Oracle Financial Services Retail Performance Analytics

User Guide

Release 8.1.1.0.0

Jun 2021

F33064-01

ORACLE®
Financial Services

ORACLE®

OFS Retail Performance Analytics User Guide

Copyright © 2021 Oracle and/or its affiliates. All rights reserved.

Document Control

Version Number	Revision Date	Change Log
1.0	April 2021	Created the OFS RPA user guide for Release 8.1.1.0.0.

Table of Contents

1 Preface

This section provides information about the Oracle Financial Services Retail Performance Analytics (OFS RPA) Application User Guide. OFS RPA application is packaged as part of the OFS PFT Application Pack.

Topics:

- Audience
- Documentation Accessibility
- Access to Oracle Support
- Structure
- Conventions
- Abbreviations
- Related Information Sources
- Additional Documents to Read

1.1 Audience

This user guide is intended for the users of the Oracle Financial Services Retail Performance Analytics (OFS RPA) application.

See Related Information Sources and Additional Documents to Read for more Oracle product information.

1.2 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

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

1.3 Structure

This user guide has been segregated into the following chapters:

- Introduction
- Overview of Process Flow
- Dimension Loading Process
- Time Dimension Population
- Customer Dimension Population
- Account Dimension Population

- Exchange Rate History Population

- Account Summary Population

- Fact Transaction Summary

- Customer Summary Population

- Fact Data Population

- Cube Build Process

- Time Series Forecasting

- Segmentation

- Overview of OFS RPA Reports

- What-If Analysis

- Visibility

- Appendix A: Adding a Dimension

- Appendix B: Adding a New Measure

- Appendix C: Developing a New Cube

- Appendix D: Defining a Batch

- Appendix E: List of Hardcoded Members

- Appendix F: Run Rule Framework

- Appendix G: Loading Multiple Load Runs of OFSAA

- Appendix H: PA Metrxs Computation Logic

1.4 Conventions

The following text conventions are used in this document:

Table 1: Document Conventions

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, file names, text that appears on the screen, or text that you enter.
Hyperlink	Hyperlink type indicates the links to external websites, internal document links to sections.

1.5   Abbreviations

The following table lists the abbreviations used in this document:

Table 2: Abbreviations

Abbreviation	Meaning
AIX	Advanced Interactive eXecutive
BDP	Big Data Processing
DBA	Database Administrator
DDL	Data Definition Language
DEFQ	Data Entry Forms and Queries
DML	Data Manipulation Language
EAR	Enterprise Archive
EJB	Enterprise JavaBean
ERM	Enterprise Resource Management
ETL	Extract, Transform, Load
FTP	File Transfer Protocol
HDFS	Hadoop Distributed File System
HTTPS	Hypertext Transfer Protocol Secure
J2C	J2EE Connector
J2EE	Java 2 Enterprise Edition
JCE	Java Cryptography Extension
JDBC	Java Database Connectivity
JDK	Java Development Kit
JNDI	Java Naming and Directory Interface
JRE	Java Runtime Environment
JVM	Java Virtual Machine
LDAP	Lightweight Directory Access Protocol
LHS	Left Hand Side
MFA	Multi-Factor Authentication
MOS	My Oracle Support
OFSAA	Oracle Financial Services Analytical Applications
OFSAAI	Oracle Financial Services Analytical Application Infrastructure
OFSAAAI	Oracle Financial Services Advanced Analytical Applications Infrastructure Application Pack
OHC	Oracle Help Center

Abbreviation	Meaning
OLAP	On-Line Analytical Processing
OLH	Oracle Loader for Hadoop
ORAAH	Oracle R Advanced Analytics for Hadoop
OS	Operating System
RAM	Random Access Memory
RDBMS	Relational Database Management System
RHEL	Red Hat Enterprise Linux
SCD	Slowly Changing Dimension
SFTP	Secure File Transfer Protocol
SID	System Identifier
SSL	Secure Sockets Layer
TNS	Transparent Network Substrate
URL	Uniform Resource Locator
VM	Virtual Machine
WAR	Web Archive
XML	Extensible Markup Language

1.6 Related Information Sources

This section identifies additional documents related to the OFS IPARPA Application. You can access the below documents online from the Oracle Help Center (OHC) Documentation Library for OFS PFT Pack Release 8.1.0.0.0:

OHC Documentation Library for the Oracle Financial Services Profitability Management Applications Pack (OFS PFT Pack) Release 8.1.0.0.0:

Installation Guide:

- OFS Performance Analytics Applications Pack Installation and Configuration Guide, Release 8.1.1.0.0

User Guides:

- Oracle Financial Services Institutional Performance Analytics User Guide Release 8.1.1.0.0

- Oracle Financial Services Retail Performance Analytics User Guide Release 8.1.1.0.0

- Oracle Financial Services Enterprise Financial Performance Analytics User Guide Release 8.1.1.0.0

Oracle Financial Services Performance Analytics Applications Pack Security Guides:

- Oracle Financial Services Institutional Performance Analytics Security Guide Release 8.1.0.0.0

- Oracle Financial Services Retail Performance Analytics Security Guide Release 8.1.0.0.0

- [Oracle Financial Services Enterprise Financial Performance Analytics Security Guide Release 8.1.0.0.0](#)

Oracle Financial Services Performance Analytics Applications Pack Cloning Reference Guides:

- [Oracle Financial Services Institutional Performance Analytics Cloning Reference Guide Release 8.1.0.0.0](#)

- [Oracle Financial Services Retail Performance Analytics Cloning Reference Guide Release 8.1.0.0.0](#)

- [Oracle Financial Services Enterprise Financial Performance Analytics Cloning Reference Guide Release 8.1.0.0.0](#)

1.7 Additional Documents to Read

Oracle Financial Services Profitability Management Applications Pack is built on the Oracle Financial Services Advanced Analytical Applications Infrastructure (OFS AAI). See the following OFS AAI documents as no separate documents are required at the pack or application level for Oracle Financial Services Profitability Management Applications Pack:

- [OFS Analytical Applications Infrastructure (OFS AAAI) Application Pack Installation and Configuration Guide Release 8.1.1.0.0](#)

- [OFS Analytical Applications Infrastructure Administration Guide Release 8.1.0.0.0](#)

- [OFS Analytical Applications Infrastructure Cloning Reference Guide Release 8.1.0.0.0](#)

- [OFS Analytical Applications Infrastructure Security Guide Release 8.1.0.0.0](#)

- [OFS Analytical Applications Infrastructure User Guide Release 8.1.0.0.0](#)

You can access the common document from the OHC Documentation Library:

- [OFS Analytical Applications 8.1.0.0.0 Technology Matrix](#)

2 Overview of OFS Retail Performance Analytics (OFS RPA)

This guide explains the concepts of Oracle Financial Services Retail Performance Analytics and provides step-by-step instructions for navigating the Retail Performance Analytics user interface. Oracle Financial Services Retail Performance Analytics (OFS RPA) is a complete end-to-end web-based Business Intelligence solution that provides a 360-degree view of the customer relationship for key insights into the customer life-cycle.

OFS RPA 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 also provides you deep insights into customer engagements across target segments and products/Line Of Business (LOB) including lending, credit cards, and so on. It proactively manages the growth through strategic insights into the retail business performance. OFS RPA helps you to monitor customer distribution across credit and delinquency bands and related exposures.

The OFSRPA solution is a part of the Performance Analytics Pack and is packaged along with AAI and other applications. This OFS RPA is supported for Oracle 18c and 19c.

OFSRPA solution is built using:

- OBIEE for Dashboard and Reports activities
- Essbase for cubes

This manual deals with essential Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) required for OFS RPA activities, process flow for the data transformation, 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 the OBIEE Answers tool.

2.1 OFS Retail Performance Analytics (OFS RPA Run Chart)

Run chart, as the name suggests helps the consultant understand the sequence of tasks to be performed to ensure the correctness of data flow in the product. It contains the details about Data loading for all mandatory tables for the product and the sequence of execution of seeded tasks/batches.

2.1.1 Verification of Exchange Rates T2T

To verify that a join in the T2T FSI EXCHANGE RATES has come correct as part of the installer in a pack-on-pack scenario, and if need be, then change it.

To change the ANSIJOIN, follow these steps:

1. Navigate to the Data Management Framework, select **Data Management Tools**, and then select **Data Mapping** to display the **Data Mapping** screen.

Figure 1: Data Mapping screen
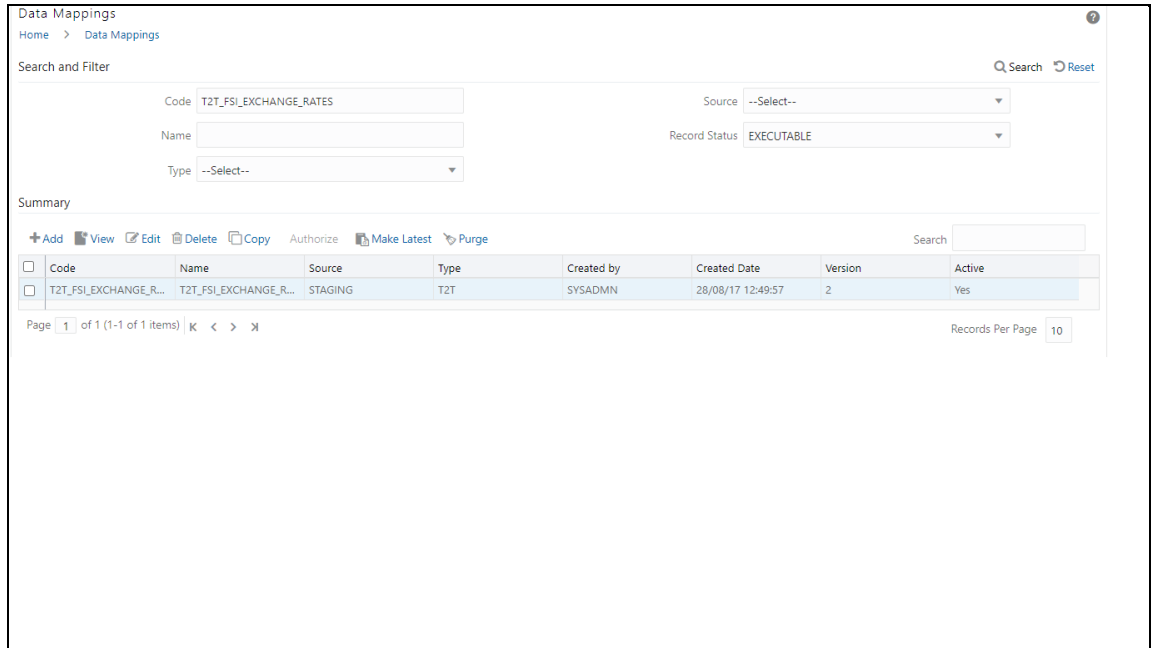
2. Edit the T2T definition.

   The ANSIJOIN part should be **VW_FSI_RATE_TRIANGULATION INNER JOIN DIM_RUN ON DIM_RUN.N_RUN_SKEY = -1**

3. If the RHS part of the equation is **$RUNSK** then RHS should be changed to **-1**.

Figure 2: Mapping Details screen
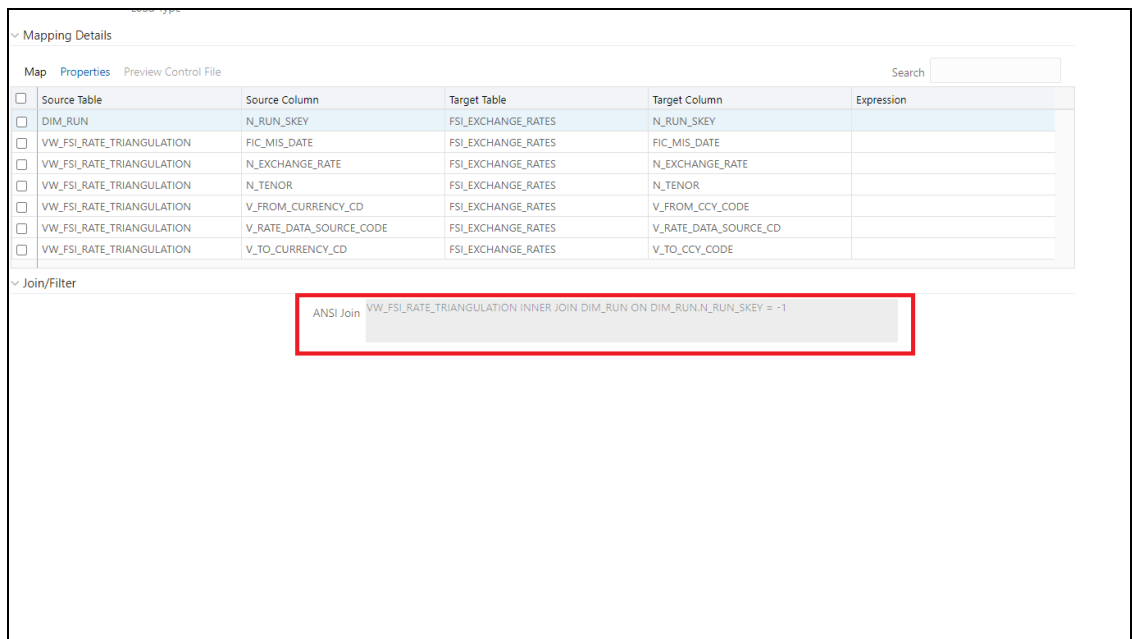
4. Click Map to modify the ANSIJOIN parameter.

Figure 3: Mapping
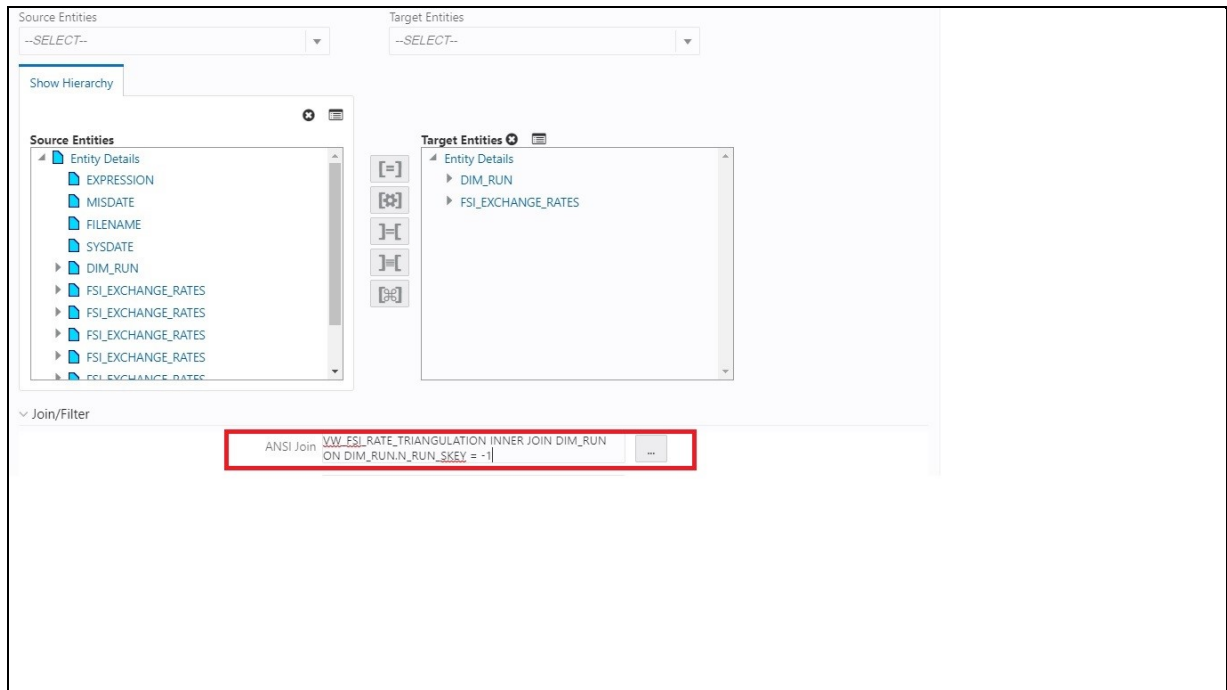
5. Change the **VW_FSI_RATE_TRIANGULATION INNER JOIN DIM_RUN ON DIM_RUN.N_RUN_SKEY** to **-1** if required.

6. Click **OK**.

7. Click **Save**.

Oracle Financial Services Retail Performance Analytics (OFS RPA) provides the Run Chart listing the tasks required for the population of data for OFS RPA Reports. This covers the following tasks:

- Stage Dimension Load

- Common data Population (for example Exchange Rate)

- Set up table population

  - Dimensions

  - Facts

- Profitability tables Population

- Model Execution

- Derived Entity Refresh

- Cubes Build

Download the *OFS RPA Cloud Service 8.1.1.0.0 Run Chart* from the MOS page.

3   Overview of OFS RPA Process Flow

Oracle Financial Services Retail Performance Analytics (OFS RPA) utilizes OBIEE technology to:

- Gain deep insight into customer engagements across target segments and products or LOB including lending, credit cards, and so on.

- Perform Wallet share analysis and Customer Profitability.

- Understand the efficiency of investments (like marketing, branch, and channel, and so on) over time.

- Monitor customer distribution across credit and delinquency bands and related exposures.

- Perform an enterprise-wide revenue analysis across customer segments, products, and reporting lines including fee income, interest, and interchange.

- Summary performance of the LOBs, overall Profitability, and Portfolio mix.

- Customer trends across performance drivers like Sales, Balances, Deposits, Product subscriptions (revenue services), Credit scores, Delinquency bands, Losses, and so on.

- LOB-specific performance reports can be analyzed against key dimensions like customer segments, product family, region, branch, risk scores, and so on.

- Analyze expenses across customer segments, products, and channels to understand ROI.

