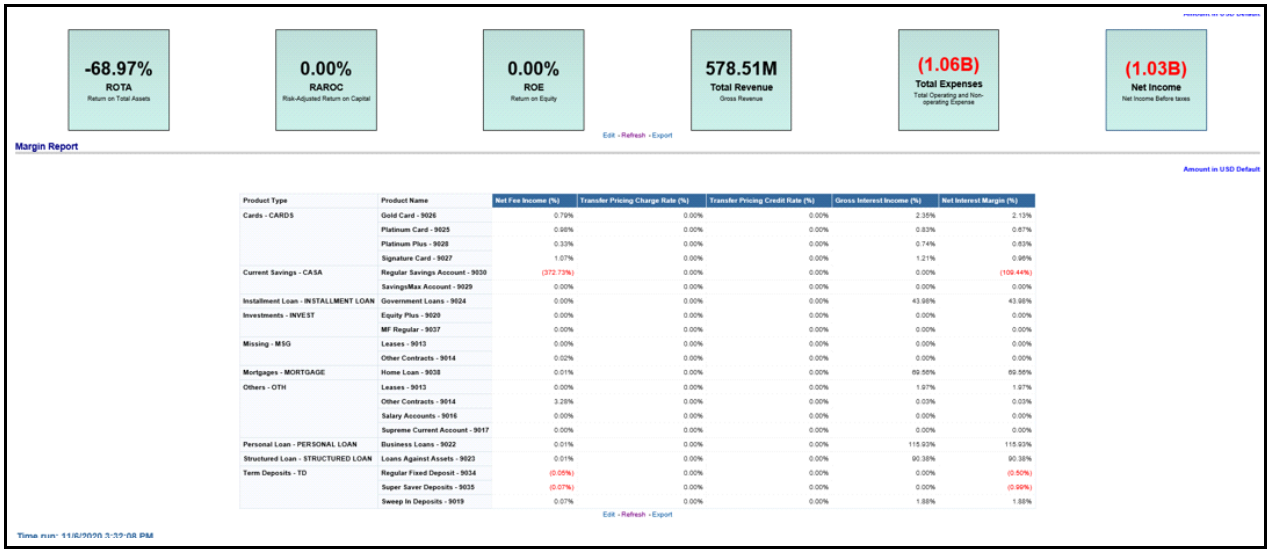
- Revenue Distribution by LOB with Percentage of Revenue by each LOB Selected from LOB Dropdown - Tabular Report
- Revenue Distribution by LOB Selected from LOB Dropdown -Pie Chart, where each Pie Slice Represent each LOB.



Performance Summary Tab

This Page contains reports on containing:

- **Performance Summary Report-RAPM (Risk Adjusted Performance Metric)**: Shows Key Performance Indicators like Return on Total Assets, Risk Adjusted return on Capital (RAROC), Return on Equity (ROE), Total Revenue, Total Expenses, Net Income, and so on.
- **Margin Report**: Captures margin on various financial parameters.



Top N Summary Tab

This tab contains, Top Rank Tables Reports.

The reports in this page can be generated at following granularity:

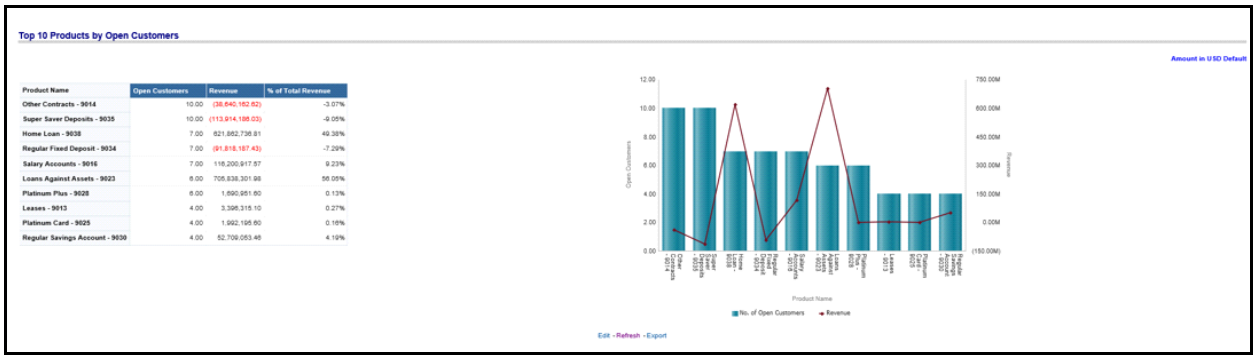
- Dashboard Level Filters
  - Legal Entity (Drop Down Filter)
  - Product Type (Drop Down Filter)
  - As of date - Calendar Date Selection

- Currency (Drop Down Filter)
- Amount - Denomination (Drop Down Filter)

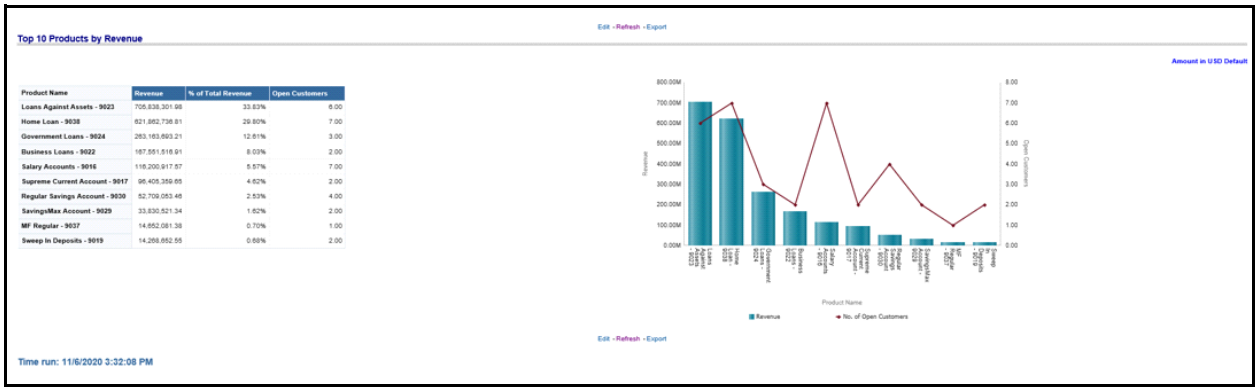
- Page Level Filters
  - Period level (Radio Button Selection)

The Values shown here are average values.

Top 10 Products by Open Customer - Table and Graph



Top 10 Products by Revenue - Table and Graph



Cross Sell Summary Tab

This tab contains the Cross Sell Summary report. This report displays the Business that is acquired through Cross Sell in Analysis.

CHAPTER 16 What-If Analysis

This chapter discusses the following topics:

Introduction

This report enables the user to account for the change in profitability owing to any probable changes in the projected components of profitability. The probable change can be defined by the user and is termed as 'Variation'. User could define the parameters to which variation is being applied and the magnitude of variation. The net effect on profitability as a result of these variations can be applied.

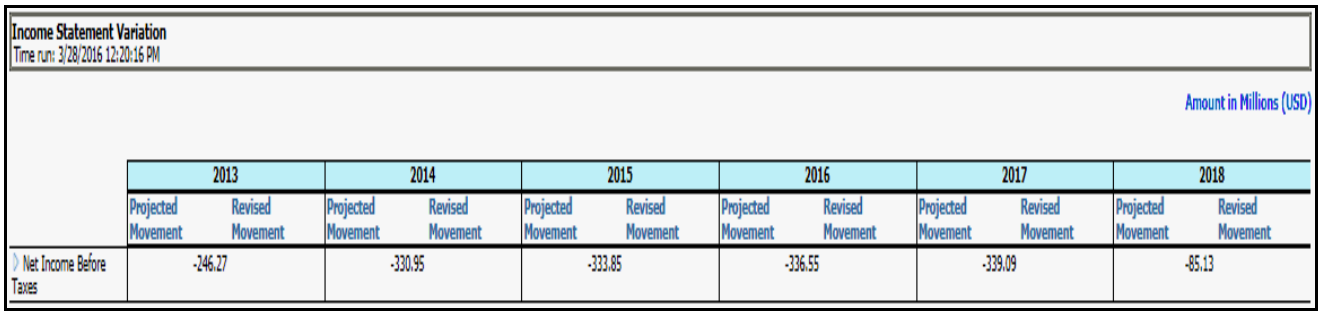
The effect of variations on profitability can be analyzed at differing levels of granularity like enterprise, LOB, Product, Customer, and Account. This selection is enabled to the user through dashboard prompt selections. The projected data of income statement is available at an account level. Aggregations are done based on the desired level of granularity. The projections are created based on historical data of account.

User could define the variations through the UI, which when imposed on the income statement provide the resulting net income. The resulting income statement post applying the variations is called 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. The magnitude of variation being applied can be specified either in 'percentage' or 'absolute'. If the variation is specified as percentage then the value of the component to which variation is being applied changes by the corresponding percentage value for the specified time. Similarly, when variation is applied in absolute terms the value of the component to which variation is being applied changes by the corresponding absolute value for the specified time.

Certain users should have the authority to save a scenario which can later be accessed by other users for reference. The variations once applied can be applied on the income statement by either of the following two methods:

- Basic - The variations that are applied get simply aggregated with the modified values of components to show the resulting net income. The basic version supports variations to be applied to multiple parameters at the same time.
- Advanced - The variations that are applied also affect the other components it is correlated to and the modified values of all such parameters gets aggregated to show the resulting net income. In the Advanced version variation can be applied to only a single component at a time.

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



Income Statement Variation
Time run: 3/28/2016 12:20:16 PM

Amount in Millions (USD)

	2013		2014		2015		2016		2017		2018	
	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement
Net Income Before Taxes	-246.27		-330.95		-333.85		-336.55		-339.09		-85.13	

The following input parameters are applicable for IPA:

Table 35. What-if Input Parameters

V_PARAMETER_NAME	V_PARAMETER_VALUE
VAR_STRT_DT_IPA	This input parameter indicates the start date for the variance calculation.
VAR_END_DT_IPA	This input parameter indicates the end date for the variance calculation.
TSHLD_FCTR_IPA	This input parameter indicates the threshold factor for replines.
EXCL_LMT_IPA	This input parameter indicates the maximum outlier exclusion percentage.

Note: There is limitation on graph. Default upper limit for the graph will be based on underlying data in fact table. Variations applied above the normal can be applied using the grid. If the user wants to apply variation beyond the upper limit shown in graph, it is not possible through graph and hence, it should be applied through grid. The same applies to Negative Values. The graph does not allow applying negative variations. That is, the graph nodes cannot be dragged below the X Axis. This change needs to be done using the grid mode.

Confuguration for What-If Analysis

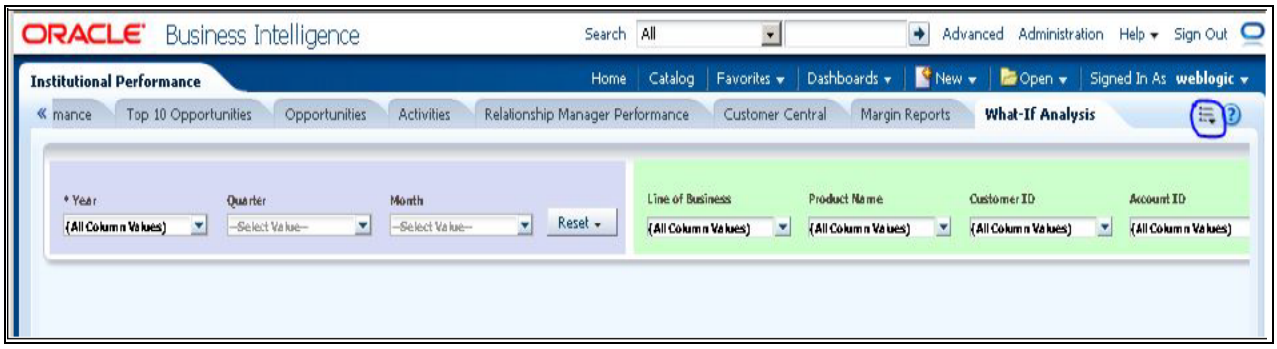
Configure connection to the What-If Variation application page in OFSAAI in the Create Scenario Analysis. OFSAAI is Oracle Financial Services proprietary tool which uses Java to enable users to apply variations on the projected data.

Example: Assume that the ofsaa hostname is `10.184.150.107` and the OBIEE analytics port is 7001. Hence, the OBIEE analytics access url would be:

http://`10.184.150.107:7001/PFT805`.

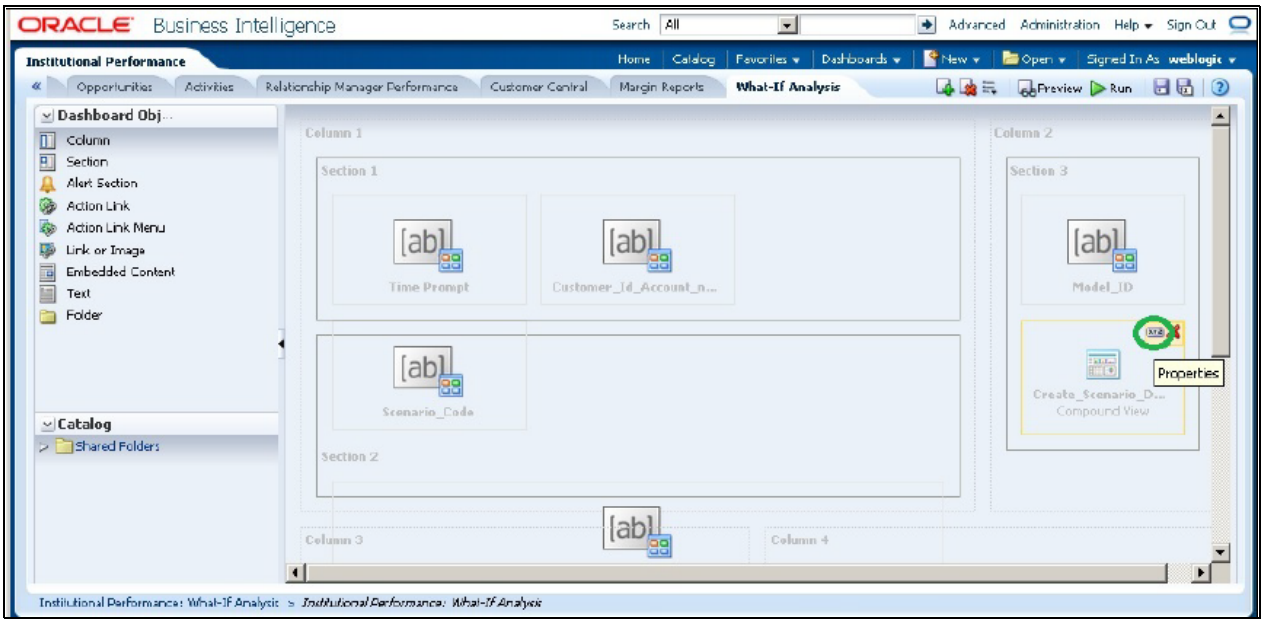
To configure these details to the What-If analysis framework, the user needs to perform the following steps:

1. Navigate to What-If Analysis Dashboard Page and Edit Dashboard Page. This page would be under the Dashboard – Institutional Performance for OFSIPA.