Figure 4: Objectives of RPA
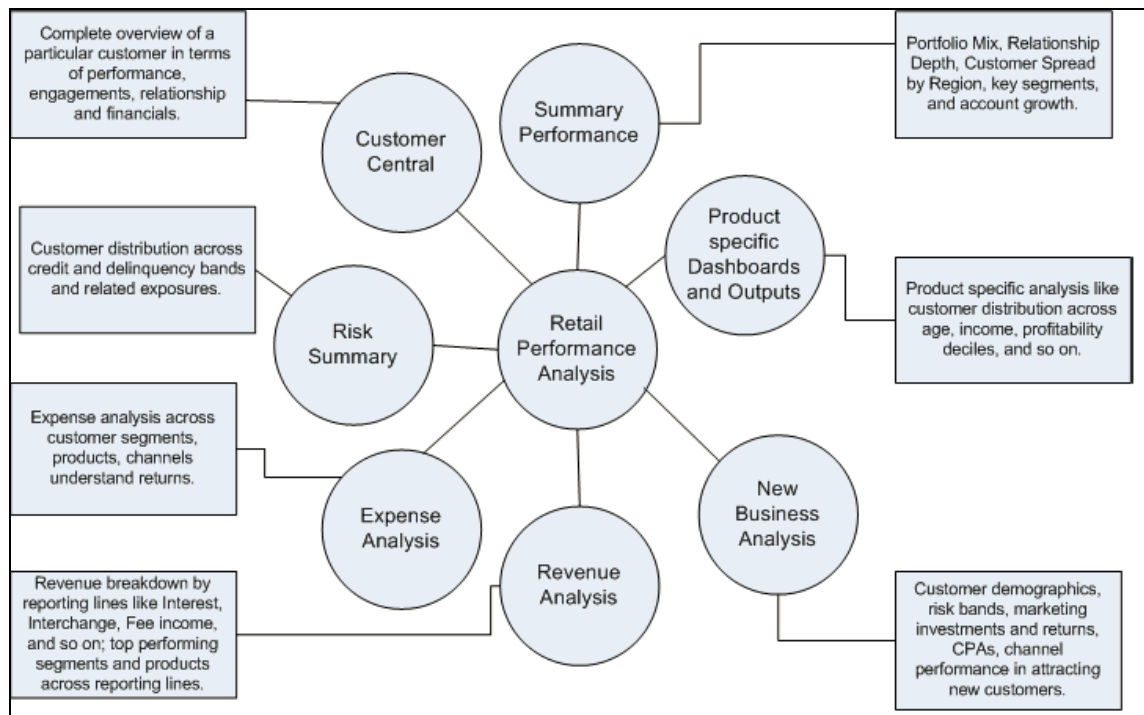
For details on OFSRPA reports and how OBIEE is being utilized, see Overview of OFSRPA Reports.

OFSRPA is designed for OBIEE reading data from a relational database. The relational database comprises of various dimensions and facts in the BI data model.

OFSRPA can be independently licensed and installed to work on top of the OFSAA Infrastructure.

3.1 Data Flow

Retail Performance Analytics data model contains the staging tables from which data is loaded into 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 5: Staging Tables
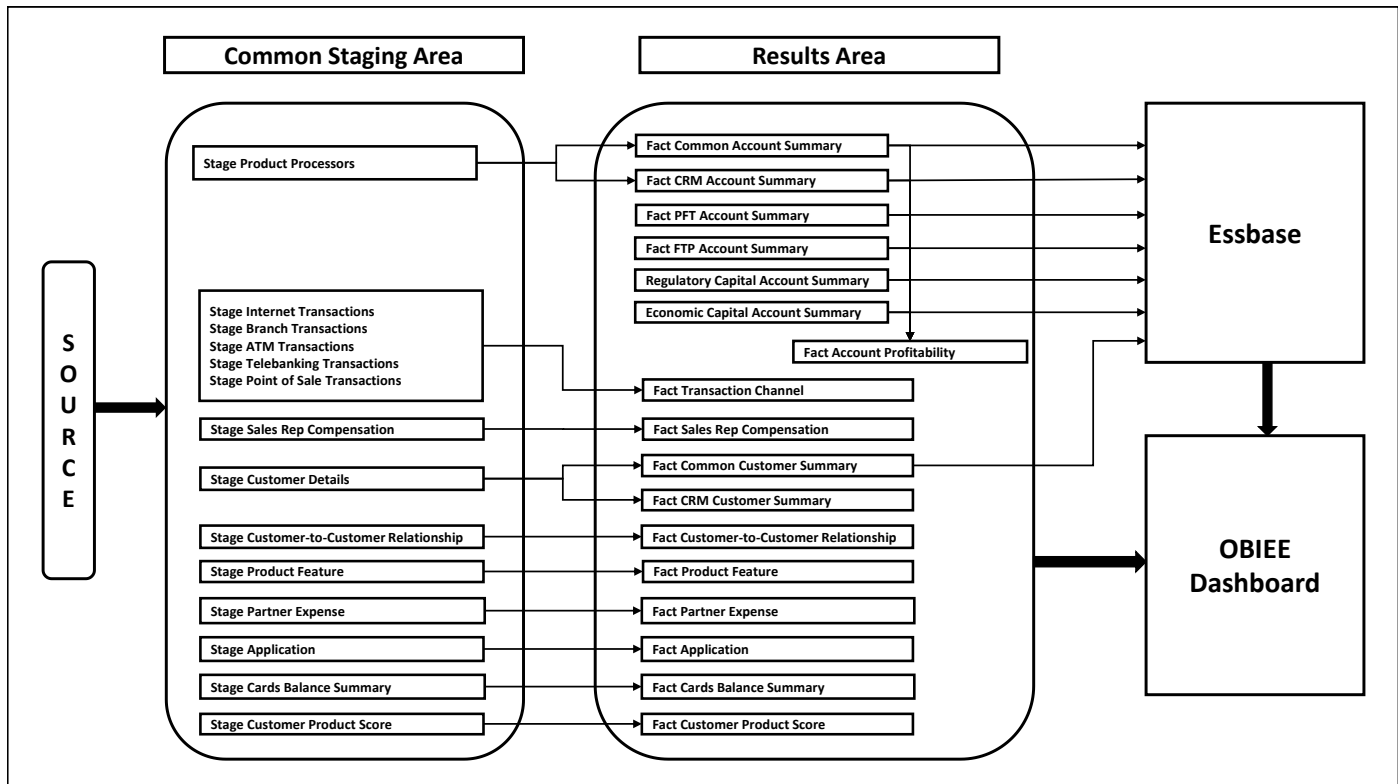
3.2 Dimension Data Flow

Dimension Data in the OFS RPA 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, Campaign, and so on.

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

Following is the list of dimensions used in OFSRPA.

Table 3: OFSRPA Dimensions

Dimension Entity Name	Staging Entity Name (or Names)	Loading or Maintenance method
Account Status Dimension	Stage Account Status Master	SCD
Application Reject Reasons Dimension	Stage Application Reject Reason Master	SCD
Application Status Dimension	Stage Application Status Master	SCD
Application Type Dimension	Stage Application Type Master	SCD
Attrition Dimension	Stage Attrition Reason Master	SCD
Authorization Decision Reasons Dimension	Stage Auth Decision Reason Master	SCD
Balance Category Dimension	Stage Credit Card Balance Category Master	SCD
Card Type Dimension	Stage Card Type Master	SCD
Channel Transaction Dimension	Stage Transaction Channel Type 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	SCD
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
Home Ownership Dimension	Stage Home Ownership Master	SCD
Household Dimension	Stage Household Master	SCD
Industry Dimension	Stage Industry Master	SCD
LoB Dimension	Stage LOB Master	SCD
Management Dimension	Stage Account Mgmt Master	SCD
Merchant Dimension	Stage Merchant Master	SCD
Merchant Category Dimension	Stage Merchant Category Master	SCD
Migration Reasons Dimension	Stage Migration Reason Master	SCD
Offer Dimension	Stage Offer Master	SCD
Reason Dimension	Stage Opportunity Win Loss Reason Master	SCD
Organization Structure Dimension	Stage Organization Structure Dimension	SCD
Partner Dimension	Stage Partner Master	SCD
Pool Identification Dimension	Stage Pool Identification Master	SCD
Prepayment Reason Dimension	Stage Prepayment Reason Master	SCD

Dimension Entity Name	Staging Entity Name (or Names)	Loading or Maintenance method
Product Dimension	Stage Product Master	SCD
Loan Product Category Dimension	Stage Product Category Master	SCD
Product Feature Dimension	Stage Product Feature Master	SCD
Product Type Dimension	Stage Product Type Master	SCD
Prospect Dimension	Stage Prospect Master	SCD
Retention Offer Type Dimension	Stage Retention Offer Master	SCD
Sales Representative Dimension	Stage Sales Rep Master	SCD
Sales Stage Dimension	Stage Sales Stage Master	SCD
Terminal Dimension	Stage Terminal Master	SCD
Terminal Type Dimension	Stage Terminal Type Master	SCD
Transaction Dimension	Stage Transaction Master	SCD
Transaction Channel Dimension	Stage TXN Channel Master	SCD
Txn Failure Reason Dimension	Stage Transactions Failure Reason Master	SCD
Transaction Status Dimension	Stage Transactions Status Master	SCD
Vendor Dimension	Stage Vendor Master	SCD
Vintage Dimension	Stage Vintage Master	SCD
Reporting Line Dimension	Reporting Line Dimension Members, Reporting Line Member Translation, Reporting Line Member Attributes, Reporting Line Hierarchies	AMHM/DT
Band Dimension	Band Dimension Members, Band Member Translation, Band Member Attributes	AMHM/SCD
Account Dimension	Stage LC Contracts	SCD
Account Dimension	Stage Commitment Contracts	SCD
Party Dimension	Stage Party	SCD
Account Dimension	Stage OD Accounts	SCD
	Stage TD Contracts	SCD
	Stage Trusts	SCD
	Stage Loan Contracts	SCD
	Stage Mutual Funds	SCD
	Stage Bills Contracts	SCD
	Stage CASA Accounts	SCD
	Stage Guarantees	SCD
	Stage Leases Contracts	SCD
	Stage MM Contract	SCD

Dimension Entity Name	Staging Entity Name (or Names)	Loading or Maintenance method
	Stage Annuity Contracts	SCD
	Stage Borrowings	SCD
	Stage Card Account	SCD
	Stage Investments	SCD
Region Dimension		Direct Load
Acquisition Channel Dimension		Direct Load
Instrument Category Dimension		Seeded
Currency Dimension		Seeded
Gender Dimension		Seeded
Marital Status Dimension		Seeded
Calendar Dimension		DT
Account Dimension	Staging Product Processor Tables like: - 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 Swaps Contracts - Stage Repo Contracts - Stage Option Contracts - Stage Mutual Funds - Stage Futures And Forwards	SCD

Some of the stage data can also come from master data management interfaces. In such cases, data from the 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 tables in the atomic schema.

3.2.1 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 CRM Account Summary, and so on.

Some of the Fact tables are loaded with processed fact information from other fact tables. Examples include Fact CRM Customer Summary, and so on.

Table 4: 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 Contracts Stage Trusts	T2T
Fact CRM Account Summary	Stage	Stage Commitment Contracts Stage Mutual Funds 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	T2T

Fact Entity Name	Source	Source Entities	Method of populating measures
Fact Common Customer Summary	Stage	Stage Commitment Contracts Stage Mutual Funds 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 Fact Transaction Channel	T2T/DT
Fact Application	Stage	Stage Applications	T2T
Transaction Channel	Stage	Stage Internet Transactions Stage Branch Transactions Stage ATM Transactions Stage Telebanking Transaction Stage Point Of Sale Transactions	T2T
Fact Cards Balance Summary	Stage	Stage Credit Card Balance 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 Account Profitability	Fact	Fact Common Account Summary Fact FTP Account Summary Fact PFT Account Summary	DT
Exchange Rate History	Stage	Stage Exchange Rates	T2T
Fact Account Segment Score	Fact	Fact Common Account Summary	T2T
Fact Account Segment MOB Summary	Fact	Fact Account, Fact Profitability, Fact Common Account Summary, Fact Account Segment Score	DT
Fact Party Account Role Map	Stage	Stage Party Account Role Map	T2T
Fact Party Financials	Stage	Stage Party Financials	T2T

This table will be populated in presence of the RCA application with the T2T. In absence of RCA, this table has to be directly loaded if the data is available.

The OFSRPA uses some materialized views registered as *Derived Entity*, that must 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.

Table 5: Derived Entities and Dependent Objects

MATERIALIZED_VIEW	REFERENCED_NAME	REFERENCED_OBJECT_NAME
ACNTSMRM	FCT_COMMON_ACCOUNT_SUMMARY	Table
	FCT_CRM_ACCOUNT_SUMMARY	Table
CUSTDETM	DIM_CUSTOMER	Table
	DIM_CUSTOMER_TYPE	Table
	DIM_GENDER	Table
	FCT_COMMON_CUSTOME R_SUMMARY	Table
FCSTCUSR	VW_ACCT_VAL_FCST_CUSTAGG_RPA	Table
FCSTLTMR	VW_FORECAST_LTV_RPA	Table
FCSTREPR	VW_ACCT_VAL_FCST_REP AGG_RPA	Table
FSIUSRD	FSI_USER_DATA_ACCESS	Table
MGMTPFTM	ACNTSMRM	Table
	CUSTDETM	Table
	FCT_ACCOUNT_MGR_REL	Table
	FCT_ACCOUNT_PROFITABILITY	Table
MVRACPRO	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
MVRCUSAG	A_DIM_REP_CURRENCY	Table
	DIM_ACCOUNT	Table

MATERIALIZED_VIEW	REFERENCED_NAME	REFERENCED_OBJECT_NAME
	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
MVRPROAG	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
	MVUSRACC	Table
MVUSRACC	DIM_ACCOUNT	Table
	FCT_COMMON_ACCOUNT_SUMMARY	Table
	FSIUSRD	Table
RTHREPMV	WITH_REP_LINE_DIRECT_IND_RPA	Table

MATERIALIZED_VIEW	REFERENCED_NAME	REFERENCED_OBJECT_NAME
USRMGRMV	FSI_M_USER_MANAGER_MAP	Table

3.3 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 the erwin data model:

Figure 6: Fact Account Feature Map
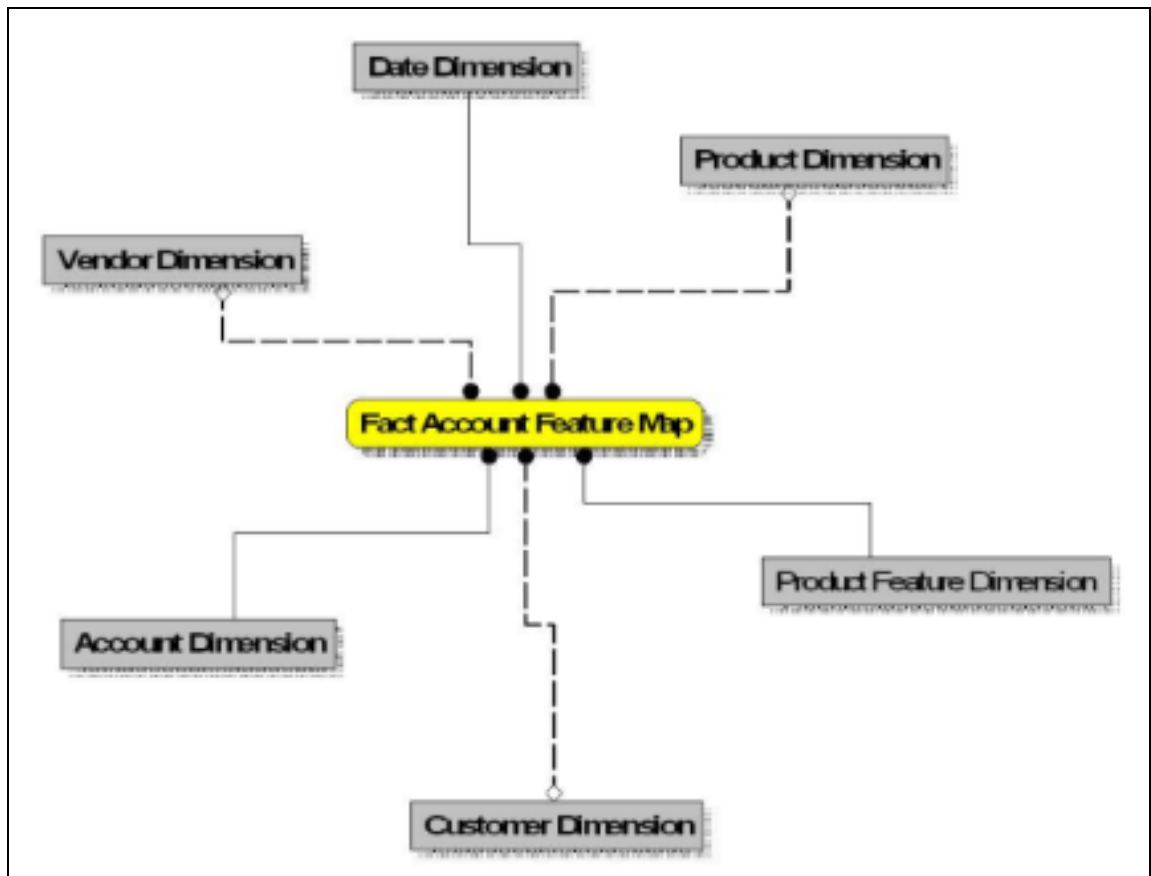
Figure 7: Fact Account Manager Relationship
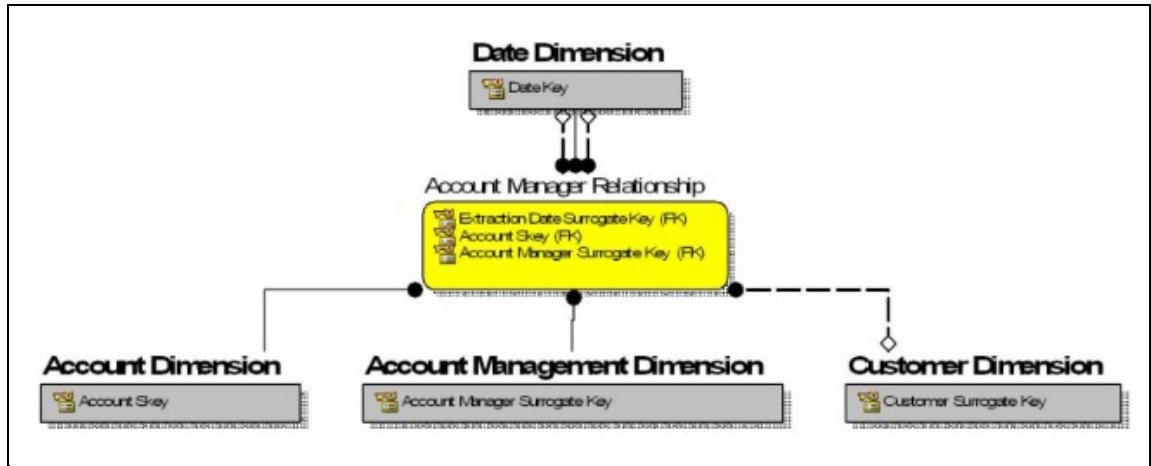
Figure 8: Fact Account Party Role
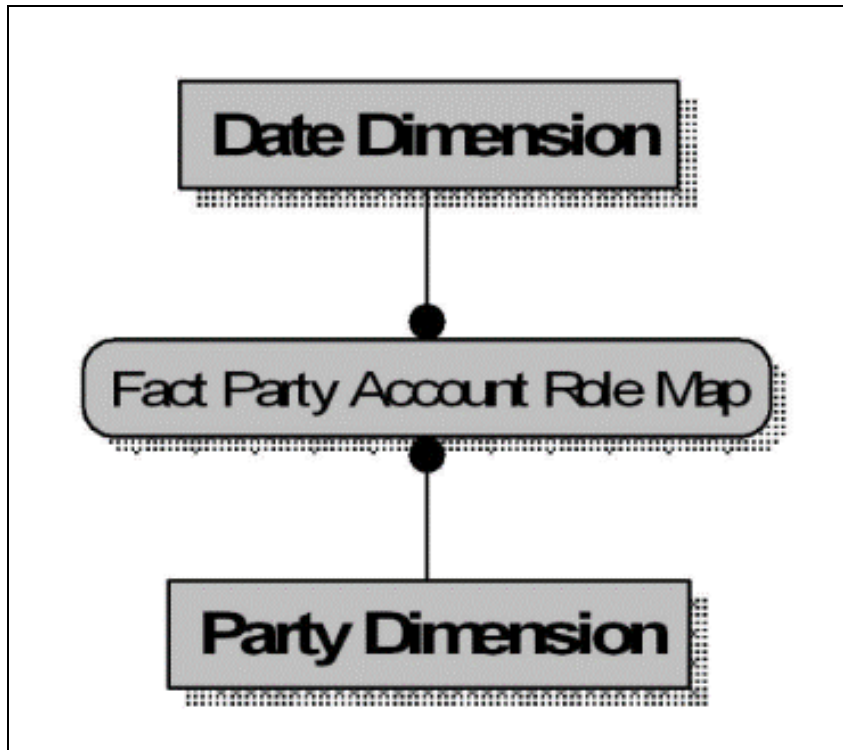
Figure 9: Fact Account Profitability
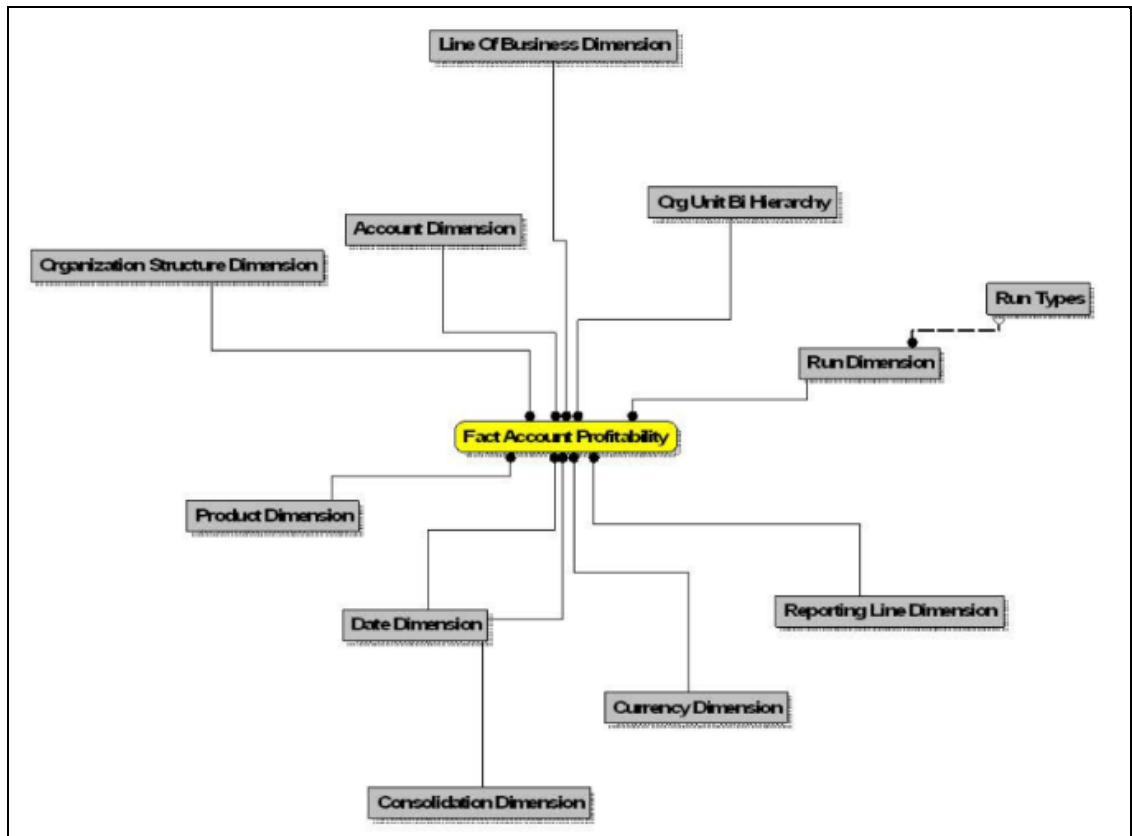
Figure 10: Fact Account Segment MOB Summary
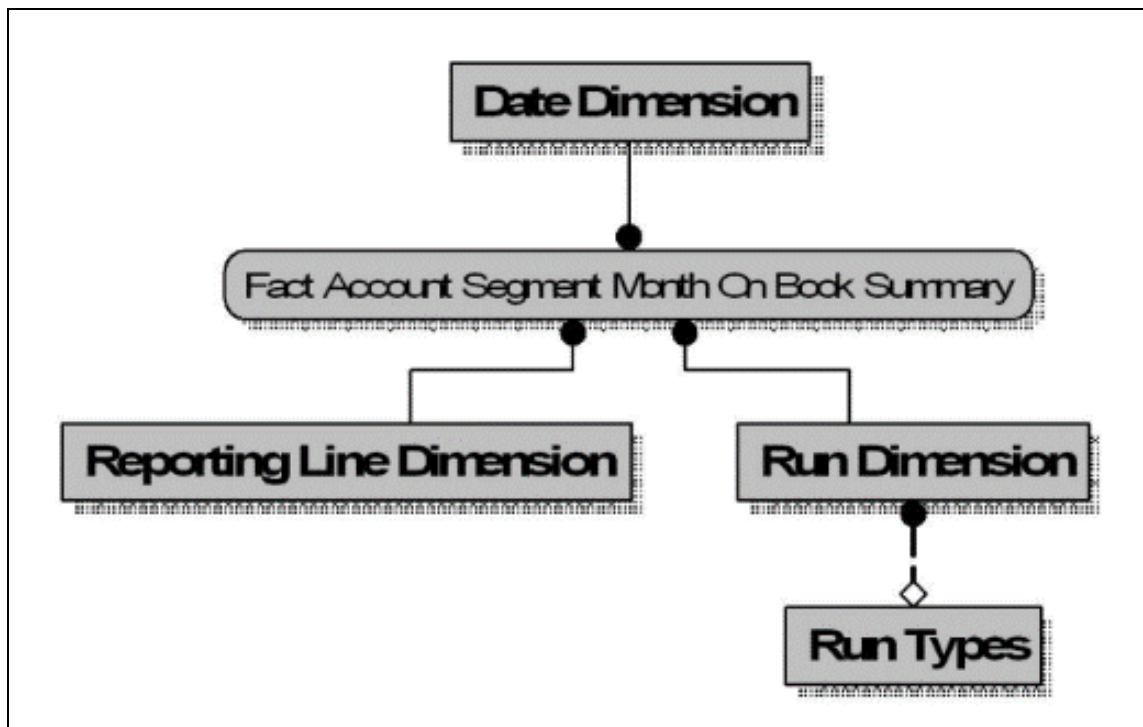
Figure 11: Fact Account Segment Score
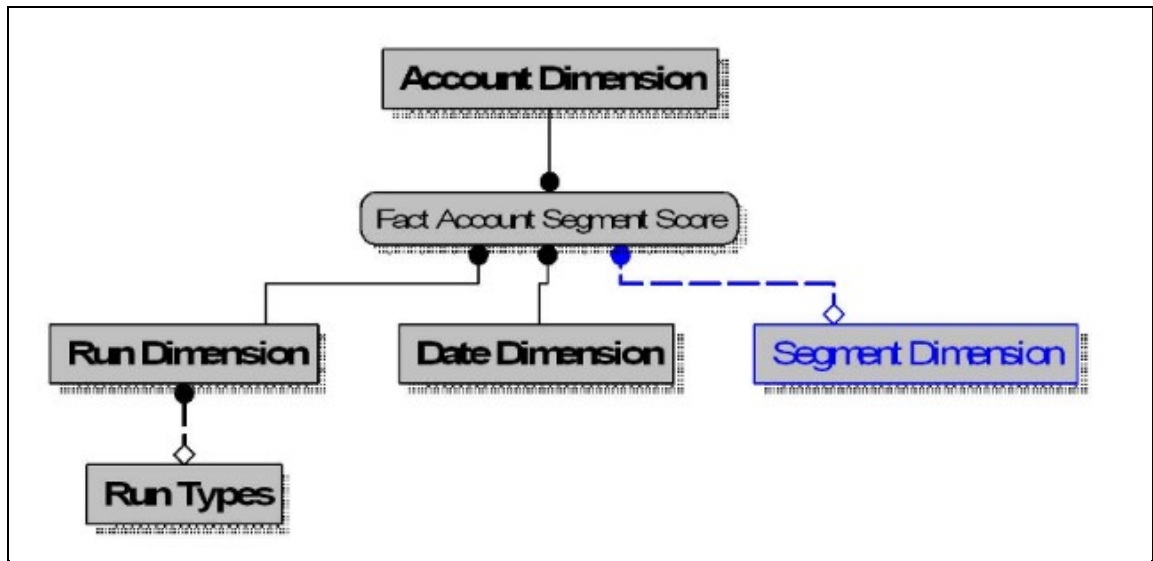
Figure 12: Fact Applications Summary
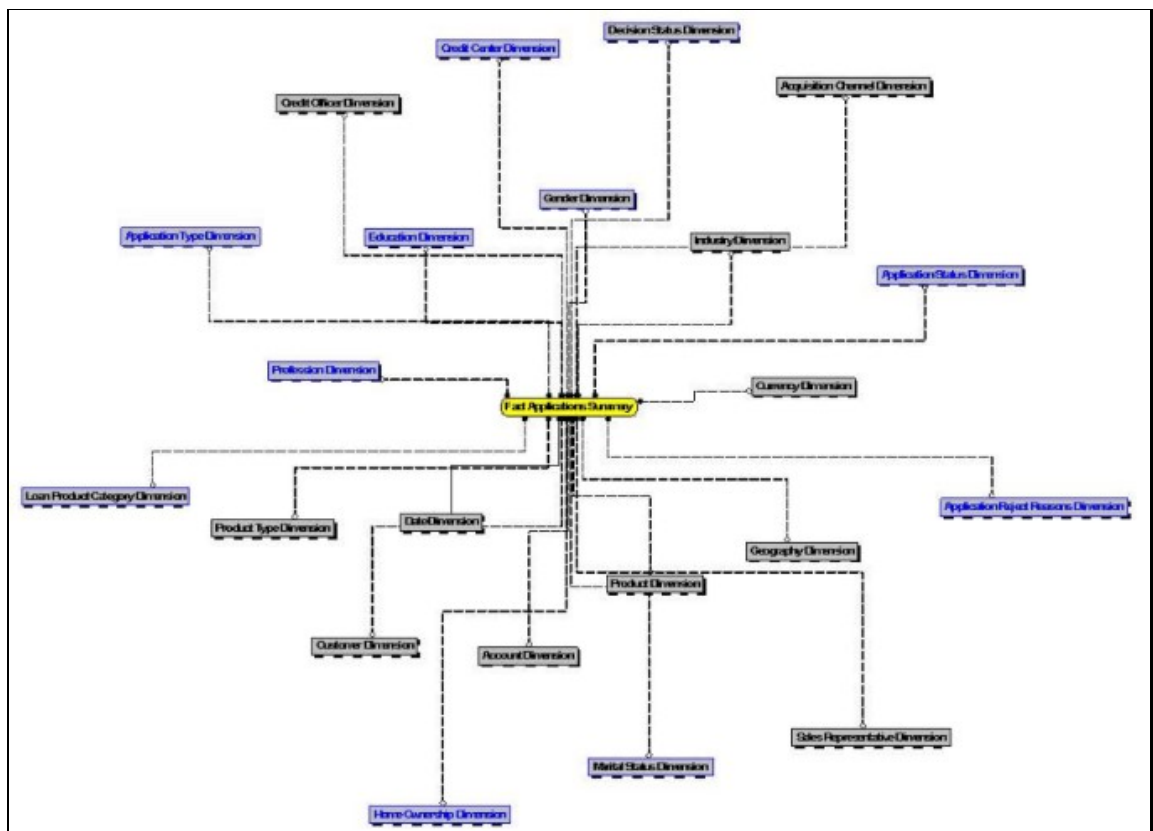
Figure 13: Fact Common Account Summary
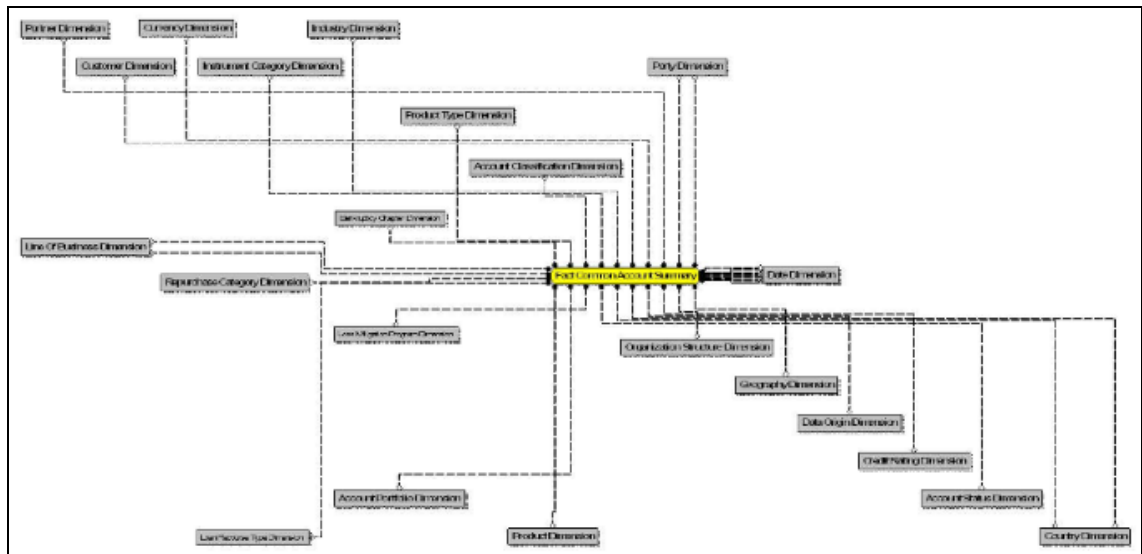
Figure 14: Fact Common Customer Summary
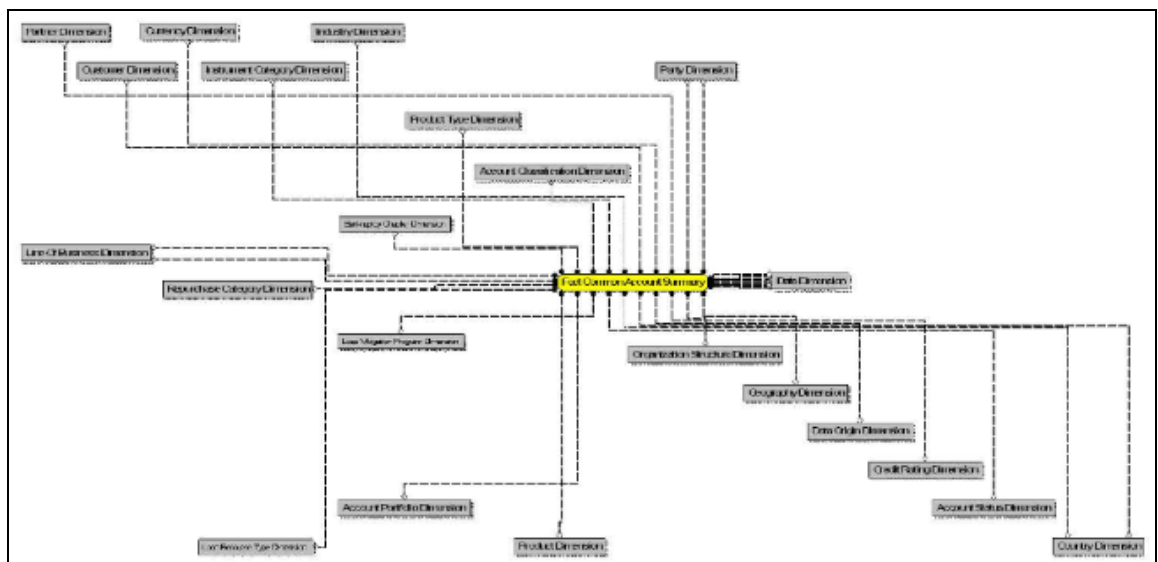
Figure 15: Fact CRM Account Summary
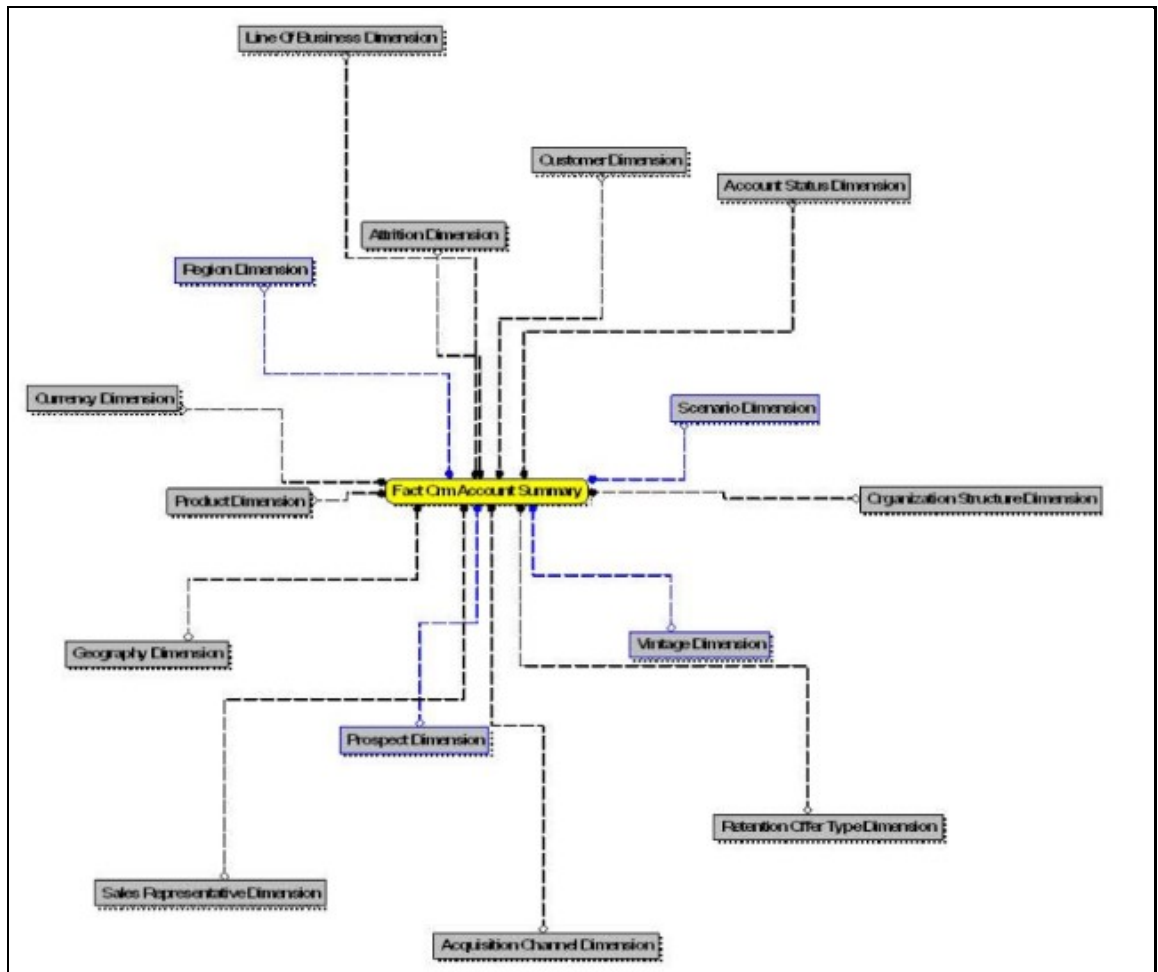
Figure 16: Fact Customer to Customer Relationship