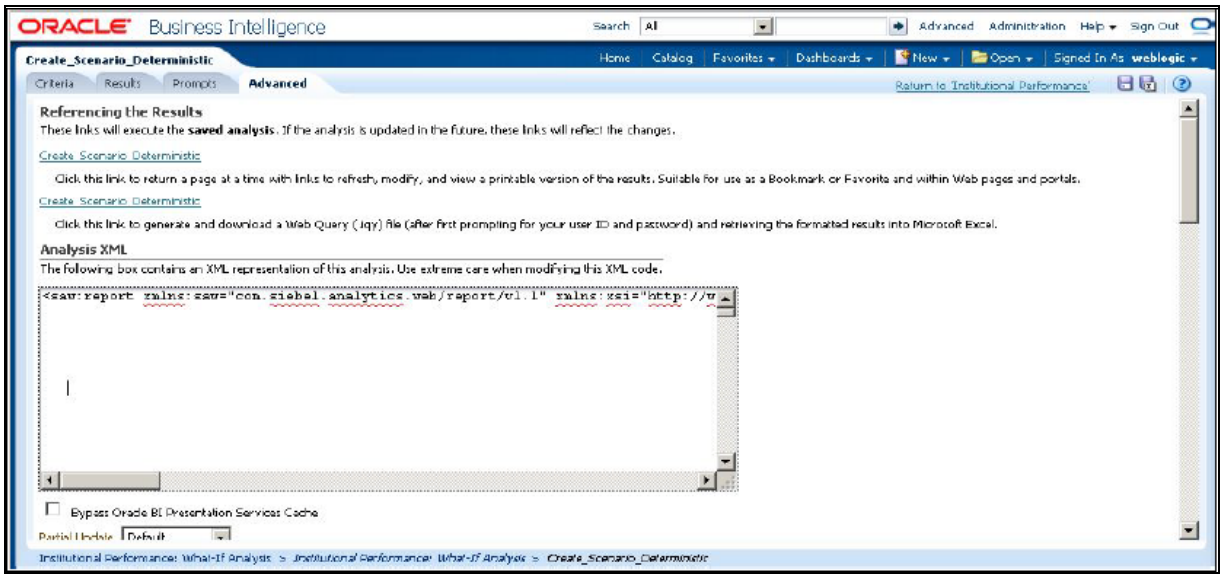


- Edit the analysis Create Scenario.

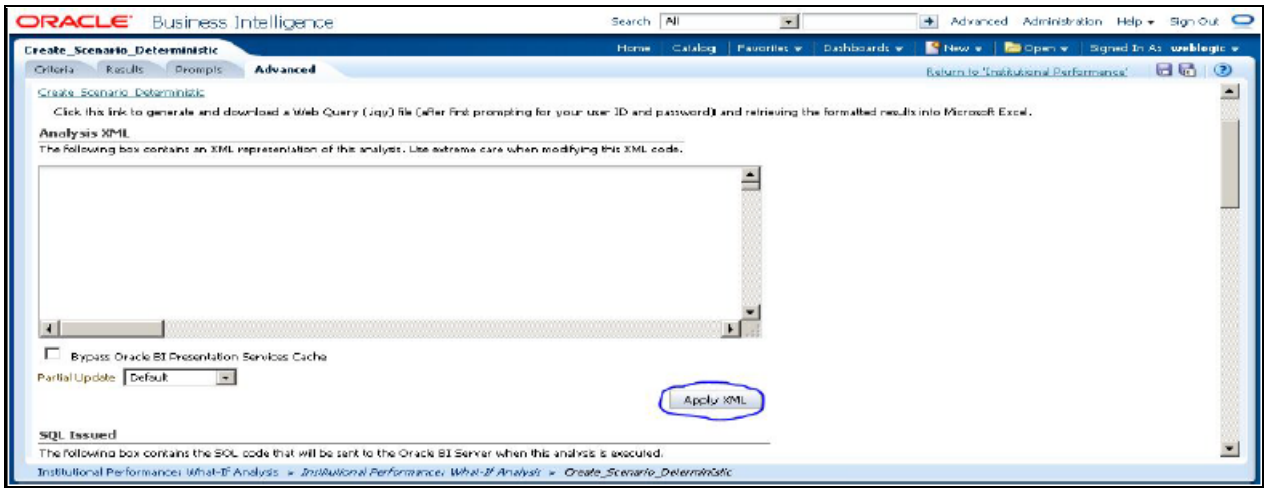
Note: Only users with OBIEE roles higher than BI Author will be able to edit.



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



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



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

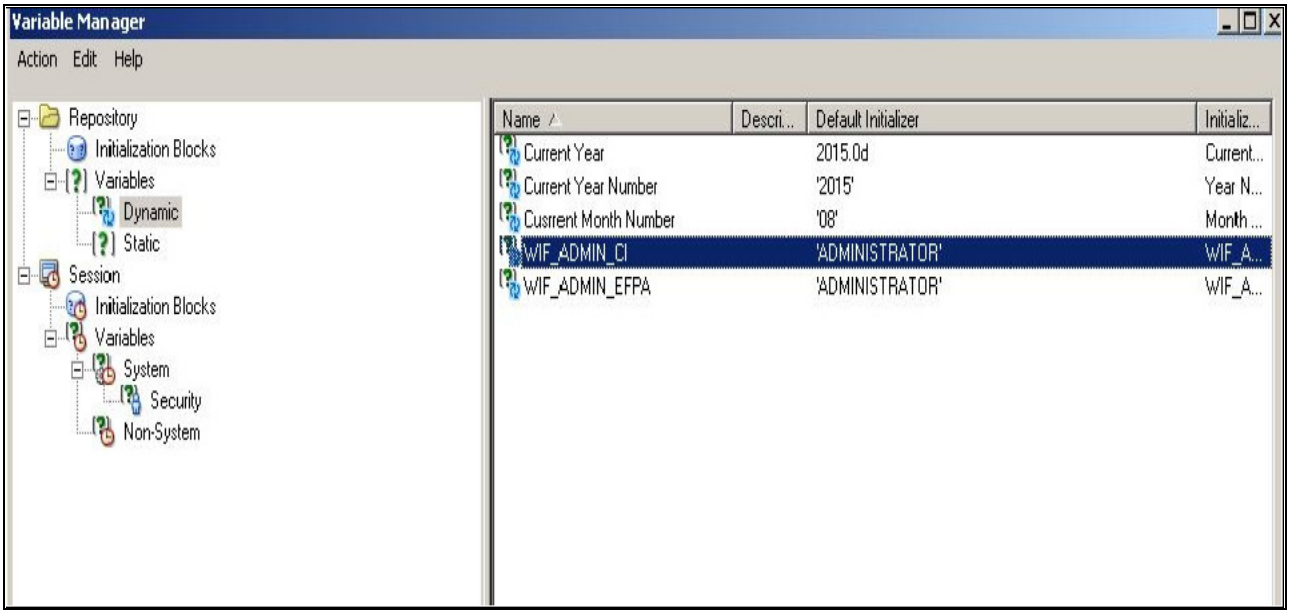
   Example: Assume that the user hostname is – 10.184.150.107 and the OBIEE analytics port is 7001. Hence, the OBIEE analytics access url would be:

   http://10.184.150.107:7001/analytics.

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

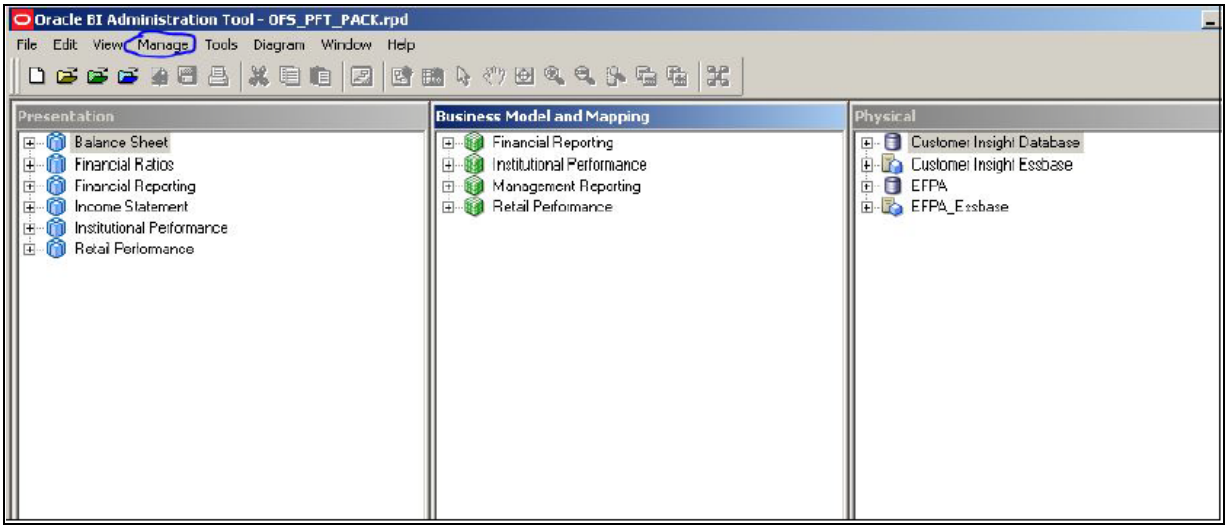
```
update fsi_m_wif_model_defn set output_page =
replace(replace(output_page,'##hostname##',
'10.184.150.107' ), '##port##','7001')
/
Commit
/
```

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

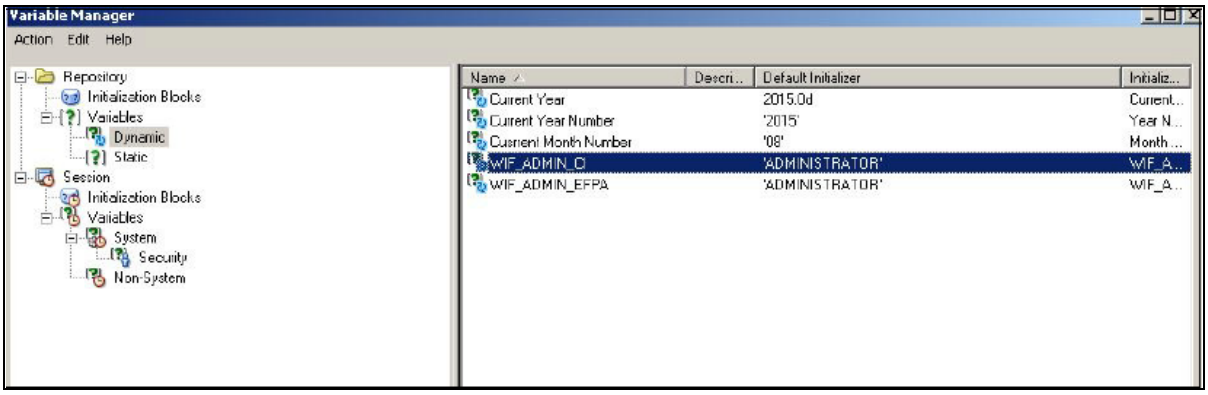


3. Configure What-If Admin Role in RPD to configure security roles to restrict Scenario Creation. If a user is not a What-If administrator, the user will only have access to ¡§Display Results¡¨. The results of this operation will not be persisted beyond one session per user.

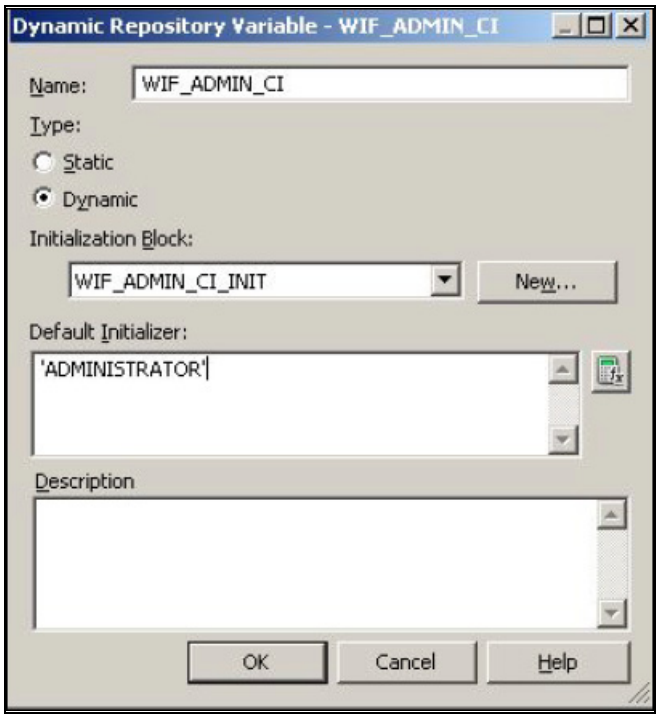
   - Open the OFS_PFT_PACK RPD and navigate to the Variable Definition Screen.



- Modify the `WIF_ADMIN_CI` Repository Dynamic Variable:



- Edit the default initializer to enter the desired What-If Administrator role. The user with this role will have the privilege to create and save a scenario. Users without this privilege will only be able to create a scenario, but not save it.



4. 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/`:

   Insert the below tag inside the "Context" tag as the first nested tag:

   <Loader delegate="true"/>

   Insert the following attributes for all the "Resource" tags under the "Context" tag :

   ```
   removeAbandonedOnBorrow = "true"
   removeAbandonedOnMaintenance="true"
   ```

Example :

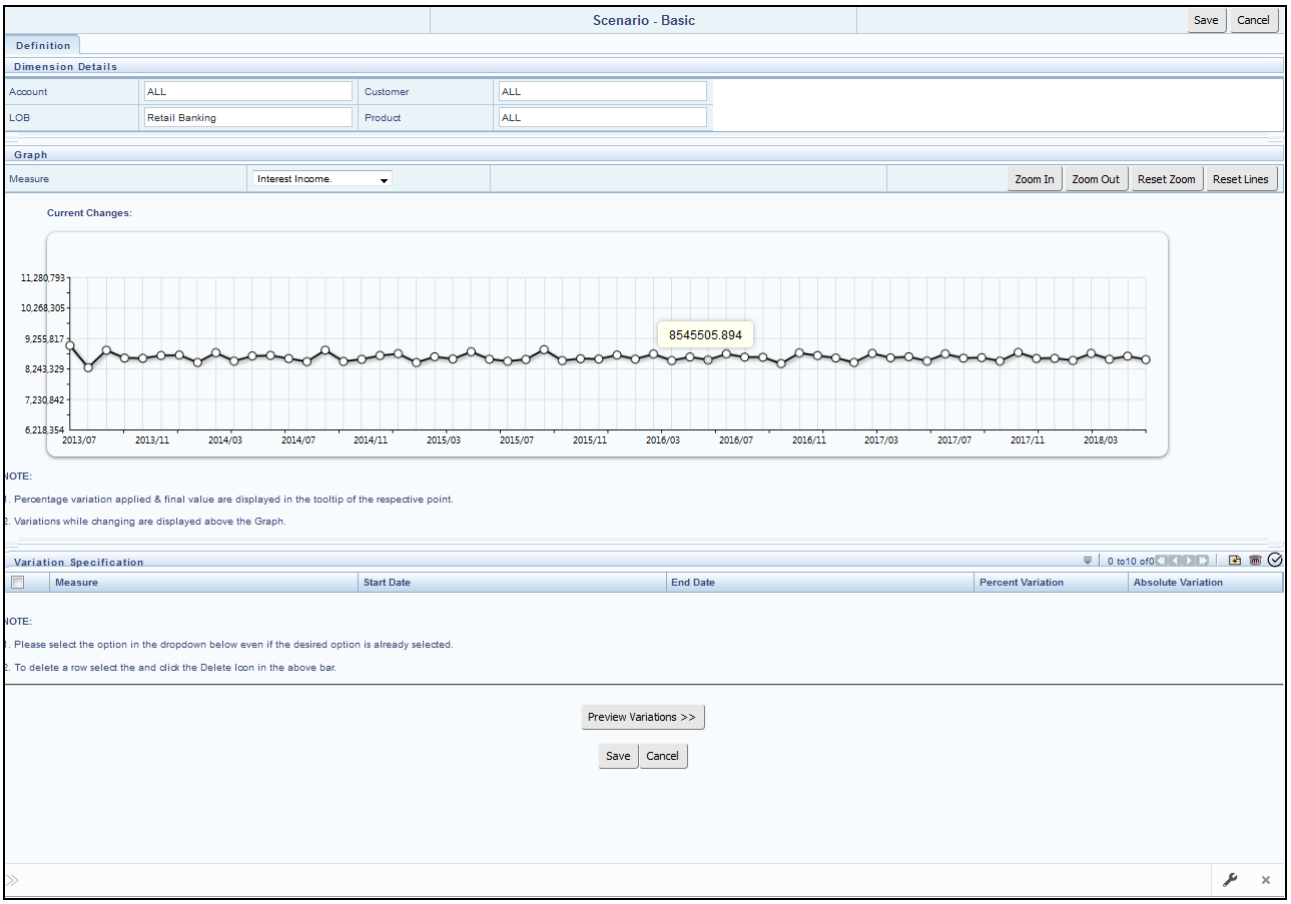
```
<Context path="/PFT" docBase="/scratch/ofsaaapp/tomcat-7.0.19/webapps/PFT" debug="0" reloadable="false" crossContext="true">
        <Loader delegate="true"/>
        <Resource auth="Container"
                  name="jdbc/FICMASTER"
                  type="javax.sql.DataSource"
                  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"/>
```

Basic Scenario

The following procedure describes the steps to create a variation:

1. Navigate to the OBIEE Dashboard page of What If Analysis.
2. Click **Create Scenario** after selecting the relevant deminsions to display the **Scenario - Basic** screen.



3. Apply the necessary **Dimension Details** for the following:
   - Account
   - Customer
   - LOB
   - Product
4. Select the relevant repline **Measure** from the dropdown list to which you want to apply the variation.
5. Select a point on the graph and drag to apply the desired variations. Percentage variation and applied, final values in the dropdown are displayed on the graph and as a tooltip on the point that is being changed.
6. Select the relevant details for the following under **Variation Specification** section of the screen.
   - Measure

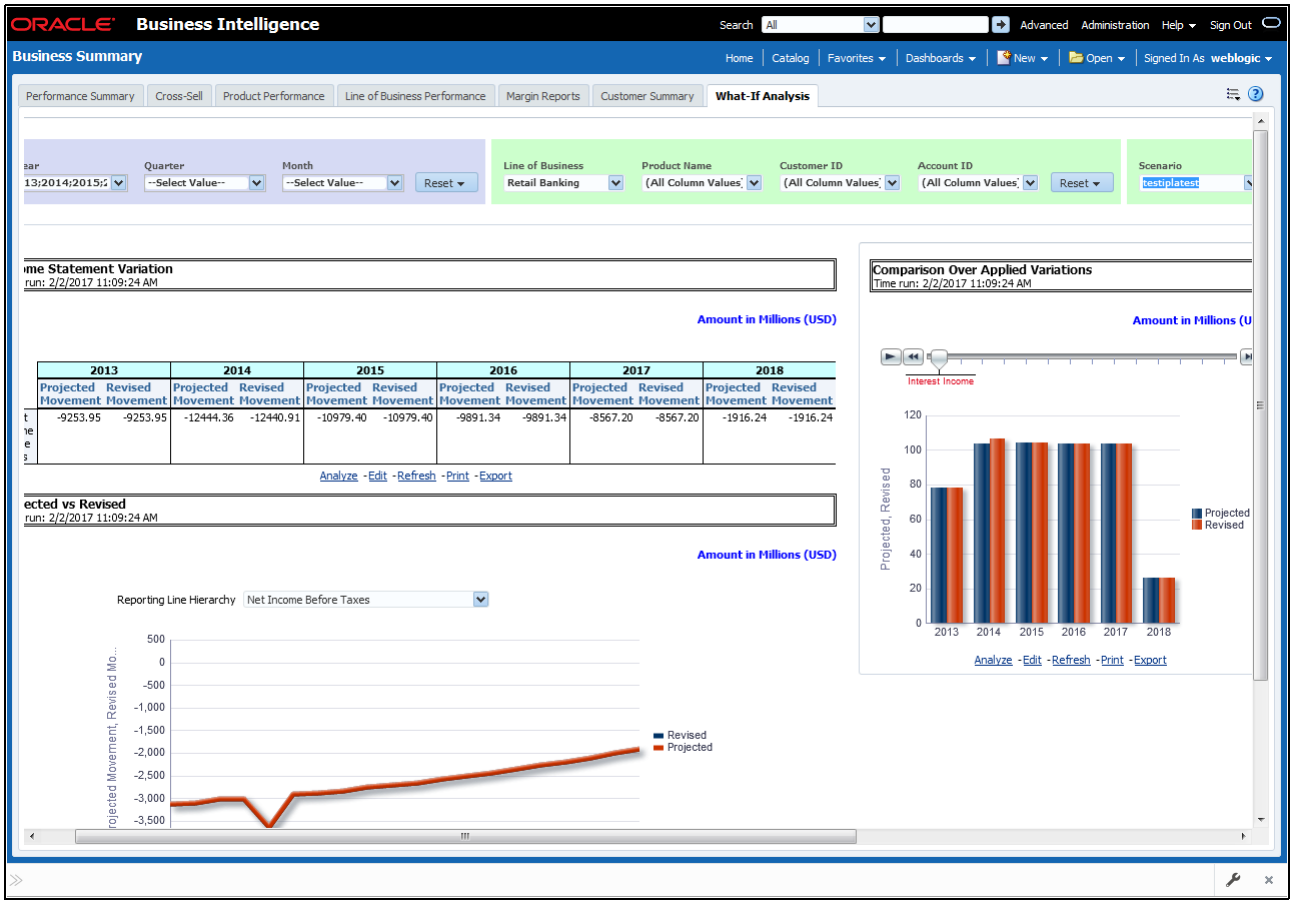
- Start Date
- End Date
- Percent Variation: Enter the % value. If you enter this, you cannot enter the Absolute Variation.
- Absolute Variation: Enter the absolute value. If you enter this, you cannot enter the Percent Variation.

If there is an overlap in dates among various variation specification rows, the same can be overridden to apply the respective change.

Always the most recent changed value is considered for variation and the other value will be cleared.

7. Click **Preview Variations** to see the results.
8. Click **Save** after confirming the variations. After this, you will be redirected to the OBIEE screen where the applied variations can be seen and analyzed further.

   The OBIEE screen is displayed as shown below:



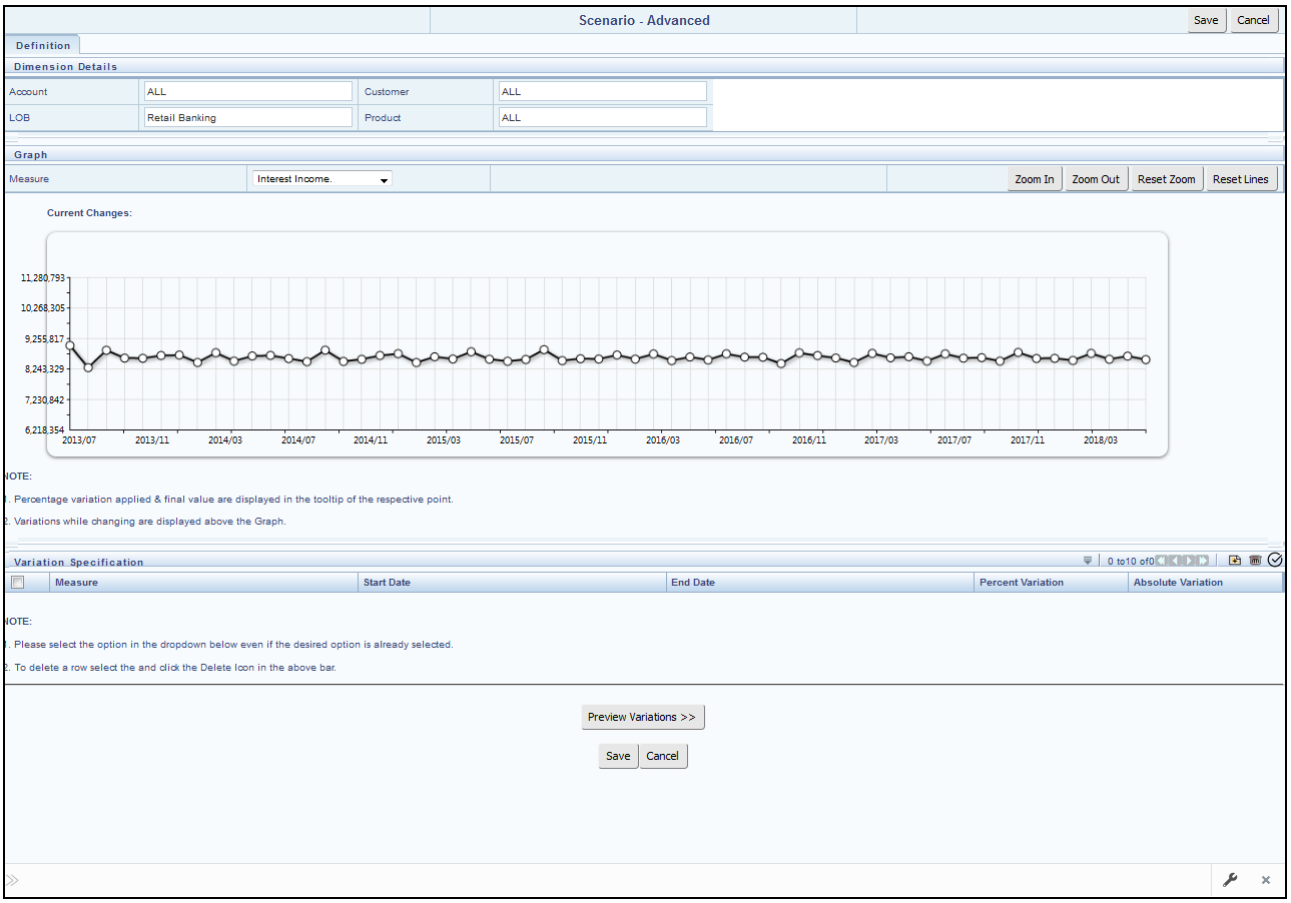
Advanced Scenario

In the advanced scenario, when a change is applied to a repline, then all its corresponding correlated changes affecting other replines are also made. These correlated changes can be preview in the preview results view by clicking on the Preview Variation button at the bottom of the page.

The following procedure describes the steps to create a variation:

1. Navigate to the What If Definition dashboard to display the Scenario - Advanced screen.

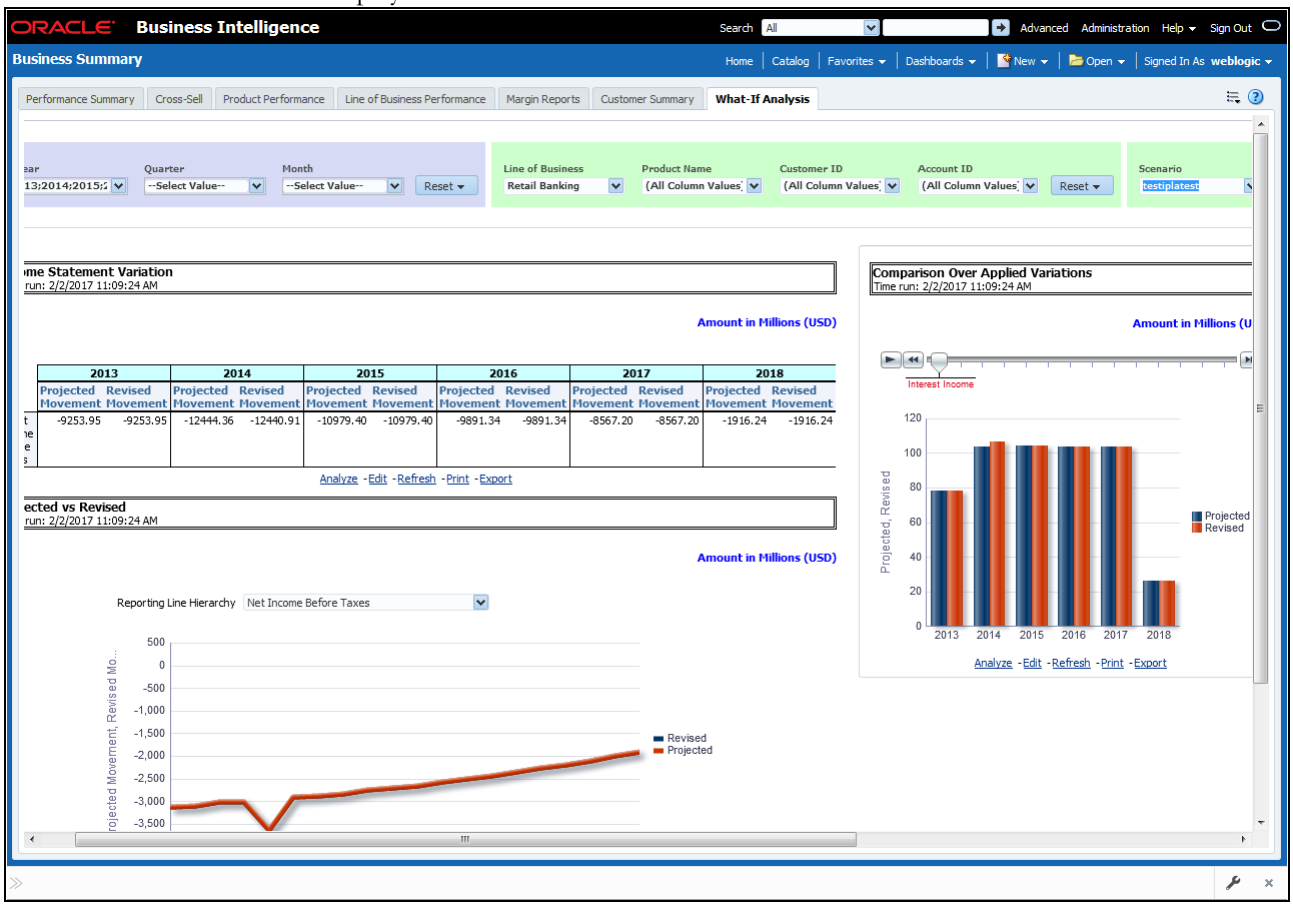
   This screen allows you to zoom in, zoom out and reset the zoom.



2. Apply the necessary **Dimension Details** for the following:
   - Account
   - Customer
   - LOB
   - Product
3. Select the relevant repline **Measure** to which you want to apply the variation.
4. Select a point on the graph and drag to apply the desired variations.

5. Click **Preview Variations** to see the results.
6. Click **Save** after confirming the variations. After this, you will be redirected to the OBIEE screen where the applied variations can be seen and analyzed further.

   The OBIEE screen is displayed as shown below:



IPA Integration with Price Creation and Discovery (PCD)

The input from IPA to PCD has been enhanced from the current setup with projected values for extended future periods based on the available current actual data for each customer account. Output on current and projected values have been determined for each product type through various measures such as EOP Balance, Fee Income, Other Income of Customers, Expenses, Credit utilization ratio, and credit card revolving rate.

Reporting Line Correlation Calculations

For the reporting lines, regression co-efficents are calculated using the R-model based on the threshold values. It is cosidered that a pairwise relationship exists between independant and dependant reporting lines.

In what-if analysis, you can make variations to the value of a variable. Variations can be applied only to the below reporting lines in the income statement:

- Interest Income
- Interest Expenses
- Transfer Pricing Charge
- Transfer Pricing Credit
- Non Interest Income
- Operating Expenses
- Net Credit Losses
- Other Revenue

The following parameters are available in the FSI_MODEL_PARAMETERS table:

- Start date of the reference time period
- End date of the reference time period
- Percentage of values that lie within the threshold
- Percentage of outliers that need to be removec

The following steps are used in repline correlation calculation:

1. Excluding Outliers
2. Testing for Stationarity
3. Testing for Cointegration

Excluding Outliers

For each variable, the sigma and mean are calculated within the reference time period as defined in the database. If the value of variable lies out side the threshold provided ,the respective pairs are excluded for all associated variables.

Pairs are excluded based on the Mahalanobis distance,i.e., pairs are excluded in descending order of their absolute distance from the mean.