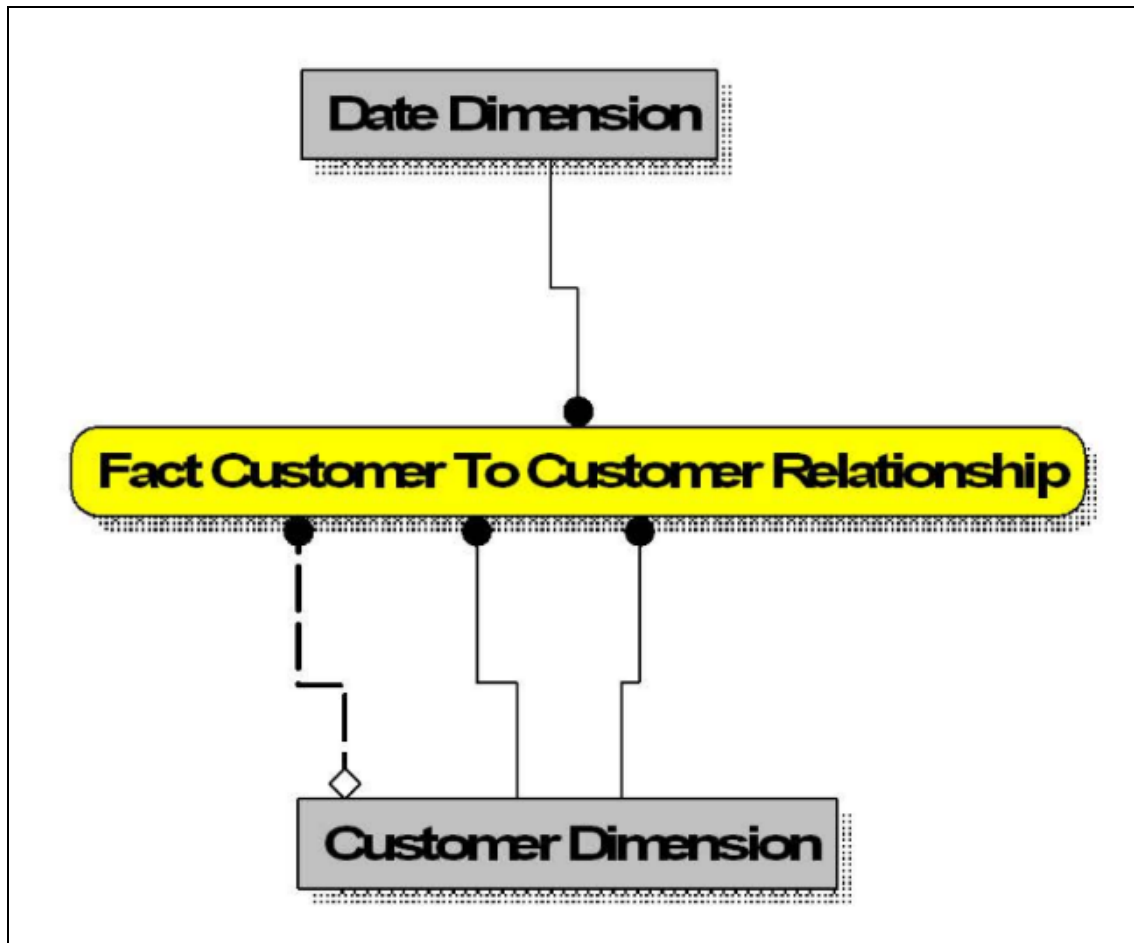
Figure 17: Fact Economic Capital Account Summary
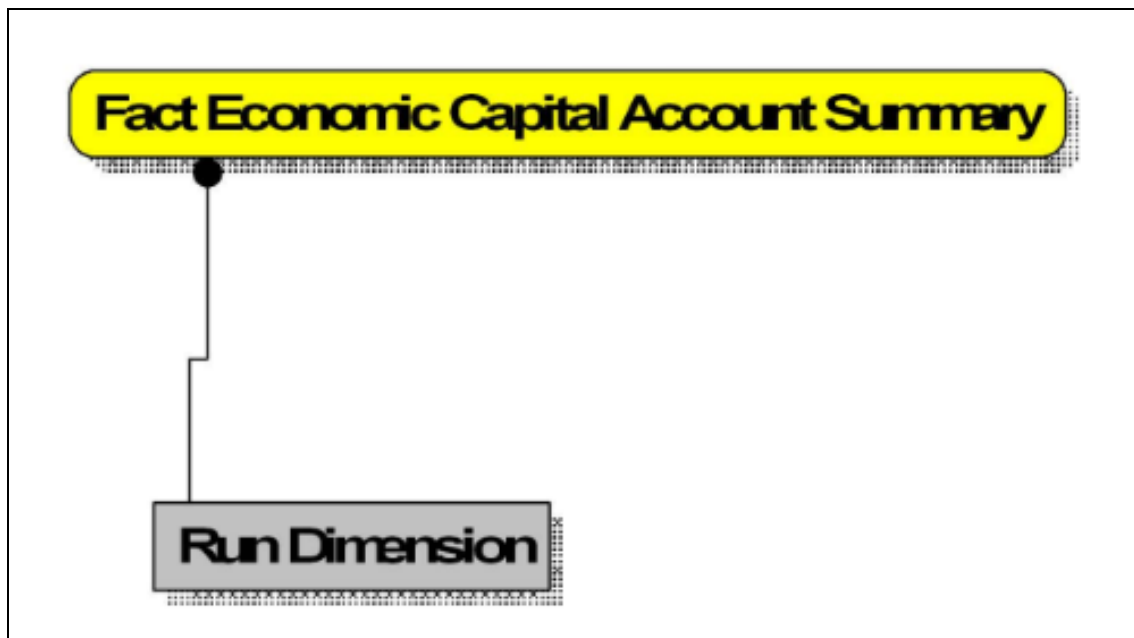
Figure 18: Fact Opportunity
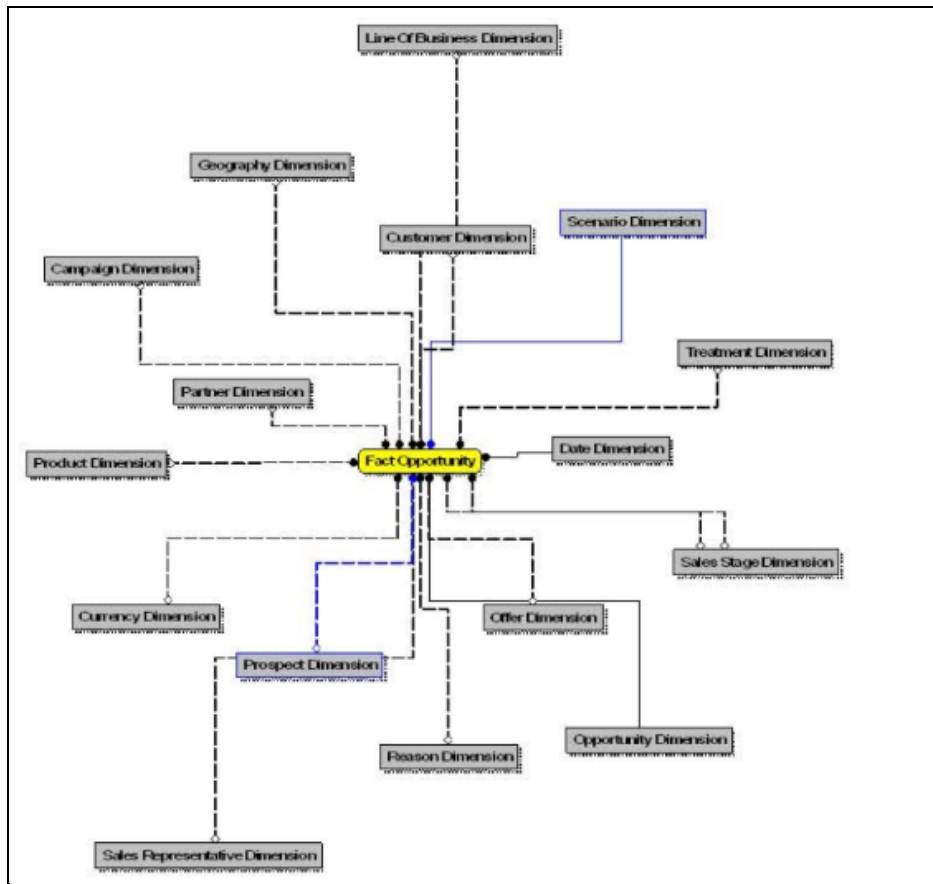
Figure 19: Fact Opportunity Activity
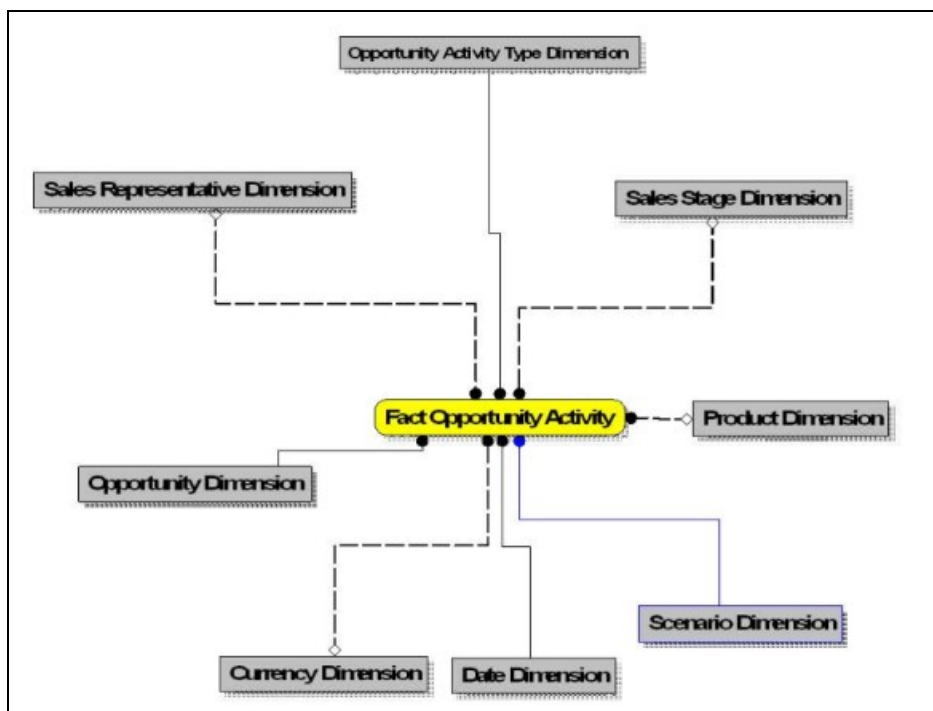
Figure 20: Fact Regulatory Capital Account Summary
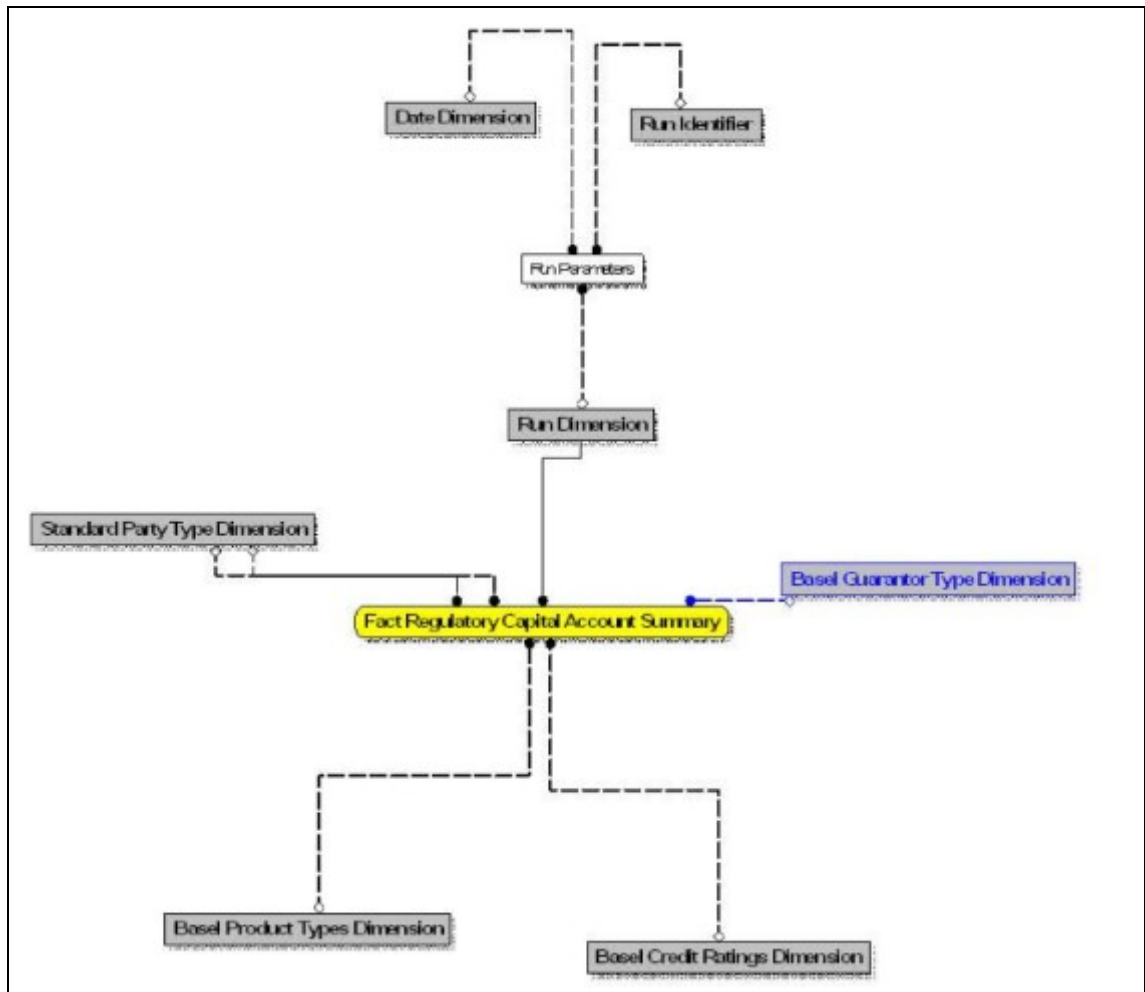
Figure 21: Fact Sales Representative Compensation
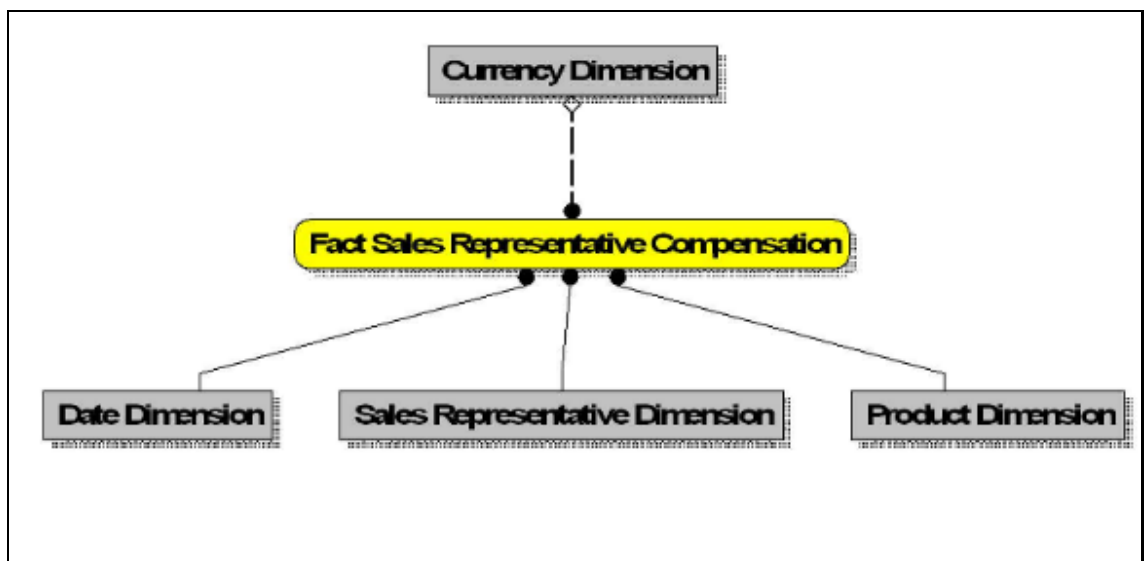
Figure 22: Fact Transaction Summary
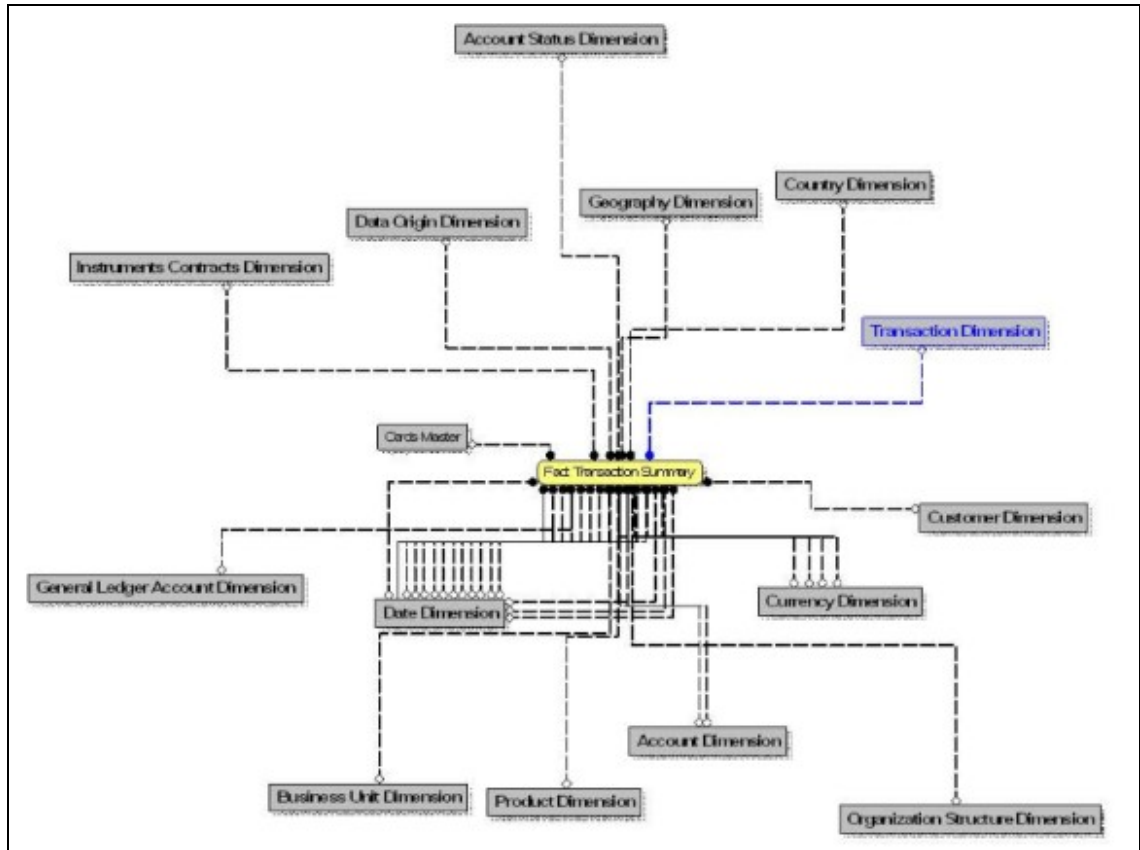
Figure 23: Funds Transfer Pricing (FTP) Account Summary
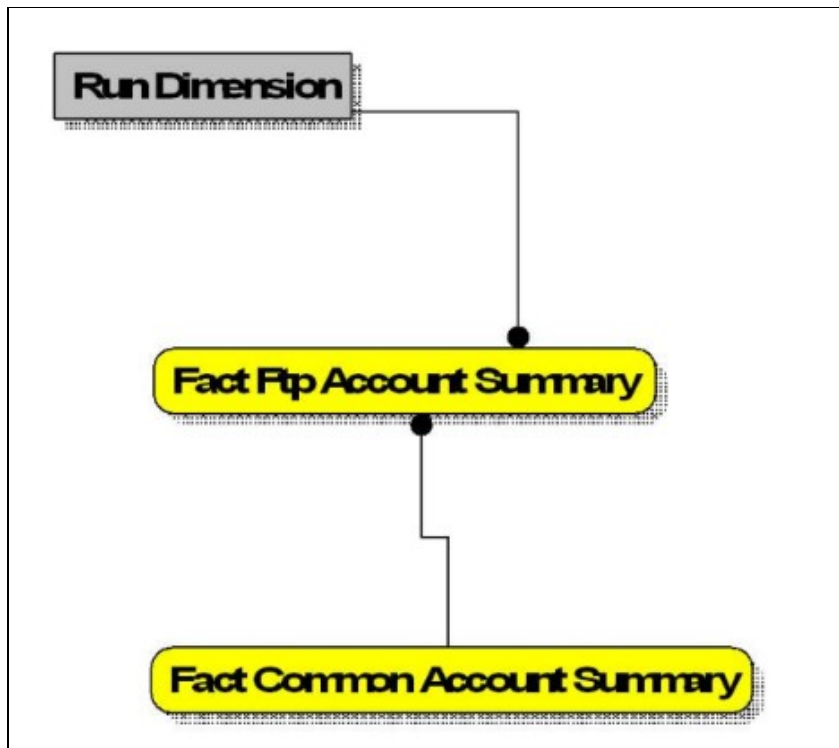
Figure 24: Profitability Management (PFT) Account Summary
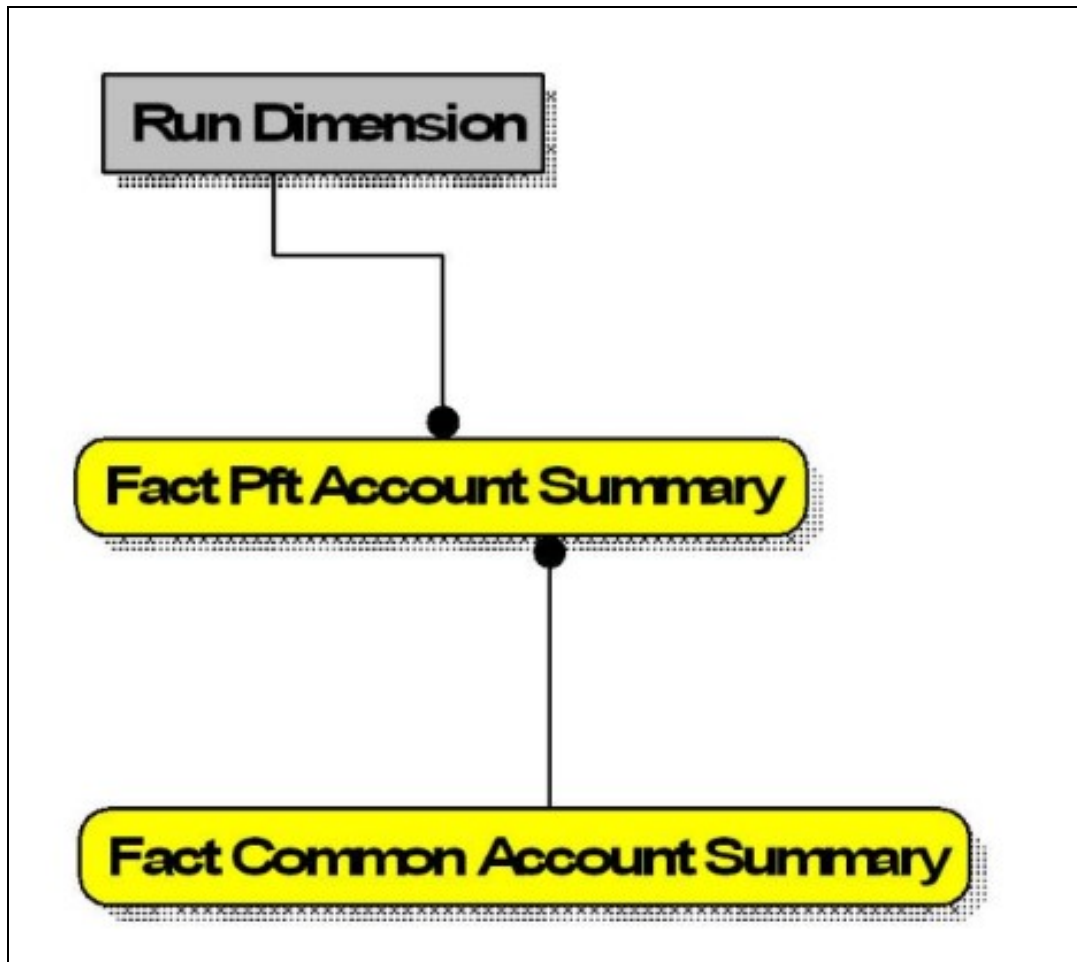
Figure 25: PFT Customer Summary
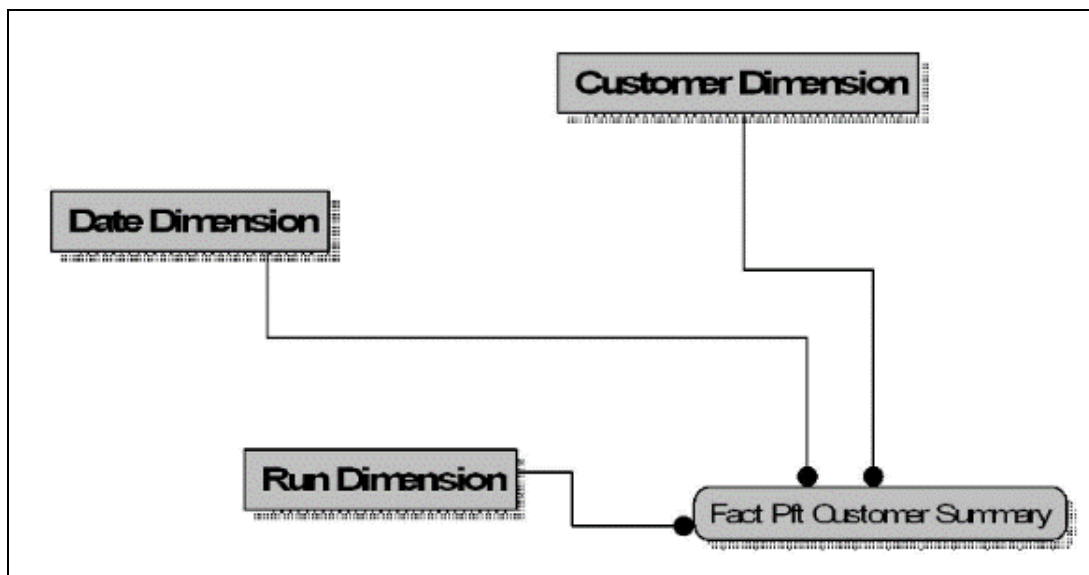
3.4 Data Flow: OFS PACS BI Data Model to Essbase Cubes