Testing for Stationarity

After the outliers are excluded, the ADF test is used to check for stationarity on the time replines. The stationary is checked for each repline at two levels: I(0) and I(1). If any time repline is not found to be stationary, do a differencing of data and repeat the test.

Note: adf.test is a function of the R-library. A limitation of the R-library is that the stationary value can be calculated only if the records or data points are more than or equal to 6.

Results are reported and used in the cointegration test.

Testing for Cointegration

After the stationary test is done, the causal relations between regression variables are checked. Then cointegration is done.

The following table shows the action that is performed for pairwise stationarity and pairwise integration based on the stationarity level:

Table 36. Testing for Cointegration

Pairwise Stationarity	Pairwise Cointegration	Action
Both I(1)	Exists	Do regression without any transformation
	Does not exist	Do regression after differencing
Both(0)	NA	Do regression without any transformation
One I(1) other I(0)	NA	Do regression after differencing I(1) series

What-If Analysis Limitation

There is limitation on graph. Default upper limit for the graph will be based on underlying data in fact table. Variations applied above the normal can be applied using the grid. If the user wants to apply variation beyond the upper limit shown in graph, it is not possible through graph and hence, it should be applied through grid.

The same applies to Negative Values. The graph does not allow applying negative variations. That is, the graph nodes cannot be dragged below the X Axis. This change needs to be done using the grid mode.

CHAPTER 17

Service Calls to IPA

This chapter discusses the following topics:

Introduction

Customer insight web service is designed to get consumed by other applications in order to get the profitability details. This web service will work at two different levels: customer level and account level.

To fetch the customer details, set the request level as customer level and the customer id for that particular customer must be part of input. To fetch account details, set the request level as account level and the account number for that particular account must be part of input.

Within one request you can request for either one or multiple customers details by sending the customer id as an input in structured input xml. Similarly, it works for accounts as well. If one customer id is invalid, then the request to fetch data for multiple customers will get completely discarded. This way is similar for account numbers at account level.

Server side settings

In the server side, there is a file CUSTINSconfig.properties which allows server side user to configure web-service. The following attributes can be configured:

- Infodom

  Currently, CI web service will be enabled for one infodom and one user only at a time for a setup. That information domain has to be provided here.

- UserId

  Currently, CI web service will be enabled for one infodom and one user only at a time for a setup. The server side user has to provide this information as the web service do not validate user/password.

- Locale

  Provide 'en_US'

- runId

Provide 'VIEW_PROF_WS_RUN' as this is a generic web service which allows any package to be called at run-time.

- threadWaitTime

  Provide a number here. Unit of the value will be milliseconds. This is a time that a web service call waits for an output to be generated. If an output is ready within that time, it is sent to the user else a system generated Reference Id is sent to the user, using which the user can get the data later as explained in the Input Structure.

Client Side Settings

In the server side, there is a file CUSTINSClientConfig.properties which allows server side user to configure web-service. The following attributes can be configured:

- wsdlSchemaLocation

  This is the url of wsdl. To generate it, take contextURl, for example, http://10.241.32.163:9085/OFSAAI73new and append " /CustomerProfitabilityService?wsdl" to it. In this case, the wsdlSchemaLocation will be "http://10.241.32.163:9085/OFSAAI73new/CustomerProfitabilityService?wsdl".

- targetNamespaceURI

  Provide this as http://webservice.customerinsight.custIns.fsapps.ofs.com/ at all time.

- serviceName

  Provide this as "CustomerProfitabilityService".

Input Structure

Input for this web service is an xml file. The required information is embedded into suitable xml tags. Input xml structure, that is, request xml will vary based on request level and the type of request. If the request is new, then it will have one input format and for polling the same request the structure will be different.

Examples

- Input xml for new request to get particular customer's detail

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
<REQUEST>
<REQUEST_TYPE>VIEW_PROFITABILITY</REQUEST_TYPE>
<REFERENCE_NUMBER></REFERENCE_NUMBER>
<REQUEST_LEVEL>CUSTOMER_LEVEL</REQUEST_LEVEL>
<PARTY_ID>OBIB1C47</PARTY_ID>
<REQUEST>
```

  Here for a new request, the request type must be VIEW_PROFITABILITY otherwise it throws an error saying invalid request type. Here `REFERENCE_NUMBER` must be blank as it is new request. As you would like to fetch customer data you must put the request level as `CUSTOMER_LEVEL`. And the value for the tag `PARTY_ID` specifies the customer id whose detail you need to fetch.

When you need to fetch data for multiple customers the request xml will be as mentioned below.

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
<REQUEST>
<REQUEST_TYPE>VIEW_PROFITABILITY</REQUEST_TYPE>
<REFERENCE_NUMBER></REFERENCE_NUMBER>
<REQUEST_LEVEL>CUSTOMER_LEVEL</REQUEST_LEVEL>
<PARTY_ID>OBIB1C47</PARTY_ID>
<PARTY_ID>OBIB1C49</PARTY_ID>
<PARTY_ID>OBIB1C49</PARTY_ID>
<PARTY_ID>OBIB1C50</PARTY_ID>
<REQUEST>
```

Each PARTY_ID tag contains the customer id for one customer. In this way, you can request for multiple customers data.

- Input xml for polling request to get customer's detail

  When you send a new request, sometimes the request takes more processing time. So in that case, the output will be one reference number which you can use for polling for the same request. The input xml structure will remain same irrespective of your request, whether for one customer or multiple customers. It is as follows:

  ```
  <?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
  <REQUEST>
  <REQUEST_TYPE>RE_REQUEST</REQUEST_TYPE>
  <REFERENCE_NUMBER>56</REFERENCE_NUMBER>
  <REQUEST_LEVEL></REQUEST_LEVEL>
  <REQUEST>
  ```

  Here the REQUEST_TYPE tag must have the value RE_REQUEST. Provide the number which you have received as an output initially inside REFERENCE_NUMBER tag.

- Input xml for new request to get particular account's detail

  ```
  <?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
  <REQUEST>
  <REQUEST_TYPE>VIEW_PROFITABILITY</REQUEST_TYPE>
  <REFERENCE_NUMBER></REFERENCE_NUMBER>
  <REQUEST_LEVEL>ACCOUNT_LEVEL</REQUEST_LEVEL>
  <ACCOUNT_NUMBER>OBIB2C19A1</ACCOUNT_NUMBER>
  <REQUEST>
  ```

  Here for a new request, the request type must be VIEW_PROFITABILITY otherwise it throws an error saying invalid request type. Here REFERENCE_NUMBER must be blank as it is a new request. To fetch account data, provide the request level as ACCOUNT_LEVEL. And the value for the tag ACCOUNT_NUMBER specifies the account number whose detail you need to fetch.

  When you need to fetch data for multiple accounts data the request xml will be as mentioned as follows:

  ```
  <?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
  ```

```
<REQUEST>
<REQUEST_TYPE>VIEW_PROFITABILITY</REQUEST_TYPE>
<REFERENCE_NUMBER></REFERENCE_NUMBER>
<REQUEST_LEVEL>ACCOUNT_LEVEL</REQUEST_LEVEL>
<ACCOUNT_NUMBER>OBIB2C19A1</ACCOUNT_NUMBER>
<ACCOUNT_NUMBER>OBIB2C19B1</ACCOUNT_NUMBER>
<REQUEST>
```

Each ACCOUNT_NUMBER tag contains the account number for one account. In this way, you can request for multiple accounts data.

- Input xml for polling request to get account's detail

  When you send a new request it may happen that your request may take some more processing time. So in that case the output will be one reference number which you can use for polling for the same request. The input xml structure will remain same irrespective of your request was for one account or multiple accounts. It is as follows:

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes"?>
<REQUEST>
<REQUEST_TYPE>RE_REQUEST</REQUEST_TYPE>
<REFERENCE_NUMBER>56</REFERENCE_NUMBER>
<REQUEST_LEVEL></REQUEST_LEVEL>
<REQUEST>
```

  Here the REQUEST_TYPE tag must have the value RE_REQUEST. Provide the number which you have received as an output initially inside REFERENCE_NUMBER tag.

Output Structure

- Output xml when the request is still processing at both customer and account level

```
<?xml version = "1.0" encoding = "UTF-8"?>
<RESPONSE>
<STATUS>TIMEOUT</STATUS>
<REFERENCE_NUMBER>36</REFERENCE_NUMBER>
<RESPONSE>
```

- Output xml structure when you send invalid customer id

```
<?xml version = "1.0" encoding = "UTF-8"?>
<RESPONSE>
<STATUS>ERROR</STATUS>
<ERROR_MESSAGE>FAILED TO FETCH CUSTOMER
DETAILS</ERROR_MESSAGE>
<RESPONSE>
```

- Output xml structure when you send invalid account number