Reports of OFS PACS application can be configured to work on Relational database or Hyperion Essbase Multi-dimensional databases, which are 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.

OFS PACS application has the following seeded cube metadata:

Table 6:Seeded Cube Metadata

Cube Code	Cube Name	Fact Entities in the 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

4 Dimension Loading Process

This chapter covers the following sections:

- Dimension Tables Population
- Overview of SCD Process

4.1 Dimension Tables Population

OFS Performance Analytics Cloud Service uses the SCD component to handle dimensional data changes.

4.2 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

  https://www.oracle.com/technetwork/middleware/data-integrator/learnmore/odi-best-practice-data-warehouse-168255.pdf

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

  https://docs.oracle.com/cd/E11882_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/owb10 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?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 the OFSPA solution are Type 1 and Type 2.

4.2.1 Prerequisites

The following are the prerequisites:

- The SCD executable should be present under `<installation home>ficdb/bin` directory. The file name is scd.
- The user executing the SCD component should have executed rights on the file.

- The setup tables accessed by the SCD component are SYS_TBL_MASTER and SYS_STG_JOIN_MASTER. SYS_TBL_MASTER stores the information such as 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 such as 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.

4.2.2   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 7: 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	The 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 is 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.

Table 8: Sample Data seeded by the Installer for the LOB 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 be manually inserted into this table.

- SYS_STG_JOIN_MASTER

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

Table 9: 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 Values for COL_TYPE Column Name 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 is VARCHAR, DATE, NUMBER based on the underlying column data type.
COL_FORMAT	VARCHAR2(15) NULL	

4.2.2.1.1 Values for COL_TYPE Column Name

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

- PK – Primary Dimension Value (can be multiple for a given Mapping Reference Number)
- SK – Surrogate Key
- DA – Dimensional Attribute (can 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 and Time

- LUB – Last Updated By

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

Table 10: Sample Data seeded by the installer for the LOB 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 into 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 the installation 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

4.2.3   Executing the SCD Component

To execute the SCD component from the OFSAAI ICC framework, follow these steps to create a batch:

8. From the **Home** menu, select **Operations**, then select **Batch Maintenance**.

9. Click **New Batch** ('**+**' symbol in **Batch Name** pane) and enter the **Batch Name** and **Description**.

10. Click **Save**.

11. Select the Batch you created in the earlier step by clicking the check box in the **Batch Name** pane.

12. Click **New Task** ('**+**' symbol in **Task Details** pane).

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

14. Select **Run Executable**, from the **Component ID** drop-down list.

15. 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>.

  For example:

  — -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 come seeded with the installation.

- **Wait**: When the file is executed, you have the choice to either wait till the execution gets completed 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**. Click **Yes** to wait for the execution to complete. Click **No** to proceed.

- **Batch Parameter**: Click **Yes** to pass the batch parameters to the executable being started; else the batch parameters will not be passed to the executable.

> **ATTENTION**   Always select **Yes** in the Batch Parameter.

- 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 process PR_REFERENCE_DATA is provided which has all the required dimensions as different tasks that are part of SCD.

4.2.4   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**, and then select **Batch Monitor**.

For more comprehensive coverage, see the Oracle Financial Services Analytical Applications Infrastructure User Guide.