```
<?xml version = "1.0" encoding = "UTF-8"?>
```

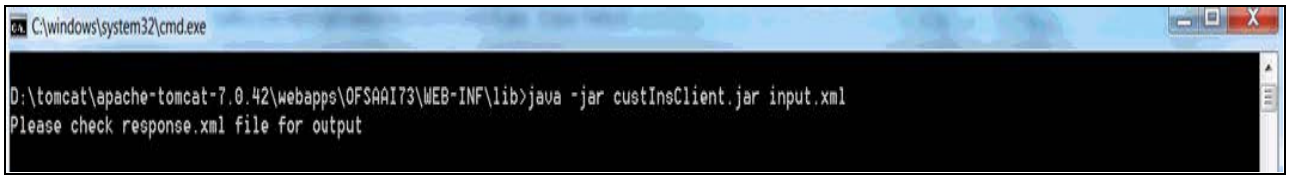
```
<RESPONSE>
<STATUS>ERROR</STATUS>
<ERROR_MESSAGE>FAILED TO FETCH ACCOUNT
DETAILS</ERROR_MESSAGE>
<RESPONSE>
```

- In case of successful response for customer or account level request, the output will be in the form of structured xml document
  - Output for successful customer level request
  - Output for successful account level request

Execute Service

To process the customer/account/re-request level request, pass one argument.

File name which contains request Input XML.



CHAPTER 18

Visibility

This chapter discusses the following topics:

Introduction

Visibility is implemented in order to restrict the user's access to the data. The user can view based on the role and the privileges assigned to the user.

Data Visibility

Data visibility refers to the data control established on the results fetched by reports depending on the user logged in.

For each user, only those accounts, which are directly handled or are handled by a subordinate, are visible.

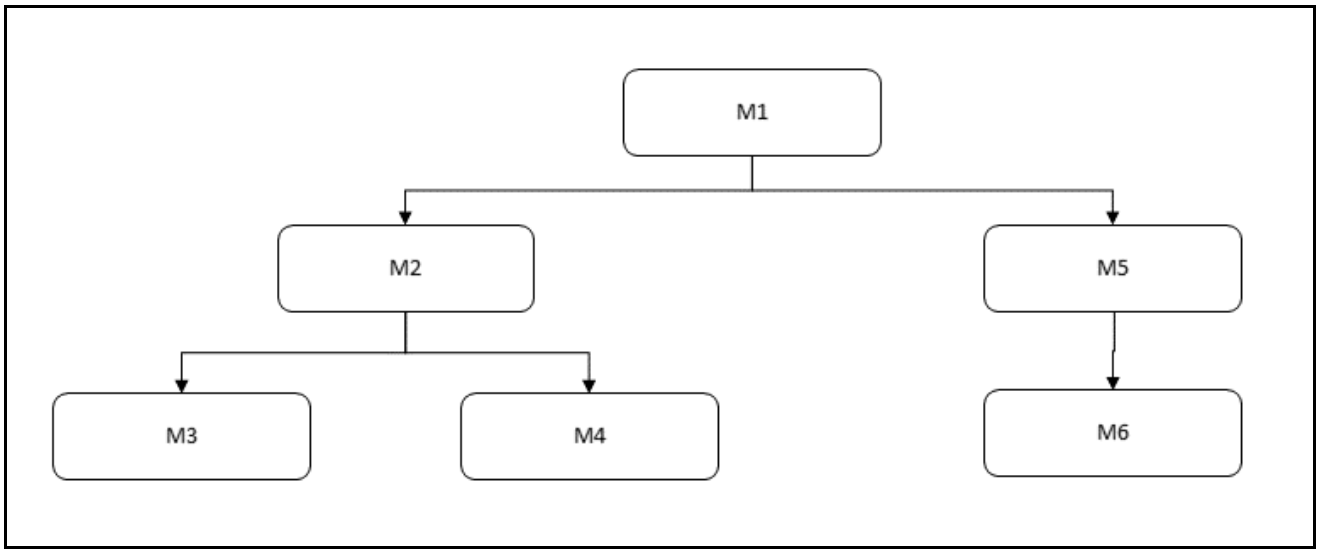
If the logged-in user is a Manager, then only those accounts which are associated with that user's organizational hierarchy will be fetched. This is achieved through the OBIEE role 'OFSAA CI Data Visibility - MGR' and using the FSI_M_USER_MANAGER_MAP table.

User has to be mapped to the user group which is assigned to the 'OFSAA CI Data Visibility - MGR' role. For more information, see the *Creating OBIEE Roles* section in the OFSIPA OBIEE Deployment Guide. After the user is created in OBIEE, then the particular log-in ID and the manager code from the DIM_MANAGEMENT table have to be populated into the FSI_M_USER_MANAGER_MAP table if that user requires restricted access.

A user logging in without assigned the 'OFSAA CI Data Visibility - MGR' role should have access to the entire data available. However, a user logging in without any associated Manager code in the FSI_M_USER_MANAGER_MAP table will end up with report errors.

The entries to the FSI_M_USER_MANAGER_MAP table have to be manually inserted (for more details, see the *Data Population as per Visibility Changes* section in the OFSIPA OBIEE Deployment Guide).

The following diagram depicts a hierarchy of Managers:



The data visibility for each of the Managers, starting from the top of the hierarchy is as follows:

- M1 user has control over the data associated with that user along with the data associated with the immediate subordinates, that is, M2, M5, and their subordinates till the end of the hierarchy.
- M2 user has control over the data associated with that user along with the data associated with the immediate subordinates, that is, M3, M4, and their subordinates till the end of the hierarchy.
- M5 user has control over the data associated with that user along with the data associated with the immediate subordinate, that is, M6 and his subordinates till the end of the hierarchy.

Note: See the OBIEE documentation about 'Setting Up Row-Level Security (Data Filters) in the Repository' if data visibility must be extended.

APPENDIX A How to Add a New Dimension

This appendix discusses the following topics:

Introduction

This section explains the steps to be performed by the user for adding a new dimension to the cube.

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 will hold 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, see Data Model Utilities Guide for adding dimension tables under "Object Management" chapter.

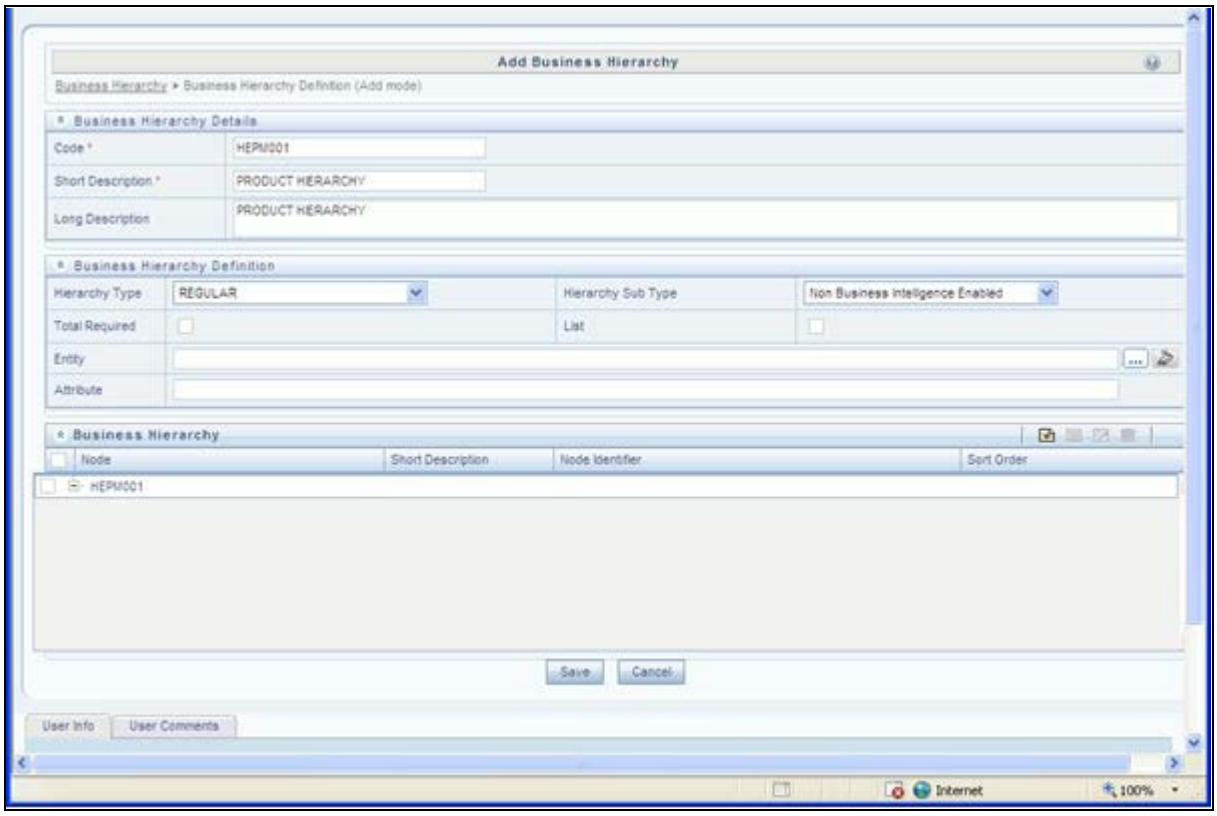
Dimension Definition Process

Step 1 - Add Business Hierarchy

To define a new **Business Hierarchy**, go to **Unified Metadata Manager**, select **Business Metadata Management** and choose the type of hierarchy.

Hierarchy Types are:

- **Regular** - For representing non-time and non-measure dimensions in a hierarchical format. For example, this type are Product, Organization Unit, and so on.
- **Measure -** For representing the measures in the hierarchical format. This corresponds to a ACCOUNT hierarchy within the ESSBASE. An example of this type is Management Reporting Line.
- **Time -** For representing the calendar or date dimension in a hierarchical format. This corresponds to a TIME hierarchy within Essbase. An example of this type is Calendar hierarchy.



Choose Hierarchy subtype. Hierarchy SubTypes are:

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

Select the "Total Required" property, if a TOTAL is required to be included as the root node of the hierarch and select the "List" property, if hierarchy is a flat list of members without any levels.

Choose the entity and attribute on which the hierarchy is based. The components for hierarchy definition differ for each subtype of the hierarchy.

If subtype is "Non Business Intelligence Enabled", then the user can add nodes and 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 that contains the child member.

For more details, see *Oracle Financial Services Analytical Applications Infrastructure 7.3* User Guide.

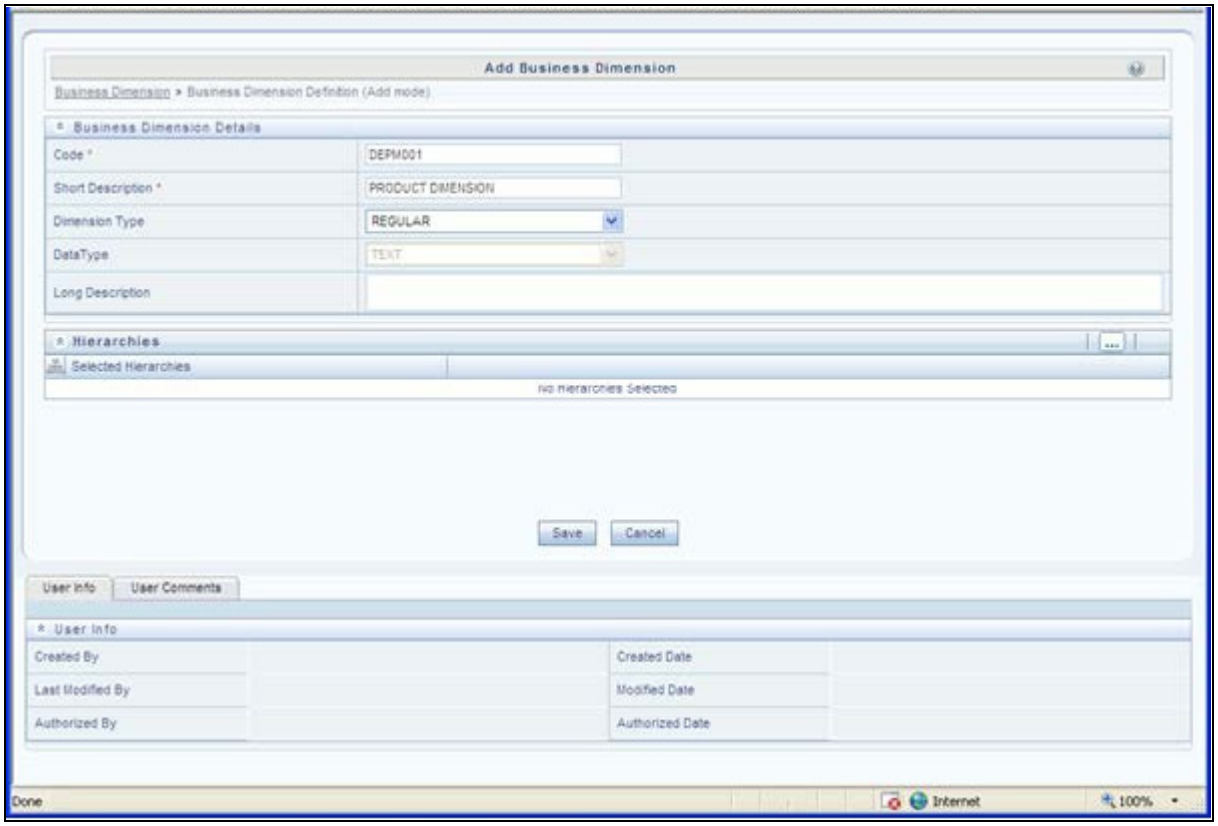
Step 2 – Add Business Dimension

To define a new **Business Dimension,** go to **Unified Metadata Manager**, select **Business Metadata Management**.

Choose the **Dimension Type**. Dimension Type is same as Hierarchy Type and helps to filter the hierarchies that will be part of the dimension. A dimension will contain one or many hierarchies. Choose the hierarchies that are part of the dimension.

The User Info grid at the bottom of the screen displays the metadata information about the Business Dimension created along with the option to add comments.

Click **Save** in **Add Business Dimension** screen to save the details.

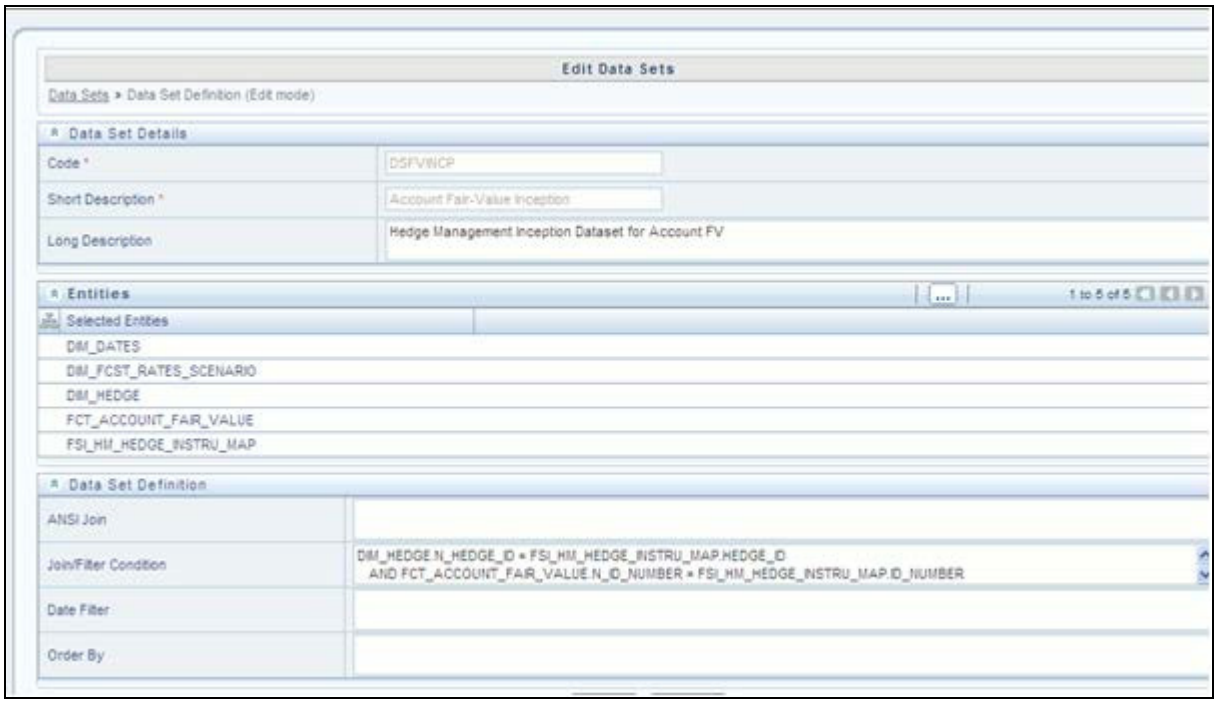


For more details, see *Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide.*

Step 3 – Modify Data Set

To modify **Data Sets**, go to **Unified Metadata Manager --> Business Metadata Management.**

Identify data sets that are based on the modified fact table. Open the data set definition. Include the new dimension table in the data set. Modify the data set JOIN to include the join clause between the fact table and new dimension table. Save the data set.



For more details, see Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide.

Step 4 – Modify Cube Definition

Modify "Cubes" in **Unified Metadata Manager -> Business Metadata Management**.

Identify the cube that needs to be modified. Open the cube definition. Add the new dimension. Map the measures to the newly added dimension and **Save** the cube definition.

For more details, see *Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide.*

Step 5 – Build Cube

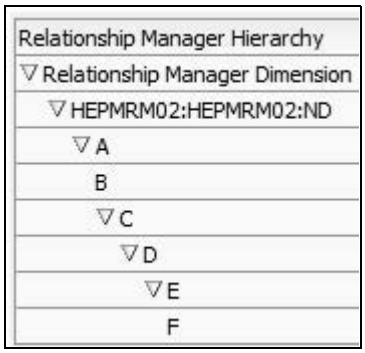
Assuming that the dimension table and fact table is loaded with relevant data, cube can be built. Define batch to execute the CREATE CUBE component that will build the outline and load data in ESSBASE.

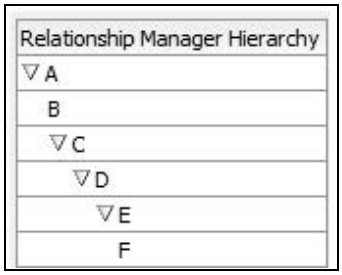
For more information on executing batch, see *Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide.*

Steps to follow while using ESSBASE Source for Relationship Manager Hierarchy

The following are the steps to follow while using ESSBASE Source for Relationship Manager Hierarchy.

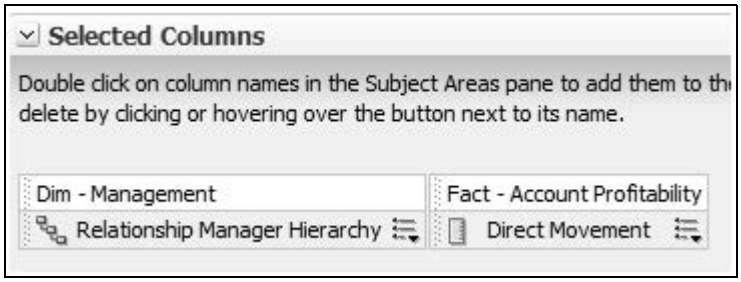
1. When creating a Parent Child hierarchy using ESSBASE, ESSBASE creates two additional parents to the existing hierarchy. For example:

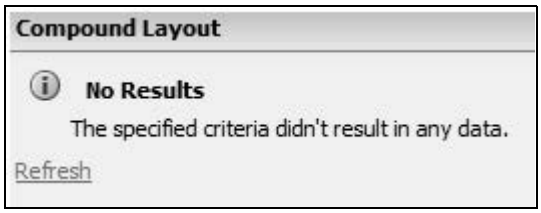




The first hierarchy is generated by RDBMS source and the second is generated by ESSBASE source. The additional parents are the Hierarchy Name and the Dimension Name of the metadata bearing the hierarchy.

2. In the context of using Relationship Manager Hierarchy for Institutional Performance, there is a concept of visibility of data implemented. This means that while using a cube source, D can see A listed as a manager in the hierarchy. However, D does not have the privilege to view the data (revenue, movement, and so on) related to A but can view the data for all the child nodes of D, for example, E and F. As a result, if Relationship Manager Hierarchy is selected along with Direct Movement, no results are displayed.

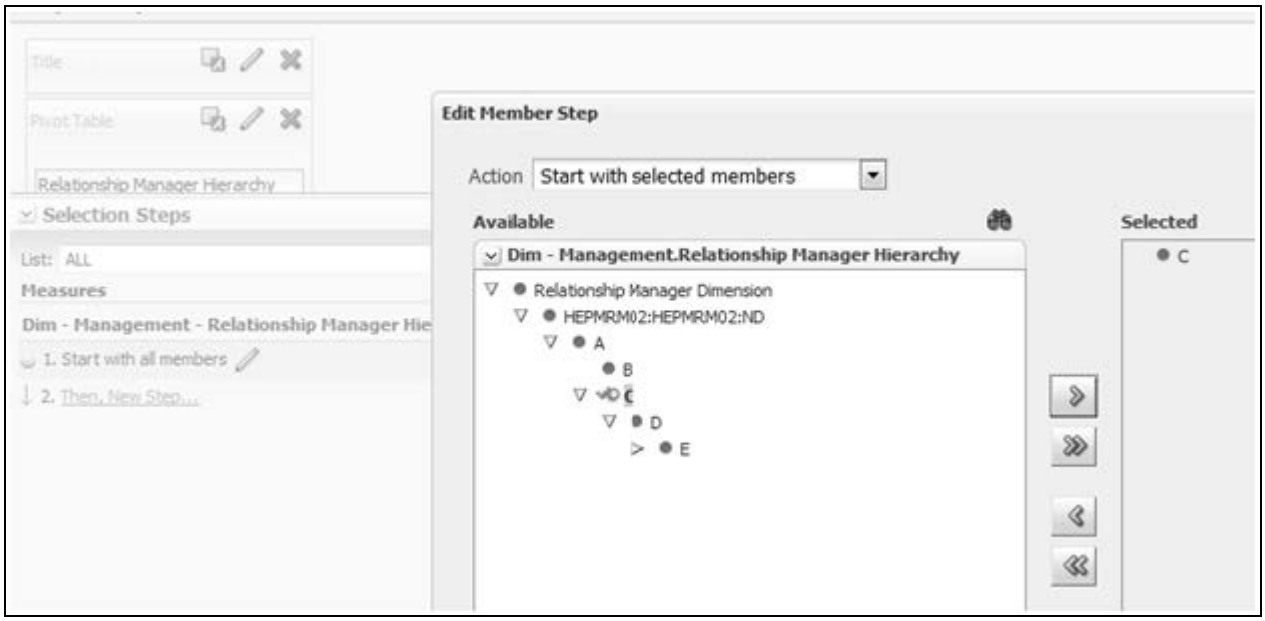




3. To view results for the logged-in Relationship Manager, the user must choose the Relationship Manager who is mapped to the user. In this case, the logged-in user is weblogic. From FSI_M_USER_MANAGER_MAP, the following is seen:

	V_USERNAME	V_MANAGER_CCDE	D.V_AM_ACCT_MANAGER_FIRST_NAME
1	RELATIONSHIP MANAGER	A01	A
2	SALES REPRESENTATIVE	A02	B
3	weblogic	A03	C

4. Thus the user must start the hierarchy with C.



As a result, the user will be able to see the data related to the manager.

Relationship Manager Hierarchy	Direct Movement
C	-827.25
D	-827.25
E	-827.25
F	-1611.25

Metadata

Technical Metadata

- **Sheet_for_DIM_STG_MAP.XLS** excel sheet lists the SCD's packaged in the IPA application.
- **OFS_IPA_Technical_Metadata.xls** lists the Institutional Performance Analytics technical metadata.

Optional Metadata

- **PFT Acc_Sum_tech.xlsx** lists the technical metadata related to PFT account summary.
- **FTP Acc_Sum_tech.xlsx** lists the technical metadata related to FTP account summary.

Business Metadata

OFSIPA Business metadata.xlsx lists the Oracle Financial Services Institutional Performance Analytics BI 6.0 Business Metadata.

Reporting Metadata

- **Customer Attributes - IPA.xlsx** lists the Customer Attributes.
- **IPA-RPD_webcat.xlsx** lists the IPA-RPD-Webcat metadata.

APPENDIX B How to Add a New Measure

This appendix discusses the following topics:

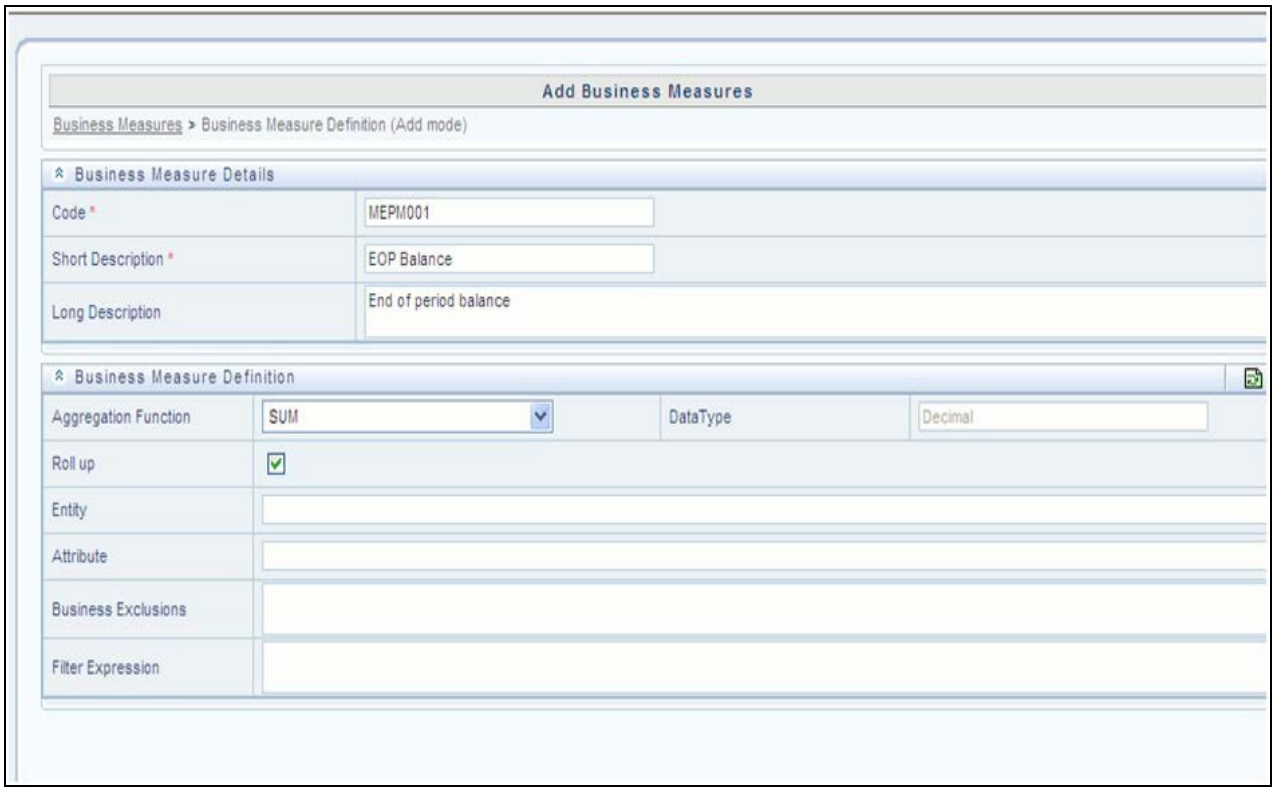
Introduction

This section details the steps to be performed by the user for adding a new measure to the cube. As a prerequisite, the fact table needs to have the column that holds values for the new measure.

Measure Definition Process

Step 1 – Add Business Measure

5. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Business Measures**.
6. 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.
7. Specify if this measure needs to be rolled up against hierarchies.
8. Select the fact table as part of the Entity.
9. Select the column of the fact table as part of the Attribute. This column will hold the value of the measure.
10. Specify Business Exclusions and Filters, if required.
11. Save the measure.



For more information on Business Measures, see Business Measures under Unified Metadata Manager chapter in *Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide*.

Step 2 – Modify Cube Definition

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Cube**.
2. Identify the cube that needs to be modified.
3. Edit the cube definition.
4. Add the new measure.
5. Map the measures to the to the required dimensions.
6. Save the cube definition.

Build Cube

Assuming that the dimension table and fact table is loaded with relevant data, cube can be built.

Define batch to execute the CREATE CUBE component that will build the outline and load data in ESSBASE.

For more information on executing the batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

APPENDIX C How to Develop a New Cube

This appendix discusses the following topics:

Introduction to Developing a New Cube

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. In case user would like to see measures against a new dimension that is not part of the existing seeded metadata, then suggest including the new dimension as part of the existing cubes instead of creating a new cube. As a prerequisite, user should have defined datasets, measures, hierarchies and dimensions before defining a cube.

Procedures to Develop a New Cube

Step 1 – Add Cube

From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Cube**. Specify the MDB details that will be created in ESSBASE.

Step 2 – Include Dimensions

Include dimensions that are part of the cube definition. Users mandatorily need to include TIME and MEASURE dimensions.

Step 3 – Specify Variations

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.

Step 4 – Specify Dataset

Specify dataset corresponding to the selected dimensions and measures. Data set will supply required data to the cube.

Step 5 – Specify Node Level Formula

If node level formula's are required to be specified for the nodes within the hierarchy, then they can be specified in this UI.

Step 6 – Save and Build

Save the cube. Define and execute batch in ICC to build the cubes.

For more information on Cubes, see Cubes under Unified Metadata Manager chapter in *Oracle Financial Services Analytical Applications Infrastructure 7.3 User Guide.*

APPENDIX D How to Define a Batch

This appendix discusses the following topics:

Introduction

Batch refers to a set of executable processes based on a specified rule. Batch Maintenance framework within OFSAAI facilitates you to create and maintain the Batch Definitions. You can process the Batch scheduled for execution from Batch Maintenance and also from other modules.

You need to have Data Centre Manager function role mapped to access the Operations framework within OFSAAI. You can access Batch Maintenance by expanding Operations section within the tree structure of LHS menu. The *Batch Maintenance* window displays a list of Batches scheduled for maintenance with the other details such as Batch ID, Batch Description, and the editable state of the Batch.

Batch Creation

You can create a batch from the *Batch Maintenance* screen as mentioned below:

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.

Table 37. Batch Details

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 alpha-numeric 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 "-".
Batch Description	Enter a description for the Batch based on the Batch Name.

Table 37. Batch Details

Duplicate Batch	(Optional) Select the checkbox 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**.

Note: For a more comprehensive coverage of configuration and execution of a batch, refer to the *Operations* Chapter in *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

APPENDIX E List of Hard-Coded Members

List of Hard-Coded Members

Following are the dimension members that are hard-coded within the application.

Table 38. Hard-coded members

Table Name	Column Name	Expected Values
DIM_CUSTOMER_TYPE	V_CUST_CATEGORY	C
FCT_CRM_ACCOUNT_SUMMARY	V_SCENARIO_CODE	PLAN, BUDGET
FCT_OPPORTUNITY_ACTIVITY	V_ACTIVITY_STATUS	O, C
DIM_BANDS	V_BAND_TYPE	AGEONBOOK TURNOVER
FCT_ACCOUNT_PROFITABILITY	N_REP_LINE_CD	98000 - Net Income Before Taxes 98500 - Tax Expense 99000 - Net Income After Taxes 107100 - Number of Customers 107130 - Number of Open Customers 107200 - Number of Accounts 107230 - Number of Open Accounts 107300 - Attrition Rate

APPENDIX F Run Rule Framework

This appendix discusses the folloiwng topics;

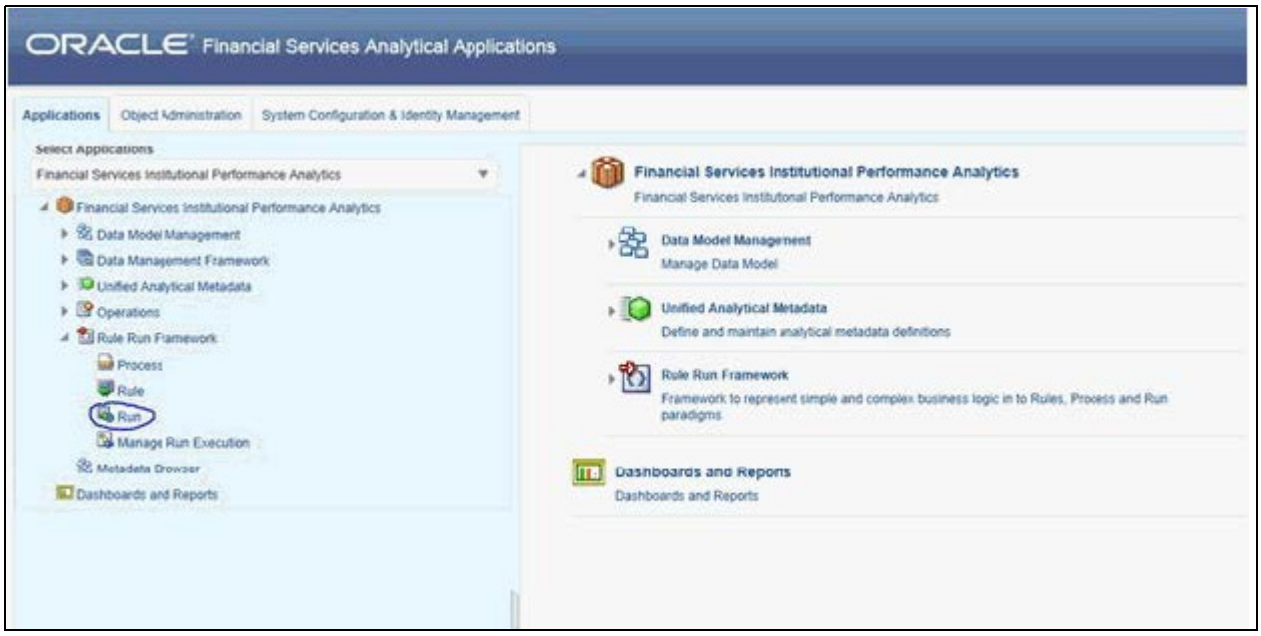
Introduction

In cases where data is required to be loaded for fact tables in multiple runs, the OFSAAI Rule Run Framework comes in handy. For example, for population of FCT_CRM_ACCOUNT_SUMMARY, the parameters of the batch execution include a parameter $RUNSK = -1. (Refer to the parameters of the batch ##INFODOM##_aCRM_CRM_Acc_Summ, Task1).

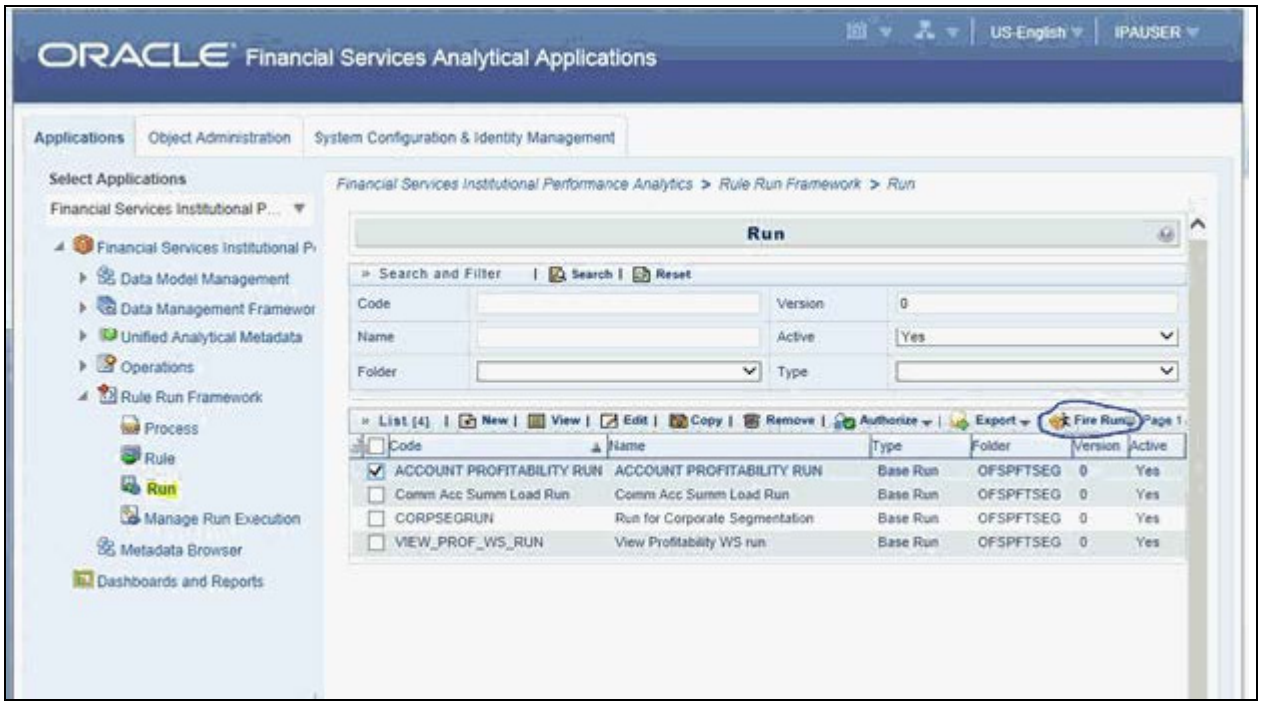
This batch execution loads the column N_RUN_SKEY in FCT_CRM_ACCOUNT_SUMMARY as -1. This will be a default run from the seeded batch. In order to be able to enter data for multiple runs, the batch tasks can be defined in Rule Run Framework. This will then create a batch internally which will load data for a different run into the fact table.

Executing a seeded run

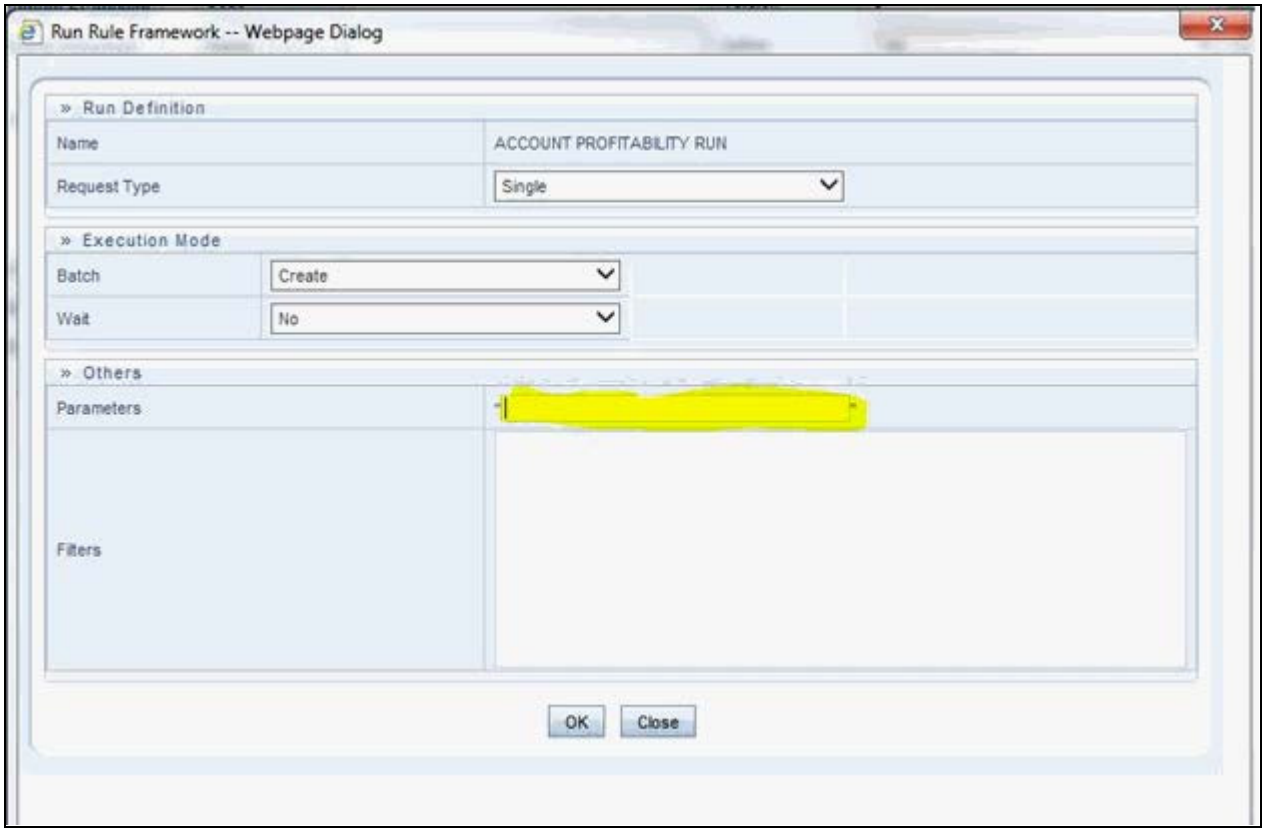
1. Navigate to**Rule Run Framework>Run**.



2. Choose a Run by checking the box before it and click **Fire Run**.

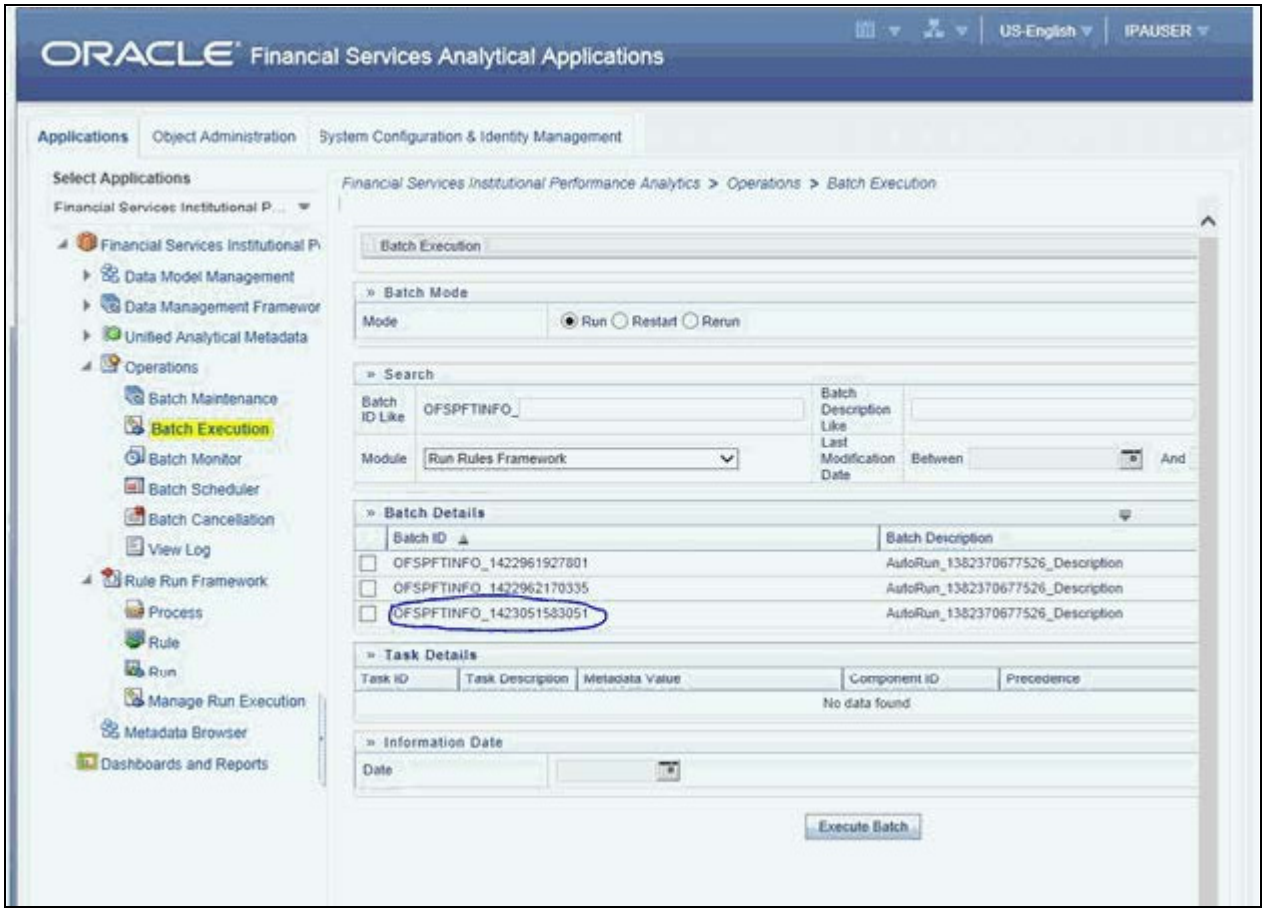


3. Enter the parameters required to execute the run (refer to details of individual runs) and click **OK**.



The following message will be displayed: *Fire run successful.*

4. Execute the batch by navigating to **Operations>Batch Execution** and select the latest batch created in the Run Rules Framework module.



Runs available for IPA

Following are the runs available for IPA:

1. ACCOUNT PROFITABILITY RUN – Run for loading FCT_CRM_ACCOUNT_SUMMARY and FCT_ACCOUNT_PROFITABILITY

   Parameters to be entered as follows:

   ##RCY##", "<INFODOM>", "FCT_ACCOUNT_PROFITABILITY

   Note: There are no quotes at the beginning and the end of the string. Consider as if the quotes are enclosing the delimiter (comma).

2. Comm Acc Summ Load Run – Run for loading FCT_COMMON_ACCOUNT_SUMMARY
3. VIEW_PROF_WS_RUN – Run for executing Web Service
4. RETSEGRUN – Run for executing Retail Segmentation Rule

APPENDIX G Loading Multiple Load Runs in OFSAA

This chapter discusses the following topics:

Overview

Multiple load run enables data to be loaded multiple times during the day for staggered processing of data by analytical applications. The degree of complexity of data required by analytical applications vary from one to the other, the load run ensures that the customer can process the data as soon as it is ready for an application to uptake. This reduces the turnaround time to reporting, by avoiding the 'end of day' type of processing of information as part of the original design.

Note: The load run is enabled only in the model and is defaulted to '0' in the model. This would not impact data previously available.

FSDF 8.0.3 staging model provides customers a flexibility to load multiple snapshots of the data in the staging tables (Product Processor's). A column named n_load_run_id was introduced as part of the primary key of the product processor tables to enable this. But the full fledged functionality to load and manage these snapshots will be part of the platform release at a later stage. Customers who would like to leverage this design in 8.0.3 release, the following mentioned changes will need to be done as a workaround to load multiple snapshot of data from staging to results tables such as Fact Common Account Summary.

For Loading Multiple Snapshots of Data for the same FIC_MIS_DATE , the existing T2T's need to be executed via Run Rule Framework and load needs to be filtered accordingly for each load run via the run filter. To enable execution of this run, navigate to $FIC_HOME/ficweb/webroot/conf and dit the file excludeURLList.cfg and add the following entry at the end of the file [SQLIA]./pr2

Note: There should not be any blank line in the file.

Features

Following are the features:

- To optimize the end-to-end data flow and the need for intra-day reporting, institutions could load intra-day records into OFSAA. Current application can only handle one set of records per date (incremental loads are not possible).
- Users need to adjust and reload data (either full or partial) for the current date.
- Users need to adjust and reload data (either full or partial) for any of past dates.

Design Details

Loading of data into OFSAA can be done in any of the following ways:

- ETL Tool
- OFSAA F2T
- OFSAA T2T
- OFSAA Excel upload
- OFSAA DIH

OFSAA data model includes load run identifier as part of the primary key for a set of staging tables. This enables data to be stored for multiple load runs for any date. OFSAA data model also has a table to maintain master information about load run and can be used for identifying/filtering load run during run execution within OFSAA.

OFSAA data model also another entity that tracks the load run mapped to the functional key of each of the staging table. Since OFSAA processing is on snapshot of data, this entity helps users to identify set of records that are latest to be used in processing. If there is a need to load multiple sets of data within a day, customers can use the below components to manage the same.

If set of data is snapshot

- Register table that got loaded in the load run through a DT (Register_load_run_details)