The status messages in the **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
```

You can access the detailed SCD component log on the application server in the `<FTPSHARE PATH>/logs/<EXECUTION DATE>/<INFODOM_NAME>` directory. 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 unable to find the paths mentioned earlier.

5   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 period (for example, income for March). Time dimension makes it possible to report the balances by Year, Quarter, or Month using the rollup functionality of cubes. Cubes make it possible to roll up 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 Quarter 1, 2, 3, and 4 data get rolled up to, say the Year 2011. The rollup of a particular balance depending on its 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 nonadditive rollups wherein a node formula is used to specify how to roll up the child member balances (for example, 3 months rolling average).

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

Topics:

- Overview of Time Dimension Population
- Tables used by the Time Dimension Population Transformation

5.1   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.

5.1.1   Prerequisites

- All the post-installation steps mentioned in the Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration Guide and the solution installation manual of Oracle Financial Services Institutional Performance Analytics have to be completed successfully.
- Application User must be mapped to a role that has the seeded batch execution function (BATPRO).
- 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide).
  - Iccserver
  - Router
  - AM Server
  - Message server

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

5.2 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*.

5.2.1 Executing the Time Dimension Population Transformation

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

For more comprehensive coverage of configuration and execution of a batch, see the 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 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 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 Dim_Dates_Population from the list of all available transformations. (This is a seeded Data Transformation that is installed as part of the OFS IPA solution installer. If you do not see this in the list, contact My Oracle Support).

   - **Parameter List**: Start Date, End Date (see the following for details on Parameter list).

     The explanation for the parameter list is:

     — **Start Date**: This is the date starting from which the Transformation will populate the DIM_DATES table. The date should be specified in the format **YYYYMMDD**.

     — **End Date**: This is the date up to which the Transformation will populate the Dim_Dates table. The date should be specified in the format **YYYYMMDD**. The 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.

    A seeded process PR_DIM_DATES 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.

5.2.2   Checking the Execution Status

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

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

The status messages in the batch monitor are:

- N: Not Started

- O: On Going

- F: Failure

- S: Success

The **Event Log** window in the **Batch Monitor** provides logs for execution with the most recent at the top of the list. If there is an error during execution, it will get listed in the log file. Even if you see **Successful** as the status in the **Batch Monitor** it is advisable to go through the log and check if there are any errors. The execution log can be accessed on the application server by navigating to the following `<FTPSHARE PATH>/logs/<EXECUTION DATE>/<INFODOM_NAME>` directory. 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.

6   Customer Dimension Population

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 the 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 table. This information is stored in the DIM_CUSTOMER table as N_CUST_SKEY.

FSI_MERGE_SETUP_DETAILS

The 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 11: FSI_MERGE_SETUP_DETAILS Table Columns

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 the Customer dimension population.
SOURCE_COLUMN	VARCHAR2 (30 CHAR)	This is the source column for the Customer dimension population.
TARGET_COLUMN	VARCHAR2 (30 CHAR)	This is the target column for the Customer dimension population.
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 to the source column for the Customer dimension population.
AGGREGATE_FUNCTION	VARCHAR2 (30 CHAR)	This is used for aggregating data for some source columns.

Here is sample data:

Table 12: 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

The 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 13: FSI_MERGE_SETUP_MASTER Table

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 the Customer dimension population.
TARGET_TABLE	VARCHAR2 (30 CHAR)	This is the target column for the Customer dimension population.
ANSI_JOIN	VARCHAR2 (4000 CHAR)	This is the join condition that results in the dataset.
FILTER_CONDITION	VARCHAR2 (4000 CHAR)	This is used for filtering the values in where clause.

Here is a sample data:

Table 2: Sample Data

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

6.1 Executing the Customer Dimension Population

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

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 the appropriate datastore from the list.

   ▪ **Datastore Name**: Select the 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.

   A seeded process PR_PARTY is provided so that the user can just modify the parameters and execute the batch.

6.1.1   Checking the Execution Status

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

For more comprehensive coverage of configuration and execution of a batch, see the 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 an 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.

7   Account Dimension Population

Topics:

- [Dimension Tables Population](#)
- [Overview of SCD process](#)
- [Tables Used by the SCD Component](#)

7.1   Dimension Tables Population

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

7.2   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 14: Type 1 SCDs Overwriting Example

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

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 to 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 15: Type 1 SCDs - Overwriting1

N_PRODUCT_ SKEY	V_PRODUCT_ NAME	D_START_DATE	D_END_DATE	F_LATEST_REC ORD_INDICATO R
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 16: Type 2 SCDs – Create another Dimension Record

N_PRODUCT_ SKEY	V_PRODUCT_ NAME	D_START_DATE	D_END_DATE	F_LATEST_REC ORD_INDICATO R
1	PL	6/30/2010	12/31/9999	N
2	Personal Loan	6/30/2010	12/31/9999	Y

A new record is inserted into 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.

7.2.1   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 executed rights on the file mentioned as a prerequisite in point 2.

3. The setup tables accessed by the 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.

7.3   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 17: 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
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 into this table manually.

- SYS_STG_JOIN_MASTER

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

Table 18: 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, BUDGE
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 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
- 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 19: Sample Data seeded by the installer for SYS_STG_JOIN_MASTER Table

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

Table Name	Column Name	Expected Values
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 into 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 the installation 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
    ```

7.3.1   Executing the SCD Component

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

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>

   For example:

   — -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 that come seeded with the install.

   ▪ **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.

ATTENTION	Always select **Y** in Batch Parameter.

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

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

   A seeded process PR_REFERENCE_DATA is provided which has all the required dimensions as different tasks that are part of SCD.

7.3.2 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**.

For 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`

You can access the detailed SCD component log on the application server in the `<FTPSHARE PATH>/logs/<EXECUTION DATE>/<INFODOM_NAME>` directory. The file name will have the batch execution ID.

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.

7.3.3   Load DIM_ACCOUNT through SCD

The SCD population in the 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 following columns into the 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 the FSI_DIM_ACCOUNT_SETUP_DETAILS table.

7.3.4   DIM_ACCOUNT SCD

The process PR_ACCOUNT is introduced where different product processors would be the source and DIM_ACCOUNT would be the target. MAP_REF_NUMs have been introduced into the 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.

7.3.5   Load DIM Tables through SCD

The process PR_REFERENCE_DATA is introduced with tasks under it. These tasks represent the SCD processes where different staging tables would be the source and Dimension Tables would be the targets. The required SCDs have been introduced into the 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.

7.3.6   Improve SCD Performance

The 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

Table 20: 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_CONT R ACTS_V	/*+ parallel (DIM_ACCO UNT,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 the old to the 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 the 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 21: SETUP_MASTER configuration

V_COMPONENT_CODE	V_COMPONENT_DESC	V_COMPONENT_VALUE
DEFAULT_GAAP	DEFAULT_GAAP	USGAAP

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

7.3.7 Handling Multiple GAAP Codes

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 the SETUP_MASTER table. The SETUP_MASTER table should have only one record WHERE V_COMPONENT_DESC = 'DEFAULT_GAAP'.

8 Exchange Rate History Population

This chapter discusses the following topics:

- Introduction
- Execution of Currency Exchange Rates Population T2T
- Currency Execution Rates - Batch Execution
- Exchange Rate History Population

8.1 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. The Exchange Rate History entity is loaded using the Table to Table Transformation process.

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

Table 22: 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

8.2 Execution of Currency Exchange Rates Population T2T

The process PR_CURRENCY_CONVERSION needs to be executed to populate fsi_exchange_rates as the entries in setup_master are seeded during installation.

FSI_EXCHANGE_RATES table has to be loaded before loading any of the other Account Summary tables.

- Metadata Browser
- Common Account Summary

8.3 Currency Execution Rates - Batch Execution

A seeded process, PR_CURRENCY_CONVERSION has to be executed for the required MIS Date.

Alternatively, the 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**.

   The data file name will be blank for any **Table to Table Load** mode. Default value refers to currency calculation. If there is a requirement for currency conversion in T2T transactions, a Default value must be provided.

9. Execute the batch created in the preceding steps.

8.4   Exchange Rate History Population

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

A seeded process, **PR_CURRENCY_CONVERSION** has to be executed for the required date.

Figure 26: PR_CURRENCY_CONVERSION

Alternatively, the 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 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 table to table transformation T2T_EXCHANGE_RATE_HIST.

   The 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 the load.

    T2T component can fail because of the 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.

8.4.1 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.

8.4.2 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 the FSI_EXCHANGE_RATE_HIST table whose 'effective date' lies in the range of these values will be validated on the 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.

9   Account Summary Population

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

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 that allow the 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 the 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, few other Account Summary tables 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 27: Account Summary Tables
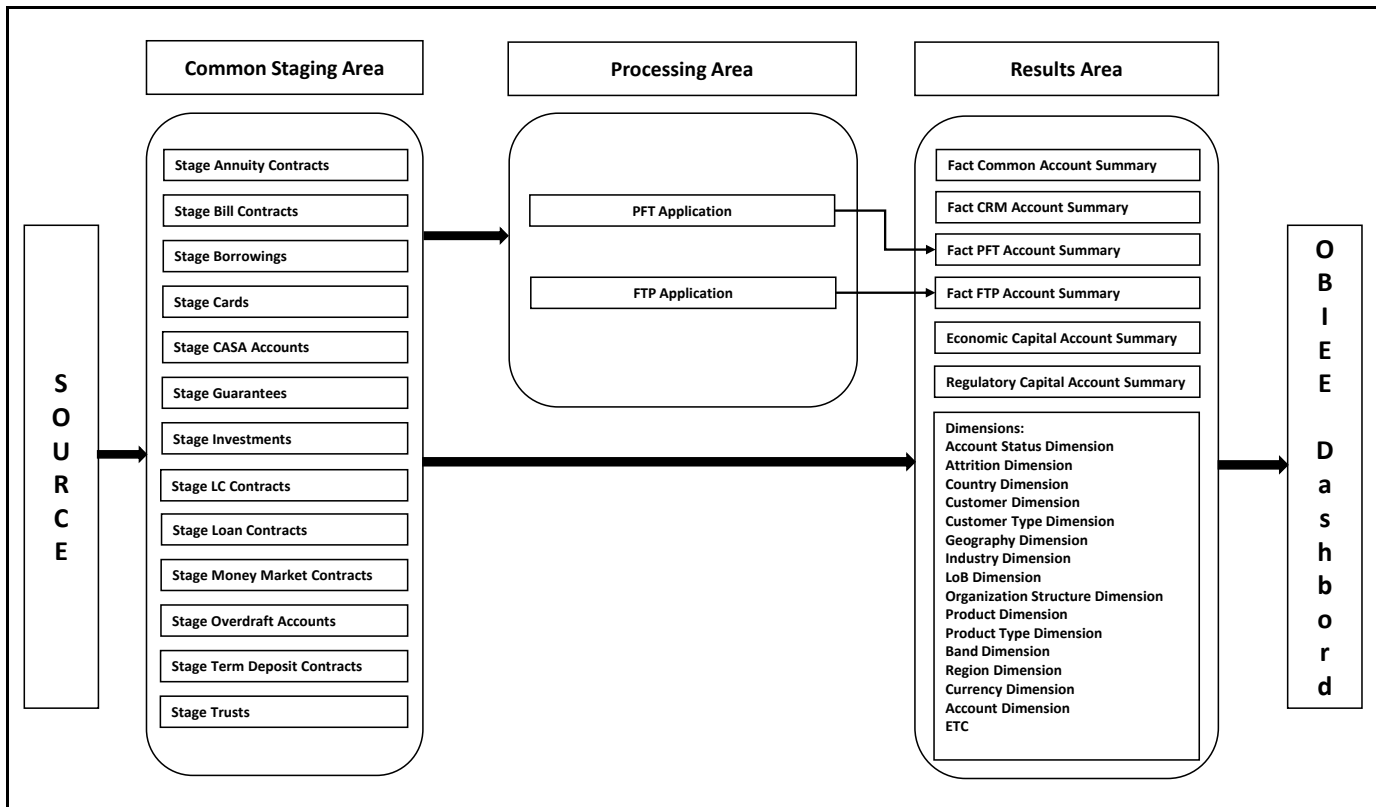
9.1 Overview of Account Summary Population

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 Definitions

9.1.1 Common Account Summary Definitions

Table 23: 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

SLNo	Source Table	T2T Definition Name	Destination Table
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_CONTRACTS	T2T_STG_COMMITMENT_CONTRACTS_CAS	FCT_COMMON_ACCOUNT_SUMMARY
16	STG_MUTUAL_FUNDS	T2T_STG_MUTUAL_FUNDS_CAS	FCT_COMMON_ACCOUNT_SUMMARY

9.1.2   CRM Account Summary

Table 24: CRM Account Summary

SLNo	Source Table	T2T Definition Name	Destination Table
1	STG_ANNUITY_CONTRACTS	T2T_STG_CRMAS_ANNUITY_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
2	STG_BILLS_CONTRACTS	T2T_STG_CRMAS_BILLS_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
3	STG_BORROWINGS	T2T_STG_CRMAS_BORROWINGS	FCT_CRM_ACCOUNT_SUMMARY
4	STG_CARDS	T2T_STG_CRMAS_CARDS	FCT_CRM_ACCOUNT_SUMMARY

SLNo	Source Table	T2T Definition Name	Destination Table
5	STG_CASA	T2T_STG_CRMAS_CASA	FCT_CRM_ACCOUNT_SUMMARY
6	STG_GUARANTEES	T2T_STG_CRMAS_GUARANTEES	FCT_CRM_ACCOUNT_SUMMARY
7	STG_INVESTMENTS	T2T_STG_CRMAS_INVESTMENTS	FCT_CRM_ACCOUNT_SUMMARY
8	STG_LC_CONTRACTS	T2T_STG_CRMAS_LC_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
9	STG_LEASES_CONTRACTS	T2T_STG_CRMAS_LEASES_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
10	STG_LOAN_CONTRACTS	T2T_STG_CRMAS_LOAN_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
11	STG_MM_CONTRACTS	T2T_STG_CRMAS_MM_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
12	STG_OD_ACCOUNTS	T2T_STG_CRMAS_OD_ACCOUNTS	FCT_CRM_ACCOUNT_SUMMARY
13	STG_TD_CONTRACTS	T2T_STG_CRMAS_TD_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
14	STG_TRUSTS	T2T_STG_CRMAS_TRUSTS	FCT_CRM_ACCOUNT_SUMMARY
15	STG_COMMITMENT_CONTRACTS	T2T_STG_CRMAS_COMMITMENTS	FCT_CRM_ACCOUNT_SUMMARY
16	STG_MUTUAL_FUNDS	T2T_STG_CRMAS_MUTUAL_FUNDS	FCT_COMMON_ACCOUNT_SUMMARY

9.1.3   FTP Account Summary

Table 25: FTP Account Summary Definitions

SLNo	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

SLNo	Source Table	T2T Definition Name	Destination Table
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_TERM_DEPOSITS	T2T_FCT_FTP_ACCOUNT_TRUSTS	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

9.1.4 PFT Account Summary

Table 26: PFT Account Summary Definitions

SLNo	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
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

SLN o	Source Table	T2T Definition Name	Destination Table
10	FSI_D_MORTGAGES	T2T_FCT_PFT_ACCOUNT_MORTGAGES	FCT_PFT_ACCOUNT_SUMMAR Y
11	FSI_D_TERM_DEPOSITS	T2T_FCT_PFT_ACCOUNT_DEPOSITS	FCT_PFT_ACCOUNT_SUMMAR Y
12	FSI_D_TRUSTS	T2T_FCT_PFT_ACCOUNT_TRUSTS	FCT_PFT_ACCOUNT_SUMMAR Y
13	FSI_D_MUTUAL_FUNDS	T2T_FCT_PFT_ACCOUNT_MUTUAL_FUN D	FCT_PFT_ACCOUNT_SUMMAR Y

9.2   Prerequisites

The following are the prerequisites to execute the Account Summary Populations T2Ts:

1. All the post-installation steps mentioned in the Oracle Financial Services Analytical Applications Infrastructure 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 a 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.)

    - Iccserver
    - Router
    - AM Server
    - Messageserver

4. Batches will have to be created for execution. 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 Dimension Loading Process and Time Dimension Population)

9.2.1   Fact Common Account Summary

The 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 before executing the T2T:

- DIM_DATES
- DIM_ACCOUNT
- DIM_CUSTOMER

- DIM_PRODUCT
- DIM_CHANNEL
- DIM_BANDS
- DIM_ORG_STRUCTURE and so on.

9.2.2 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 before 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, see the Dimension Tables Population section.

For details on populating the DIM_DATES dimension table, see the Overview of Time Dimension Population section.

For identifying fields required in Channel Transaction tables in staging for Customer Insight Application(s), see the Download Specification.

For more information on the dimensions, see the ERwin Data model.

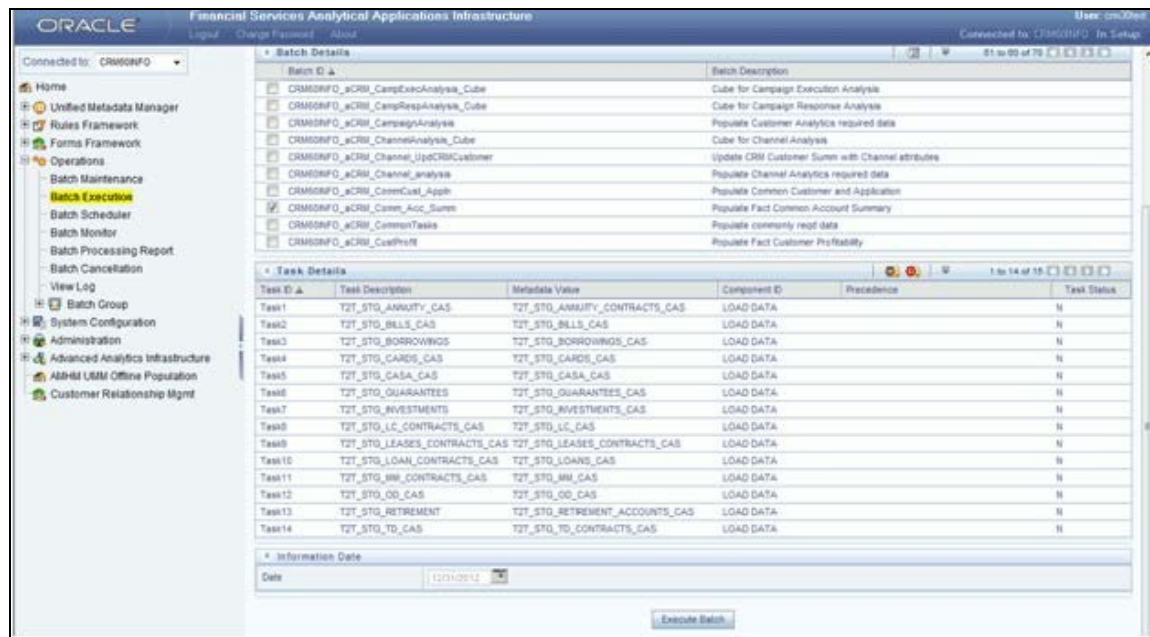## 9.2.3 Executing the Account Summary Population T2T

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

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

9.2.4 Fact Common Account Summary

A seeded process, PR_ACCOUNT must be executed for the required MIS Date.

Figure 28: PR_ACCOUNT

Alternatively, to create a new batch, follow these steps:

1. From the **Home** menu, select **Operations**, and 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 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.

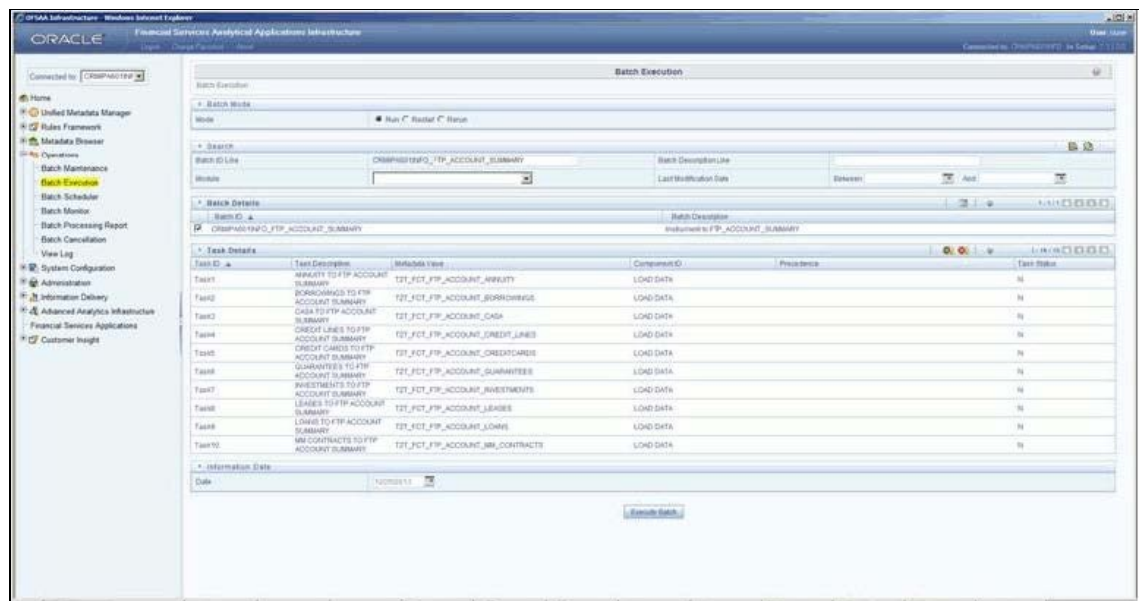   ▪ **Load Mode**: Select Table to Table from the list.

- **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. The data file name will be blank for any Table to Table Load mode. The default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, the Default value has to be provided. For example, the default value is [DRCY]='USD' Here 'USD' acts as the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

9.2.5 Fact FTP Account Summary

A seeded process, PR_FTP_ACCOUNT must be executed for the required MIS Date.

Figure 29: PR_FTP_ACCOUNT

Alternatively, to create a new batch, follow these steps:

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.

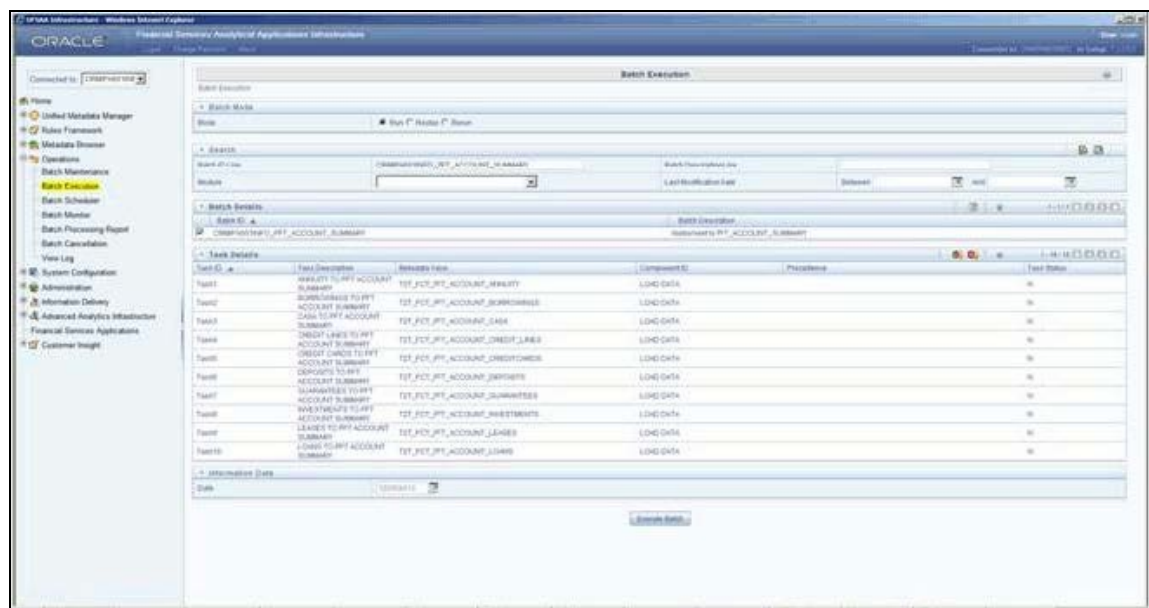7. Select the following from the Dynamic Parameters List and click Save.

- 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. The data file name will be blank for any Table to Table Load mode. The default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

9.2.6 Fact PFT Account Summary

A seeded process, PR_PFT_ACCOUNT must be executed for the required MIS Date.

Figure 30: <INFODOM>_PFT_ACCOUNT_SUMMARY

Alternatively, to create a new batch, follow these steps:

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. The data file name will be blank for any Table to Table Load mode. The 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 the 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.

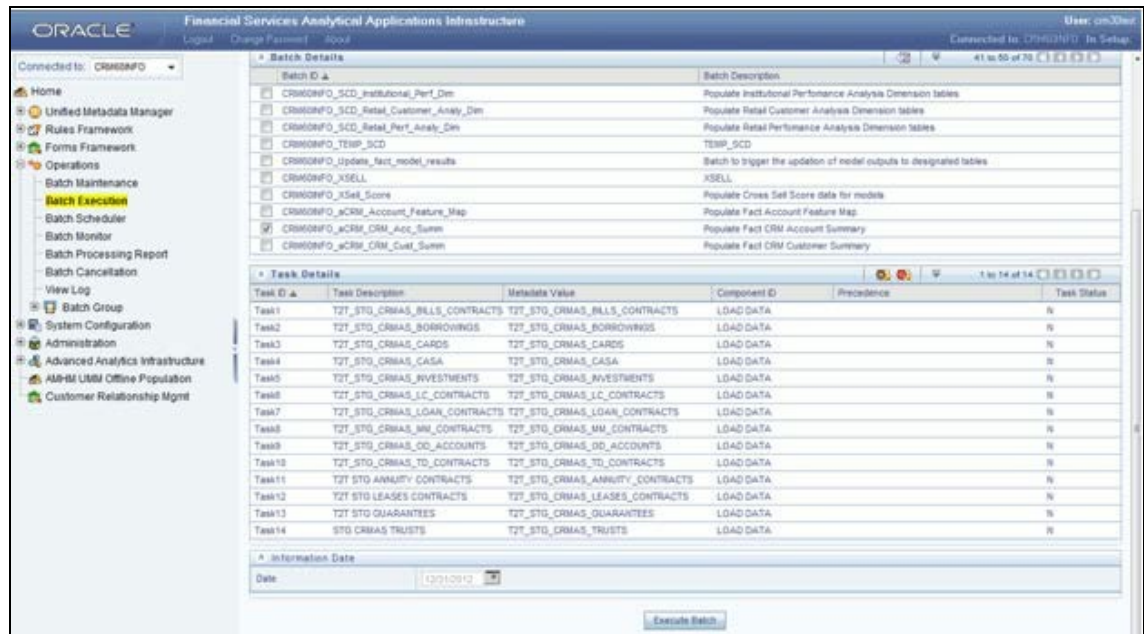   ▪ Parameter List - Pass the values 1, 180, $RUNSK= -1, USD.

   It is mandatory to pass all five parameters. Currently, the first three do not have functional significance. The last two parameters are **Run Skey** and **Reporting Currency** values, which need 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 the FCT_CRM_ACCOUNT_SUMMARY table. If the RUN_EXE_PARAMETERS table already has 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 **Precedence** in the **Task Details** pane.

14. Execute the batch created in the preceding steps.

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

9.2.7   Fact CRM Account Summary

A seeded process, PR_ACCOUNT has to be executed for the required MIS Date.

Figure 31: PR_ACCOUNT

Alternatively, to create a new batch, follow these steps:

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. The data file name will be blank for any Table to Table Load mode. The 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 the 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 five parameters. Currently, the first three do not have functional significance. The last two parameters are "Run Skey" and "Reporting Currency" values, which need 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 the [Oracle Financial Services Analytical Applications Infrastructure User Guide](#).

9.2.8 Checking the Execution Status

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

For more comprehensive coverage of configuration and execution of a batch, see the [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$

9.2.9 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.

10 Fact Transaction Summary

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

Table 27: Common Account Summary T2T Definitions

SLNo	Source Table Name	T2T Definition Name	Destination Table
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_F TS	FCT_TRANSACTION_SUMMARY
4	STG_BORROWING_COM MITMENT_TXNS	STG_BORROWING_COMMITME NT_TXNS_FTS	FCT_TRANSACTION_SUMMARY
5	STG_BORROWINGS_TX NS	STG_BORROWINGS_TXNS_FTS	FCT_TRANSACTION_SUMMARY
6	STG_CARDS_PAYMENT _TXNS	STG_CARDS_PAYMENT_TXNS_F TS	FCT_TRANSACTION_SUMMARY
7	STG_CARDS_SETTLEME NT_TXNS	STG_CARDS_SETTLEMENT_TXN S_FTS	FCT_TRANSACTION_SUMMARY
8	STG_CASA_TXNS	STG_CASA_TXNS_FTS	FCT_TRANSACTION_SUMMARY
9	STG_COMMITMENT_CO NTRACT_TXNS	STG_COMMITMENT_CONTRAC T_TXNS_FTS	FCT_TRANSACTION_SUMMARY
10	STG_COMMODITIES_TX NS	STG_COMMODITIES_TXNS_FTS	FCT_TRANSACTION_SUMMARY
11	STG_CORRESPONDENT _ACCT_TXNS	STG_CORRESPONDENT_ACCT_ TXNS_FTS	FCT_TRANSACTION_SUMMARY
12	STG_CREDIT_DERIVATI VES_TXNS	STG_CREDIT_DERIVATIVES_TXN S_FTS	FCT_TRANSACTION_SUMMARY
13	STG_FOREX_TXNS_FTS	STG_FOREX_TXNS_FTS	FCT_TRANSACTION_SUMMARY
14	STG_GUARANTEES_TX NS	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_TXN S	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

SLNo	Source Table Name	T2T Definition Name	Destination Table
20	STG_LEASES_TXNS	STG_LEASES_TXNS_FTS	FCT_TRANSACTION_SUMMARY
21	STG_LOAN_CONTRACT_TXNS	STG_LOAN_CONTRACT_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_CONTRACTS_TXNS_FTS	FCT_TRANSACTION_SUMMARY
29	STG_RETIREMENT_ACCOUNTS_TXNS	STG_RETIREMENT_ACCOUNTS_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_ACCOUNT_TXNS_FTS	FCT_TRANSACTION_SUMMARY
35	STG_FUTURES_TXNS	STG_FUTURES_TXNS_FTS	FCT_TRANSACTION_SUMMARY
36	STG_MUDARABAH_TXNS	STG_MUDARABAH_TXNS_FTS	FCT_TRANSACTION_SUMMARY

10.1 Executing the Fact Transaction Summary

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

Fact Common Account Summary - Batch Execution

A seeded process, PR_ACCOUNT has to be executed for the required MIS Date.

Alternatively, the 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:

   ▪ **Data Store 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. Click **Save**.

   The 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, a Default value has to be provided.

   For example, the 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.

11 Customer Summary Population

Fact Common Customer Summary table stores the attributes of 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 the account summary. The customer relationship table drives the relationship between accounts and customers. Common customer summary data is populated for all the active customers in the customer dimension.

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

Figure 32: Fact Common Customer Summary Dataflow
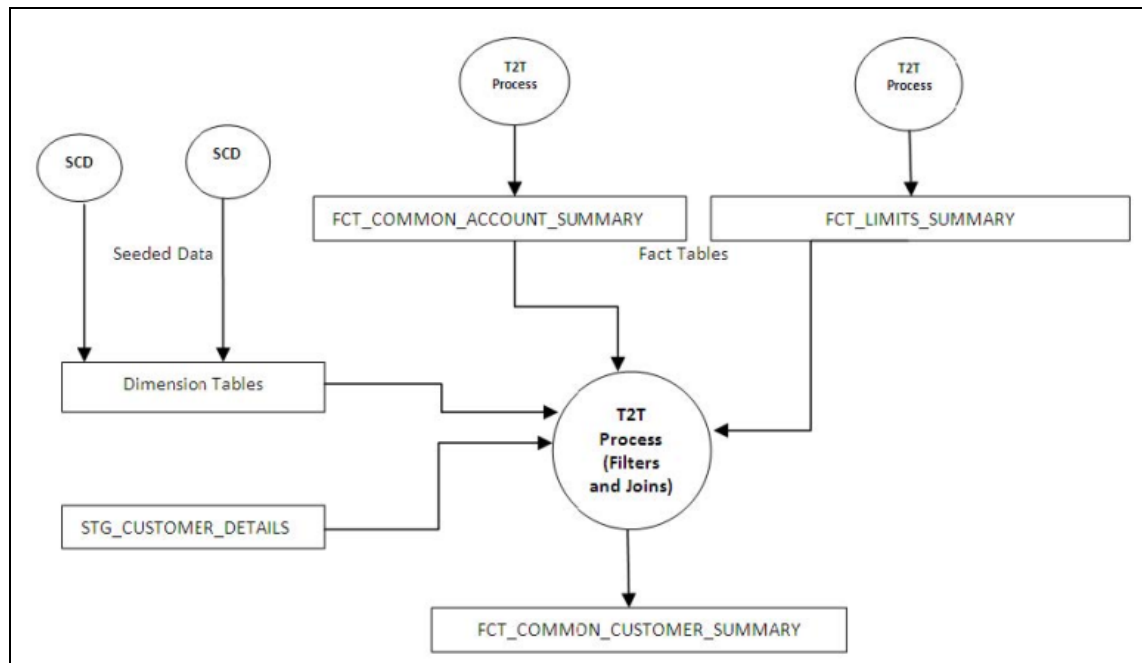
11.1 Prerequisites

Following are the lists of tables used in the population of Fact Common Customer Summary and these tables are required to be loaded before 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.

11.2 Executing the Customer Summary Population T2T

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

PR_PARTY_FACTS.

The following steps will help you to execute the batch:

1. Navigate to the Batch Execution screen.
2. Select the seeded batch PR_PARTY_FACTS where INFODOM is the information domain where the 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 33: Batch Monitor
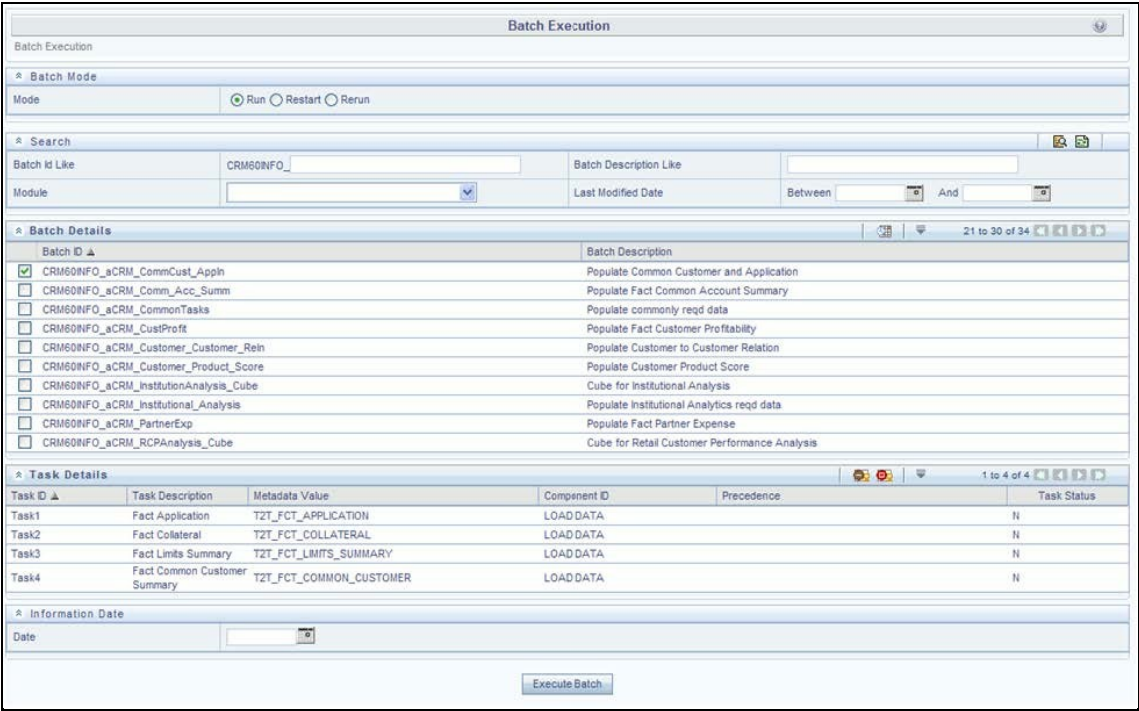
11.2.1.1   Error Messages

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 to re-load or loading existing records for the already executed AS_OF_DATE.

12   Fact Data Population

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

- Fact CRM Customer Summary

- Fact Partner Expense

- Fact Account Feature Map

- Fact Customer to Customer Relationship

- Fact Opportunity

- Fact Opportunity Activity

- Fact Sales Representative Compensation

- Fact Application

- Account Manager Relation

- Management Forecast

- Fact Account Customer Relation

- Fact Account Profitability

12.1   Fact CRM Customer Summary

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

12.1.1.1   Load Data into Fact CRM Customer Summary

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

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

Table 28: 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_	

T2T Definition Name	Source Table(s)	Destination Table
	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.

12.1.2   Prerequisites

Fact Common Customer Summary entity needs to be populated before executing the Fact CRM Customer Summary T2T. See 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 needs to have relevant data before 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 identifying fields required in Channel Transaction tables in staging for 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.

12.1.2.1   Executing the Fact CRM Customer Summary Population T2Ts

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

PR_PARTY_FACTS has to be executed for the required MIS Date.

Figure 34: Fact CRM Customer Summary Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

9. The 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.1.2.2 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

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

12.2   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 period.

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

Table 29: Seeded T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_PARTNER_EXPENSE	STG_PARTNER_EXPENSE	FCT_PARTNER_EXPENSE

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

12.2.1.1   Prerequisites

Following are the lists of tables used in the population of Fact Partner Expense and these are required to be loaded before 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 Dimension Loading Process.

For details on populating the DIM_DATES dimension table, refer to Time Dimension Population.
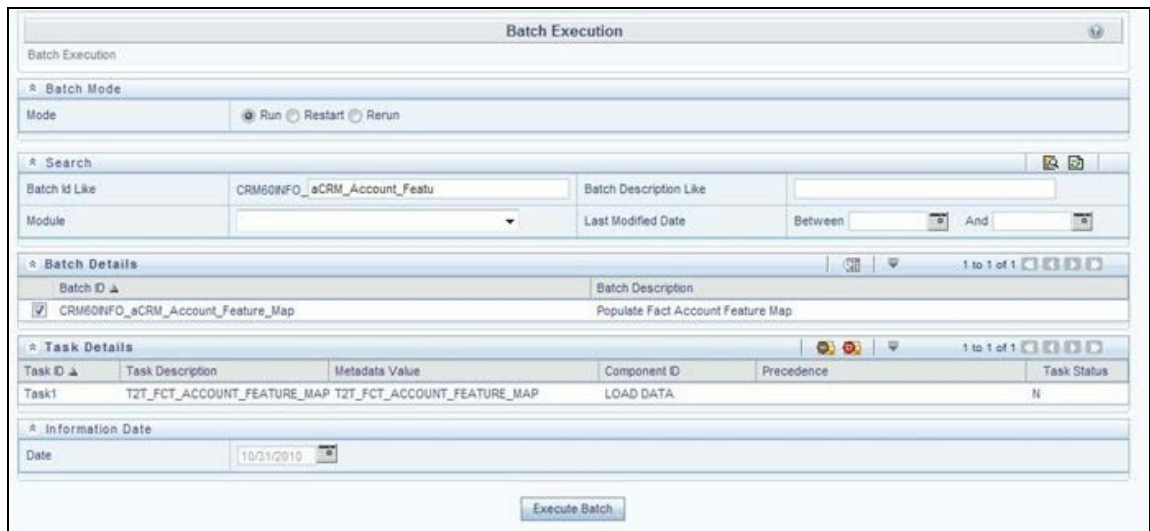
For identifying fields required in Channel Transaction tables in staging for Customer Insight Application(s), refer to Download Specification.

12.2.1.2   Executing the Fact Partner Expense Population T2T

To execute the T2T component from the 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 35: Execute Fact Partner Expense Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

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

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

9.   Execute the batch created in the preceding steps.

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

12.2.1.3   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 table can be queried for errors: FCT_PARTNER_EXPENSE$

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

12.3   Fact Account Feature Map

A product might be facilitated with its 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.

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

Table 30: Seeded T2T 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 the Customer Insight Erwin Data Model to view the detailed structure of the tables.