- Register table that got loaded in the load run through a DT (Register_load_run_details)
- Use the load run identifier to load data into OFSAA staging. You can use the same load run identifier for all the entities loaded in the same batch/group.
- Specify run-filter during OFSAA execution to filter records for the maximum run identifier within the day

If set of data is incremental

- Register load run information in master table through a DT (Register_Load_Run_Master)
- Register table that got loaded in the load run through a DT (Register_load_run_details)
- Register table that got loaded in the load run through a DT (Register_load_run_details)
- Execute another DT (Populate_Load_Run_Map) with different parameters for each entity to maintain the latest load run for each record.
- Modify the T2T's to join with the load run map tables for identifying and filtering on the latest set of records to be used in processing.

There could be some entities that can follow snapshot and some entities incremental load. Decision of snapshot vs incremental depends on the above use-case.

Data Transformations

Function - Register Load Run

Parameters - Batch ID, MIS-Date, Load Run Name, Load Run Purpose, Load Run Type

Steps

1. Check if the parameters are valid. Load run type can be 'B – Base, A – Adjustments, P – Backdated adjustments'.
2. Check uniqueness of load run name. Load run name is a user-specified string for easier retrievals could be MIS-DATE <Sequence> or the starting timestamp of load run.
3. Increment the load run id for a given MISDATE and insert the rest of the details.
4. Return the load run identifier (if possible, else user will query this table to get the load run id given a name)
5. Log messages accordingly
6. Return success/failure

Execution

Execute this DT before loading any fact for intra-day load. Use the registered load run identifier as a value to map to load run identifier field in staging. When one load run is ongoing (loading data to OFSAA using the load run identifier), do not register any new load runs. Else, make sure load run name is used as a filter instead of max load run identifier when querying the load run master table.

Function – Register Load Run Details

Parameters – batch id, mis-date, load run name, load run id, stage table name, load type

Steps:

1. Check if the parameters are valid. Load type can be 'S – Snapshot, I - Incremental'.
2. Load run name or ID can be provided. If load run name is provided, we can lookup into load run master for retrieving the ID. Check if table name exist.
3. Register the information in load run details table
4. Log messages accordingly
5. Return success/failure

Execution

Execute this DT after registering load run master and before loading any fact for intra-day load. When one load run is ongoing (loading data to OFSAA using the load run identifier), do not register any new load runs.

Function – Populate Load Run Map

Parameters – batch id, mis-date, and stage table name

Steps:

1. Check if the parameters are valid.
2. Pick the corresponding load run map table from a setup table. Pick the corresponding functional key columns and their mapping to load run map table from a setup table.

3. If record do not exist in load run map table for the functional key in staging, then insert a new record with the functional key and load run identifier.
4. If record exist in load run map table for the functional key in staging, then update latest record indicator for existing rows to 'N' and then insert a new record with the functional key and load run identifier.
5. This operation has to be done in bulk mode.
6. Log messages accordingly
7. Return success/failure

Execution

Execute this DT after loading any fact for intra-day load in case the table has incremental loads.

Multiple data runs can be done for the same date using one of the following two approach:

1. Using the Load Run ID, multiple data runs can be done for the same date. By default, the load run ID will be 0.
2. There will be multiple run_skeys generated for each extraction date. If there arefour sources, then 4 distinct run skeys are generated for the load date. Some of the out of box dashboards will show partial data as there is a filter for the Run. If all the sources need to be seen at a time, the processing should happen in a single run.
3. All the T2T's which are loading data into Fact CRM Account Summary has to be modified for ANSI Join conditions to include table DIM_DATA_ORIGIN
4. The Existing process "ACCOUNT PROFITABILITY PROCESS" is modified to have all the T2T's which are loading data into Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN_RUN_EXE_PARAM".
5. A new UMM regular BI Enabled hierarchy with only one level is created based on entity "DIM_DATA_ORIGIN".
6. New run definitions will be created for each data source which would be based on the same process "ACCOUNT PROFITABILITY PROCESS". Each run definitions would be set to have data source as the filter using the hierarchy defined earlier.

For example if there are 3 sources then,

- ACCOUNT PROFITABILITY Src1 with "Source1" as the filter
- ACCOUNT PROFITABILITY Src2 with "Source2" as the filter
- ACCOUNT PROFITABILITY Src3 with "Source3" as the filter

7. After defining the Run definitions, the run's are executed for the required MIS date. This approach requires as many run definitions for each of the sources which has to be processed at once. Each Run definition execution would be storing its Run Skey value in the target fact table.
8. Modify the existing DT "FCT_ACCT_TRANSFORMATION" for passing additional parameters to PL/SQL function "FN_FCT_ACCT_PFT_DT".
9. Grant Select privileges on few tables from Config schema user to atomic schema user.
10. Create Database views in atomic schema.

Following are the details of the approach:

1. All the T2T's which are loading data into Fact CRM Account Summary has to be modified for ANSI Join conditions to include table DIM_DATA_ORIGIN.

- Navigate to Database Extracts and to the required Application & Data Source.
- Select T2T T2T_STG_CRMAS_ANNUITY_CONTRACTS and Edit the definition.
- Modify the ANSI join condition by appending the following:

  LEFT OUTER JOIN DIM_DATA_ORIGIN ON

  DIM_DATA_ORIGIN.V_DATA_SOURCE_CODE =

  STG_ANNUITY_CONTRACTS.V_DATA_ORIGIN
- Save the T2T definition.
- Similarly do the changes for the other T2T's

  T2T_STG_CRMAS_BILLS_CONTRACTS

  T2T_STG_CRMAS_BORROWINGS

  T2T_STG_CRMAS_CARDS

  T2T_STG_CRMAS_CASA

  T2T_STG_CRMAS_INVESTMENTS

  T2T_STG_CRMAS_LC_CONTRACTS

  T2T_STG_CRMAS_LOAN_CONTRACTS

  T2T_STG_CRMAS_MM_CONTRACTS

  T2T_STG_CRMAS_OD_ACCOUNTS

  T2T_STG_CRMAS_TD_CONTRACTS

  T2T_STG_CRMAS_LEASES_CONTRACTS

  T2T_STG_CRMAS_GUARANTEES

  T2T_STG_CRMAS_TRUSTS

  T2T_STG_CRMAS_COMMITMENTS

  T2T_STG_CRMAS_MUTUAL_FUNDS

2. The Existing process "ACCOUNT PROFITABILITY PROCESS" is modified to have all the T2T's which are loading data into Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN_RUN_EXE_PARAM".
   - Edit the "ACCOUNT PROFITABILITY PROCESS".
   - Select the "Component" button.
   - Navigate to **Component>Insertion Rules><Source>** section from the LHS menu.
   - Select the T2T's which load data into Fact Common Account Summary.

     T2T_STG_ANNUITY_CONTRACTS_CAS

     T2T_STG_BILLS_CAS

     T2T_STG_BORROWINGS_CAS

     T2T_STG_CARDS_CAS

T2T_STG_CASA_CAS

T2T_STG_GUARANTEES_CAS

T2T_STG_INVESTMENTS_CAS

T2T_STG_LC_CAS

T2T_STG_LEASES_CONTRACTS_CAS

T2T_STG_LOANS_CAS

T2T_STG_MM_CAS

T2T_STG_OD_CAS

T2T_STG_TD_CONTRACTS_CAS

T2T_STG_TRUSTS_CAS

T2T_STG_COMMITMENT_CONTRACTS_CAS

T2T_STG_MUTUAL_FUNDS_CAS

- Select Precedence button and select FN_RUN_EXE_PARAM from the drop-down. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN_RUN_EXE_PARAM".

T2T_STG_ANNUITY_CONTRACTS_CAS

T2T_STG_BILLS_CAS

T2T_STG_BORROWINGS_CAS

T2T_STG_CARDS_CAS

T2T_STG_CASA_CAS

T2T_STG_GUARANTEES_CAS

T2T_STG_INVESTMENTS_CAS

T2T_STG_LC_CAS

T2T_STG_LEASES_CONTRACTS_CAS

T2T_STG_LOANS_CAS

T2T_STG_MM_CAS

T2T_STG_OD_CAS

T2T_STG_TD_CONTRACTS_CAS

T2T_STG_TRUSTS_CAS

T2T_STG_COMMITMENT_CONTRACTS_CAS

T2T_STG_MUTUAL_FUNDS_CAS

- Save the process definition as same version.

3. A new UMM regular BI Enabled hierarchy with only one level is created based on the entity "DIM_DATA_ORIGIN".

- Create a new hierarchy as mentioned in the following:
- After the hierarchy is defined, Authorize and save the metadata. Ensure data exists in DIM_DATA_ORIGIN before saving the metadata.

4. New run definitions will be created for each data source which would be based on the same process "ACCOUNT PROFITABILITY PROCESS". Each run definitions would be set to have data source as the filter using the hierarchy defined earlier.

For example if there are 3 sources then,

- ACCOUNT PROFITABILITY Src1 with "Source1" as the filter
- ACCOUNT PROFITABILITY Src2 with "Source2" as the filter
- ACCOUNT PROFITABILITY Src3 with "Source3" as the filter
- Create a new run definition say "ACCOUNT PROFITABILITY Source 1" as the run definition name.
- Code: AccountProfitabilitySrc1
- Name: Account Profitability Source 1

Type: Base Run

- Select **Add>Job**
- Select the process modified earlier from the LHS menu.
- Select **Add>Run Condition**
- Select Data Origin from the LHS menu.
- After selecting Run Condition and Job, select **Next**.
- Add the condition to Run Condition by selecting the "Launch Browser".
- Select the required source from the LHS menu.
- Select **Save**.
- Similarly, new run definitions have to be defined for each source that the user
- want to process, by repeating the above mentioned process.

5. Grant Select privileges on few tables from Config schema user to atomic schema user.

   pr2_run_object_member

   metadata_master

   metadata_element_master

   metadata_attribute_master

   metadata_locale_master

Execute the script "ConfigPrevsRunFilter.sql" by modifying the file. Replace the values ##ATOMIC_USER## with actual Atomic Schema user.

ConfigPrevsRunFilter.sql

```
GRANT SELECT ON pr2_run_object_member to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_master to ##
/
```

```
ATOMIC_USER##/GRANT SELECT ON metadata_element_master to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_attribute_master to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_locale_master to ##ATOMIC_USER##
/
```

6. Create Database views in atomic schema which are required.

Execute the script "Create_Run_Filter_Views.sql" by modifying it. Replace the values ##CONFIG_USER## with actual Config schema user and ##INFODOM## with the Infodom name.

Create_Run_Filter_Views.sql

```
CREATE OR REPLACE VIEW Vw_pr2_run_object_member AS SELECT * FROM
##CONFIG_USER##.pr2_run_object_member WHERE v_infodom_name =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_master AS SELECT * FROM
##CONFIG_USER##.metadata_master WHERE dsn_id = '##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_element_master AS SELECT * FROM
##CONFIG_USER##.metadata_element_master WHERE v_metadata_infodom =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_attribute_master AS SELECT * FROM
##CONFIG_USER##.metadata_attribute_master WHERE v_metadata_infodom =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_locale_master AS SELECT * FROM
##CONFIG_USER##.metadata_locale_master WHERE metadata_infodom =
'##INFODOM##'
/
```

7. Modify the PL/SQL function "`FN_FCT_ACCT_PFT_DT`." Execute the script " FN_FCT_ACCT_PFT_DT.sql" in atomic schema.
8. Modify the existing `DT "FN_FCT_ACCOUNT_PFT`" for passing additional parameters to PL/SQL function "FN_FCT_ACCT_PFT_DT"
   - Navigate to Post Load transformation screen.
   - Edit the definition "`FCT_ACCT_TRANSFORMATION`".
   - Navigate to Stored Procedure section.
   - Copy and paste the contents of the file "fn_fct_acct_pft.sql" into Stored Procedure Editor Section and save the definition.
   - Execute the Run's as required.

APPENDIX H PA Metrics Computation Logic

Projection Logic: We have data up to the last MIS Date based on which projections are done on Replines in FCT_ACCOUNT_VALUE_FORECAST. For each repline, for last MIS Date, there is a projection on every projected date (generally last date of every month)

PA Metrics calculation

Table 39. LIST OF APIs

Web Service	Input	Output	Version
Account Level Metrics (without UDM)	Account Number, Date	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income	IPA/RPA (8.0.5.0)
Customer Level Metrics (without UDM)	Customer Number, Date	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income	IPA/RPA (8.0.5.0)
Account Segmentation	Account Number, Date	Account Segment Code	IPA/RPA (8.0.5.0)
Customer Segmentation	Customer Number, Date	Customer Segment Code	IPA/RPA (8.0.5.0)

Note 1: Segment Metrics are done through batch process. The segment metrics considering the life time calculations of all the accounts in that segment.

Note 2: At Present Customer level segment metrics are based on the segment of first account identified (by the system) of the customer. The segment averages as computed against the identified account segment would be reported.

Data Treatment for Months

1. **Data to be considered for end of every month**
   - **Historical Data**: Take data for each month (as of month end date) from FCT_ACCOUNT_PROFITABILITY (aggregated under NIBT hierarchy)
   - **Projected Data**: For Last MIS Date, take data from FCT_ACCOUNT_VALUE_FORECAST for each subsequent Projected Date (aggregated under NIBT hierarchy: for aggregations we will consider all forecasted replines whether leaf or node)
   - Union the above two queries so that we get a data set from first month end date to last month end date. This will be inclusive of both actual + forecasted.
   - Discount each row in this dataset to the account start date; N_Origination_date
   - Sum up all the discounted values
   - Arrive at the value of NIBT

   Note: Discounting: For projection purpose, to arrive at the Net Present Value (NPV) of the future stream of NIBT; a discount factor be applied to arrive at expected value of NIBT.

2. **Treatment of Data of Month End Values and for Missing Months**

   - **Actual Data with Missing Value**: For profitability calculations or reporting the missing value should be considered as 0.
   - **For Forecasting of data**: If there is a missing value in the actual data then, the missing value treatment should be applied only to fill in a value which can be used for forecasting.

3. **Calculation of Profitability Metrics**

   - **Return on Total Assets (ROTA) =**
     - Loan Products- NIBT/ Sanctioned Limit (For Loans);
     - Line of Credit Products - NIBT/ Max of EOP Balance for Line of Credits
   - **Risk-adjusted return on capital (RAROC) = NIBT/ Sum of Unexpected Losses**
     - Unexpected Losses comprise of- Operating Risk Capital - Economic, Credit Risk Capital - Economic, Market Risk Capital - Economic.
   - **Return on Equities (ROE):**
     - Loan Products- NIBT/ Sanctioned Limit (For Loans); [N_SANCTIONED_LIMIT for Loans (first MIS Date), FCAS]
     - Credit Products -
       - NIBT/ Max of EOP Balance for Line of Credits [max(N_EOP_BAL), FCAS for Line of Credit (across all MIS Dates)]
       - Average of EOP Balance for all deposits (including term deposits) [average(N_EOP_BAL), FCAS for all deposits (including term deposits)]
   - **TOTAL REVENUE** = NET INTEREST INCOME + NON-INTEREST INCOME [Non-Interest Revenue' + 'Indirect Non-Interest Income' + 'Other Revenue]
   - **Total Expenses**

     Total Operating and Non-Operating Expenses [Net Credit Losses = Credit Losses- Recoveries of amounts previously written-off)
   - **Net Income**: Net Income Before Tax
   - **Currency**

     Metrics would be reported in reporting Currency.

     Note: As per the current functionality, by default, is there is no requested currency code, the customer would be showing in reporting currency and the Accounts are shown in it's natural currency. In case of any requested currency code, both customer and currency would be shown in requested currency code.
   - **Customer Metrics**:
     - ROTA and RAROC would be aggregated only against asset products.
     - All other metrics would be aggregated against all products.
     - Customer Metrics would be reported in Reporting Currency.

- **Customer Level Metrics Calculation**:
  - ROTA: (Sum of ROTA Numerator over all asset accounts)/(sum of ROTA denominator over all asset accounts)
  - RAROC: (Sum of RAROC Numerator over all asset accounts)/(sum of RAROC denominator over all asset accounts)
  - ROE: (Sum of ROE numerator over all accounts)/(Sum of ROE denominator over all accounts)
  - TOTAL INCOME, TOTAL EXPENSES, NET INCOME: Sum over all accounts
- **Segment Metrics**:
  - Segment Level Metrics will be reported against Segmentation done at Account Level
  - Segment Metrics to be reported in Reporting Currency.

APPENDIX I - *Web Service Usage*

The web service implemented in OFS IPA application uses Rest API. This service has request and response. This web service is used for fetching the following metrics of an account:

- ROTA _Return on Total Assets
- RAROC- Risk Adjusted Return On Capital
- ROE - Return on Equity
- Total Income
- Total Expense
- Net income

When the demographic details of the customer are entered as input, this web service returns the segment in which the customer falls in to.

If there are diverse details that do not match with any of the segment criteria, then this web service does not return any matching segment details and displays "Segment not found" error.

Using the Web Service

The URL for the web service should be appended by the following string:

/rest-api/pa/v1/metric/post

For example:

http://whf00anq:3464/ofsa/rest-api/pa/v1/metric/post

Before running the web service, open the JSON file in a suitable editor and ensure that you enter the following values:

- **Authorization**: Provide the credentials for OFSAA users with IPA BI Analyst role.
- **UserId**: OFSAA User (for example, pftuser)

Note: CUSTOMER_TYPE and SEGMENT_TYPE attributes are mandatory attributes.

REST web service is automatically available after successful installation of the application. The name of the contract is PACS_Request.json and PACS_Response.json. We recommend going through the contract thoroughly before accessing the web service.

Table 40. Web Service Values

Attribute Name	Datatype	Description and Acceptable Values
SERVICE_TYPE	String	`PERFORMANCE_METRICS` (for retrieving metrics for an existing account) Or `SEGMENTATION` (for retrieving segment of an existing account)
OPERATION_TYPE	String	`ACCOUNT` (hard coded for future provision)
ACTION_TYPE	String	`LOOKUP` (hard coded for future provision)
REQUEST_NUMBER	Number	A unique integer number. Should change each time.

PARAMS	String	`"PARAMS": [`
		`{`
		`"PARAM_CODE": "INFODOM",`
		`"PARAM_VALUE": "<VALUE OF INFODOM>"`
		`},`
		`{`
		`"PARAM_CODE": "FOLDER",`
		`"PARAM_VALUE": "<VALUE OF FOLDER>"`
		`}`
		`]`
		
		Note: The values for FOLDER is not currently used. It is recommended to use DEFAULT for this parameter.

PAYLOAD->PARTY->PARTY_ID	String	Customer ID **Note**: When PARTY_ID is not passed, it should be passed as null. Web service will not accept blank value for this parameter.

PAYLOAD -> PARTY -> PARTY_ATTRIBUTES	Repeating array of all attributes when retrieving segmentation. Example is given in the next column.	`{`
		`"ATTRIBUTE_NAME": "GENDER",`
		`"ATTRIBUTE_VALUE": "MALE"`
		`}`
		
		Note: This is not used. Give 'NULL'.

PAYLOAD -> ACCOUNTS -> ACCOUNT_NUMBER	String	Account ID **Note**: When ACCOUNT_NUMBER is not passed, it should be passed as null. Web service will not accept blank value for this parameter.

Table 40. Web Service Values

Attribute Name	Datatype	Description and Acceptable Values

PAYLOAD -> EXEC_PARAMS	String	{
		"PARAM_CODE": "AS_OF_DATE",
		"PARAM_VALUE": "<Provide the date on which you want the metrics in YYYYMMDD format>"
		},
		{
		"PARAM_CODE": "RPT_CCY_CODE",
		"PARAM_VALUE": "<Provide the ISO currency code"
		}

PAYLOAD -> ACCOUNTS -> ACCOUNT_ATTRIBUTES	Repeating array of all attributes when retrieving segmentation. Example is given in the next column	{
		{"ATTRIBUTE_NAME":"CUSTOMER_TYPE","ATTRIBUTE_VALUE":"R"},
		{"ATTRIBUTE_NAME":"SEGMENT_TYPE","ATTRIBUTE_VALUE":"D"}
		
		}
		
		CUSTOMER_TYPE: Acceptable values are:
		• I: Institutional
		• R: Retail
		
		SEGMENT_TYPE: Acceptable values are:
		• D: Demographic
		• P: Profitability
		• B: Behavioral
		• R: Risk
		• C: Corporate

The following attributes are optional attributes:

- V_PROD_CODE
- N_AGE
- V_GENDER
- V_MARITAL_STATUS
- V_INDUSTRY
- V_COUNTRY
- N_INCOME
- N_NIBT
- N_LTV
- N_EOP_BAL
- N_ENHANCED_LTV

- F_UPGRADE_IND
- N_TOTAL_TRANSACTIONS
- N_AVERAGE_TRANSACTION_SIZE
- N_CUSTOMER_MOB
- N_LIMIT_UTILIZATION
- N_CREDIT_SCORE
- N_MITIGANT_VALUE
- N_DELINQUENT_DAYS
- N_LOAN_TO_VALUE
- V_PRIMARY_OR_SECONDARY_CUST
- N_YEAR_OF_INCORPORATION
- N_CUSTOMER_INCOME
- N_TOTAL_ASSETS
- F_CUSTOMER_LISTED_FLG
- N_EMPLOYEES
- N_TOTAL_ACCOUNTS
- N_RAROC
- N_ROTA
- V_CREDIT_RATING_CODE
- N_DEBT_COVERAGE_RATIO
- N_INTEREST_COVERAGE_RATIO
-

The response will be available in the PACS_Response.json file in case of success response Status = 200.

If there are any incorrect inputs, then the response status would be 400 and json will not be returned. If there is a server error response status would be 500 and no response json will be returned.