12.3.1.1   Prerequisites

Following are the lists of tables used in the population of Fact Account Feature Map and these tables are required to be loaded before 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 Dimension Loading Process.

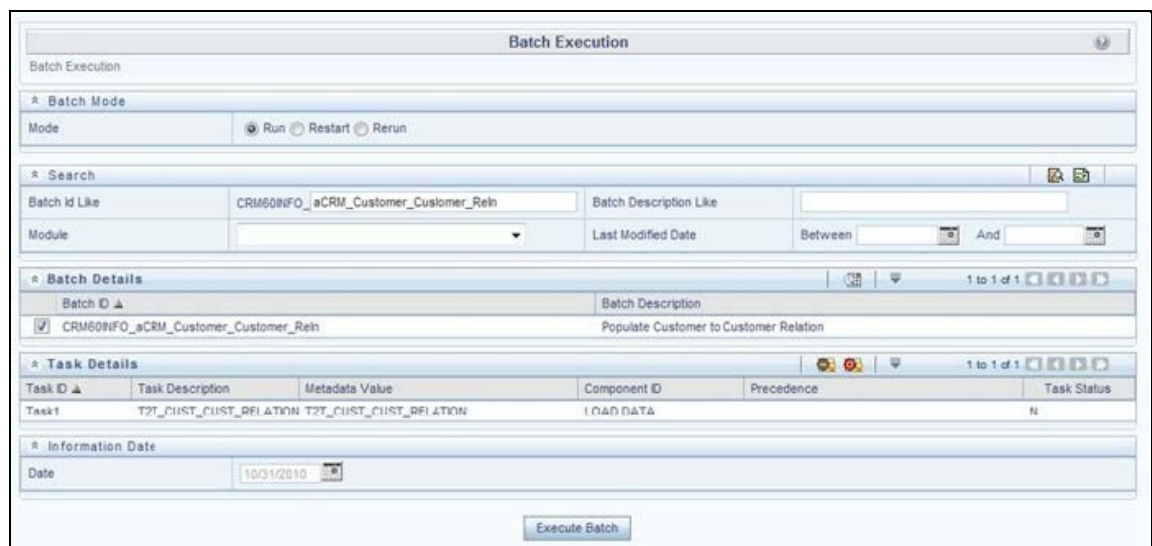For details on populating the DIM_DATES dimension table, refer to Time Dimension Population.

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

12.3.1.2 Executing the Fact Account Feature Map Population T2T

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

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

Figure 36: Execute Fact Account Feature Map Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

   The 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.3.1.3 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$

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

12.4 Fact Customer to Customer Relationship

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

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

   Table 31: Seeded T2T 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 the Customer Insight Erwin Data Model to view the detailed structure of the tables.

12.4.1.1 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 before 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 Dimension Loading Process.

For details on populating the DIM_DATES dimension table, refer to Time Dimension Population.

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

12.4.1.2 Executing the Fact Customer to Customer Relationship Population T2T

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

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

Figure 37: Execute Fact Customer to Customer Relationship Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

   The 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.4.1.3 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$

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

12.5 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, the current status of opportunity, expected revenue, probability of a win, and so on.

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

Table 32: Seeded T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_STG_OPPORTUNITY	STG_OPPORTUNITY	FCT_OPPORTUNITY

12.5.1.1 Prerequisites

The following are the lists of tables used in the population of Fact Customer to Customer Relationship and these tables are required to be loaded before 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 Dimension Loading Process.

For details on populating the DIM_DATES dimension table, refer to Time Dimension Population.

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

12.5.1.2 Executing the Fact Opportunity Population T2T

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

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

Figure 38: Execute Fact Opportunity Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

   The 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.5.1.3 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$

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

12.6 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 33: Seeded T2T 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.

12.6.1.1 Prerequisites

The following are the lists of tables used in the population of Fact Opportunity Activity and these tables are required to be loaded before 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 Dimension Loading Process.

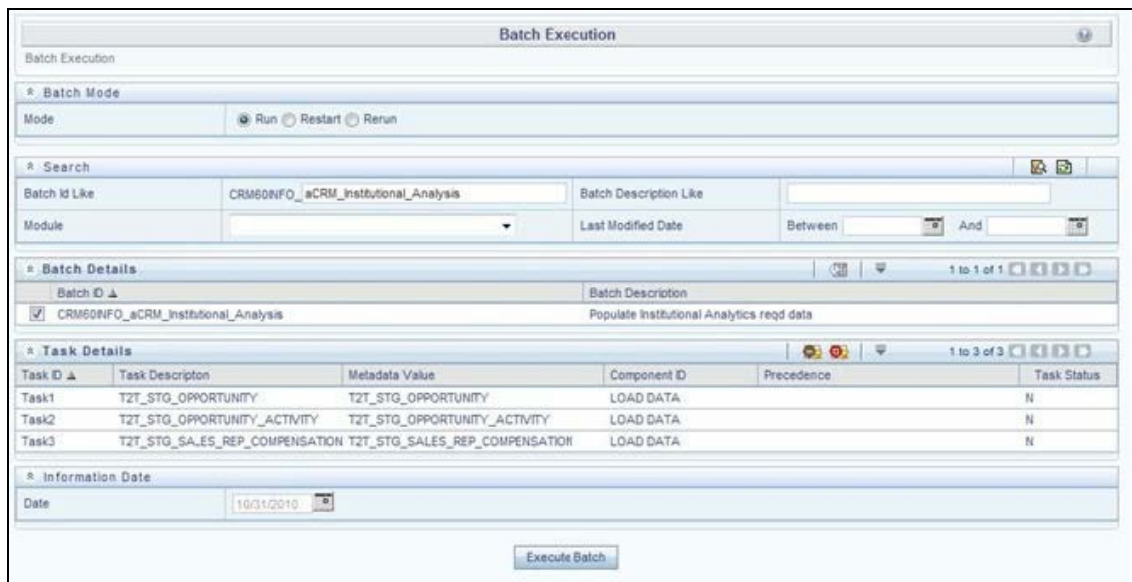For details on populating the DIM_DATES dimension table, refer to Time Dimension Population.

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

12.6.1.2 Executing the Fact Opportunity Activity Population T2T

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

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

Figure 39: Execute Fact Opportunity Activity Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

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

   For example, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.6.1.3 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$

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

12.7 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 34: Seeded T2T Definitions

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 the Customer Insight Erwin Data Model to view the detailed structure of the earlier tables.

12.7.1.1 Prerequisites

The following are the lists of tables used in the population of Fact Sales Representative Compensation and these tables are required to be loaded before 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 the DIM_DATES dimension table, see Time Dimension Population.

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

12.7.1.2 Executing the Fact Sales Representative Compensation Population T2T

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

A seeded process, PR_MANAGEMENT has to be executed for the required MIS Date.

Figure 40: Execute Fact Sales Representative Compensation Population

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

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

   For example, the 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 the [Oracle Financial Services Analytical Applications Infrastructure User Guide](#).

12.7.1.3   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$

For more information on the configuration and execution of a batch, see the [Oracle Financial Services Analytical Applications Infrastructure User Guide](#).

12.8 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 35: Seeded T2T 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.

12.8.1.1 Prerequisites

The following are the lists of tables used in the population of the Fact Application. These tables are required to be loaded before 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 the DIM_DATES dimension table, see Time Dimension Population.

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

12.8.1.2 Executing the Fact Application Population T2T

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded process, PR_APPLICATION has to be executed for the required MIS Date.

Figure 41: Execute Fact Application Population

Alternatively, the 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 to process.

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

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

   For example, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.8.1.3 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$

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

12.9 Account Manager Relation

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

Table 36: Seeded T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_ACCOUNT_MANAGERS_REL	STG_ACCOUNT_MGR_REL	FCT_ACCOUNT_MGR_REL

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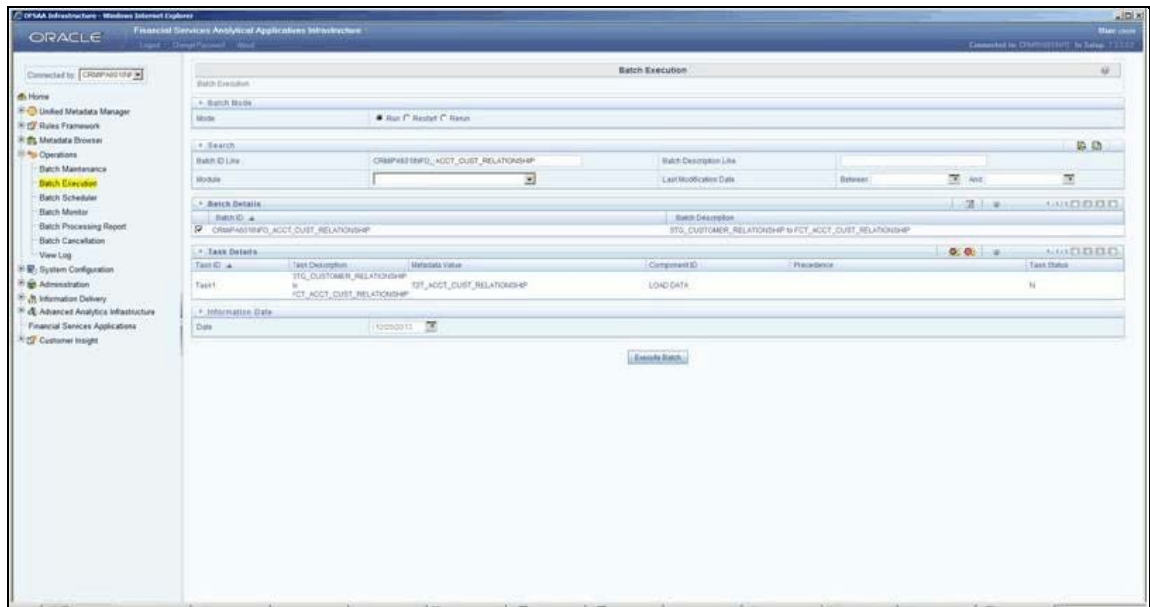
12.9.1.1 Prerequisites

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

- DIM_ACCOUNT
- DIM_CUSTOMER
- DIM_DATES
- DIM_MANAGEMENT
- STG_ACCOUNT_MGR_REL

12.9.1.2 Executing the Account Manager Relation T2T

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded process, PR_ACCOUNT has to be executed for the required MIS Date.

Figure 42: Execute Account Manager Relation

Alternatively, the 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. The data file name will be blank for any Table to Table Load mode.

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

   For example, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.9.1.3 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.

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

12.10 Management Forecast

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

Table 37: Seeded T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_MANAGEMENT_FCAST	STG_MGMT_FORECAST	FCT_MGMT_FORECAST

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

12.10.1.1   Prerequisites

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

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

12.10.1.2   Executing the Management Forecast T2T

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded process, PR_RESULTS has to be executed for the required MIS Date.

Figure 43: Execute Management Forecast

Alternatively, the 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_MANAGEMENT_FCAST' you want to process.

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

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

   For example, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.10.1.3   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.

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

12.11   Fact Account Customer Relation

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

Table 38: Seeded T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_ACCT_CUST_RELATIONSHIP	STG_CUSTOMER_RELATIONSHIP	FCT_ACCT_CUST_RELATIONSHIP

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

12.11.1.1 Prerequisites

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

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

12.11.1.2 Executing the Account Customer Relation T2T

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through Operations module), a seeded process, PR_ACCOUNT has to be executed for the required MIS Date.

Figure 44: Execute Account Customer Relation

Alternatively, the 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.

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

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

   For example, the 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 the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.11.1.3 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.

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

12.12 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 in 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

- FCT_ECO_CAP_ACCOUNT_SUMMARY

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

Table 39: Seeded Post Load Transformation Definition with Source and Destination Tables

DT Definition Name	Source Table	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	

For detailed structures of the tables, see the Oracle Financial Services Analytical Applications Data Model Data Dictionary or the Erwin Data Model.

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

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

The Account summary tables contain the revenue, costs, or expenses measures about 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 needed along with the application installer. Reporting Line Hierarchy and Reporting Line Measure Hierarchy are the two hierarchies that are used for the mapping. Reporting Line Hierarchy is a parent-child hierarchy that is based on the Reporting Line Dimension entity.

Figure 45: Reporting Line Hierarchy
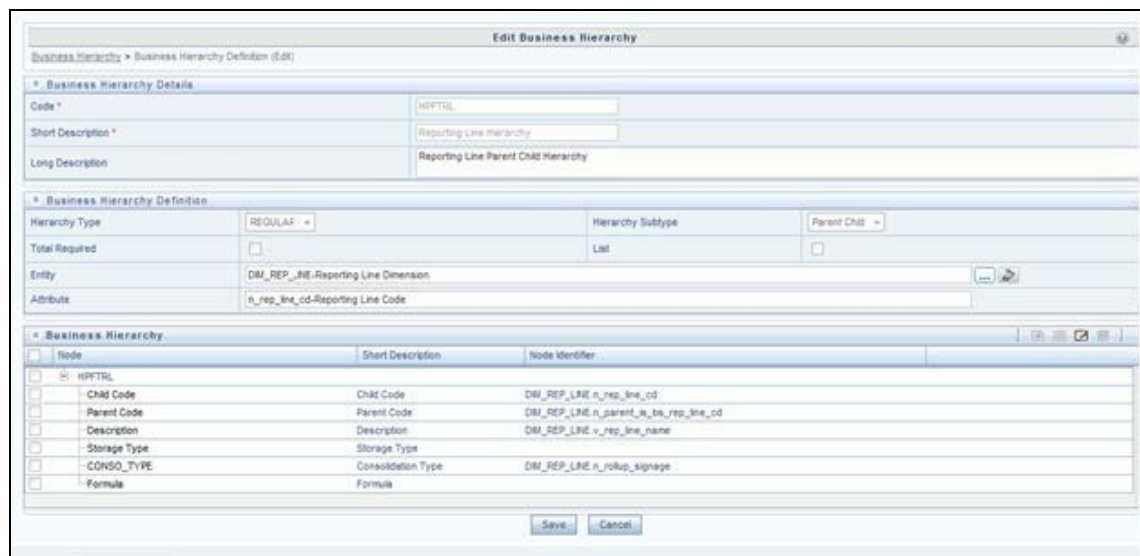
Reporting Line Measure hierarchy is a Non-Business Intelligence Enabled Hierarchy which is based on measures from the Account Summary tables.

Figure 46: Reporting Line Hierarchy

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

Figure 47: Mapper Definition

Figure 48: Mapper Definition - Reporting Line Hierarchy

For more information on defining/maintaining Mapper, see the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.12.1 Defining Mapping for Custom Reporting Line Items

Follow the below steps to define the 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 process PR_REFERENCE_DATA specifying the Reporting line hierarchy as a parameter to batch.

4. Modify the seeded Business Metadata.

5. Map Maintenance.

12.12.2 Adding Custom Reporting Line or Modifying 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 49: 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 50: Members
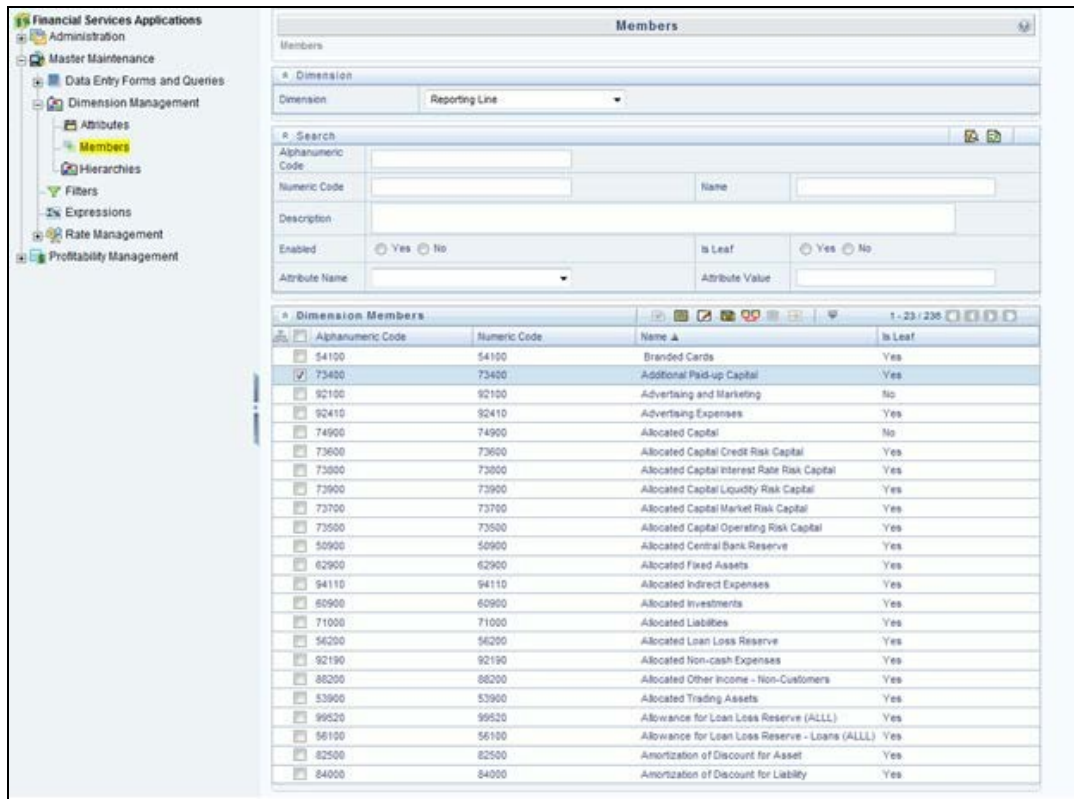
To add a new reporting line, follow these steps:

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

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

Figure 51: Member Definition (New Mode)
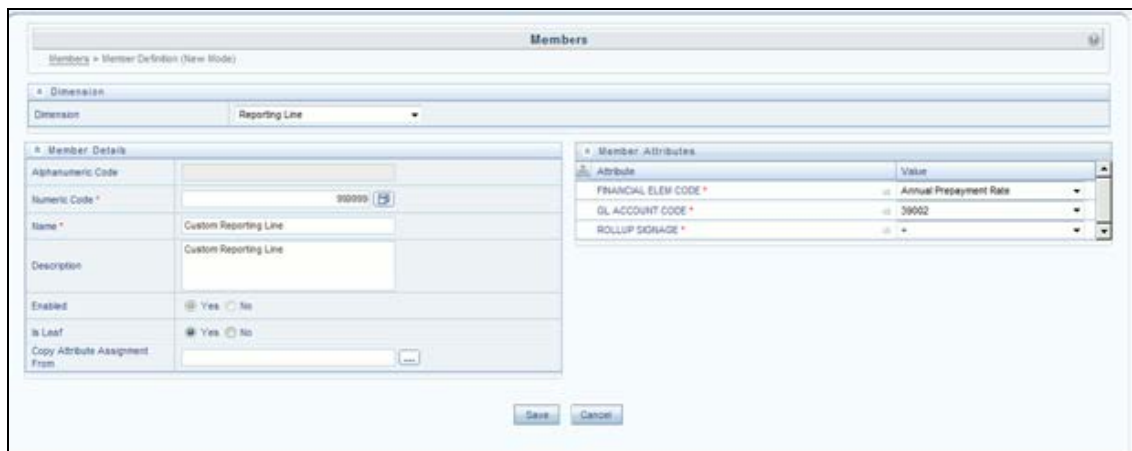
2. In the *Member Definition (New Mode)* screen:

   ▪ 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.

12.12.2.1   Modifying a Reporting Line

To modify a reporting line, follow these steps:

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

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

Figure 52: Member Definition (Edit Mode)

- Follow Step 2 from the above procedure.

To modify a reporting line, follow these steps:

- In the *Member Definition (Edit Mode)* screen, perform the following as required:
  - Modify the **Name** of the custom reporting line.
  - Modify the **Description** of the custom reporting line.
  - Modify the selection of the radio button in the **Enabled** field.
  - Modify the selection of the radio button in the **Is Leaf** field.
  - Modify the **Attributes** for the reporting line member.
- Save the Member definition.

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

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

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

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

Figure 53: Hierarchy Definition (New Mode)

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

Alternatively, insert scripts and update scripts that 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 process PR_REFERENCE_DATA specifying the Reporting line hierarchy as a parameter to batch. It populates data into the DIM_REP_LINE table. This batch invokes the DT fn_rep_line_parent_child.

Figure 54: Repline batch execution
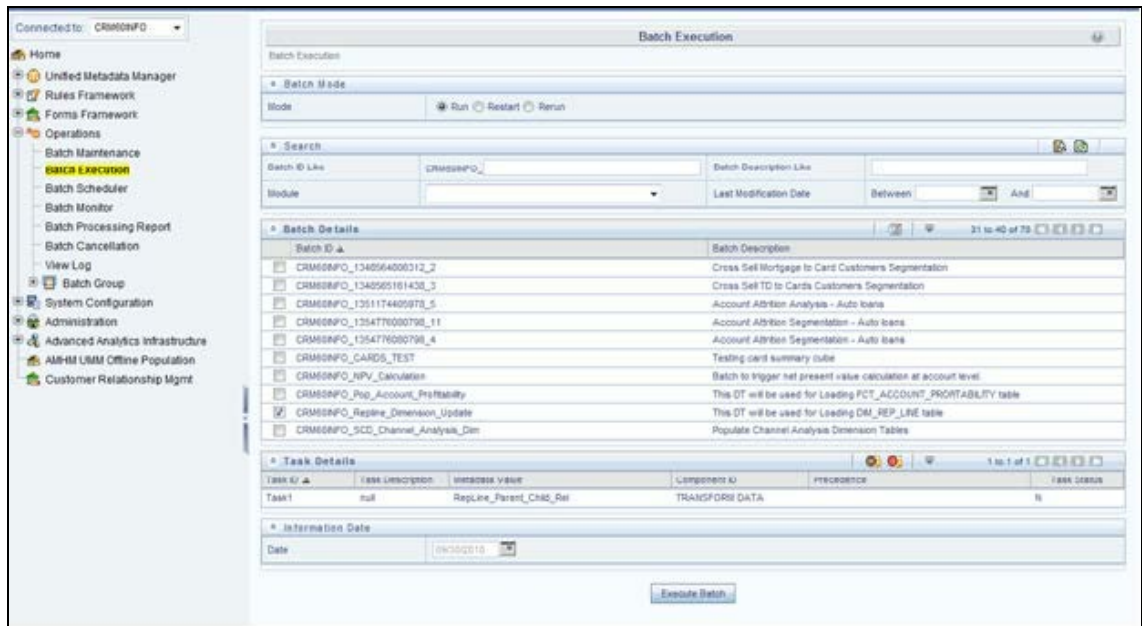
To modify existing seeded Reporting Line Hierarchy:

- Select the check box adjacent to the Reporting Line Hierarchy to be modified.

- Click the **Edit** button from the menu.

- Modify the Hierarchy as required and click **Save**.

Figure 55: Hierarchies
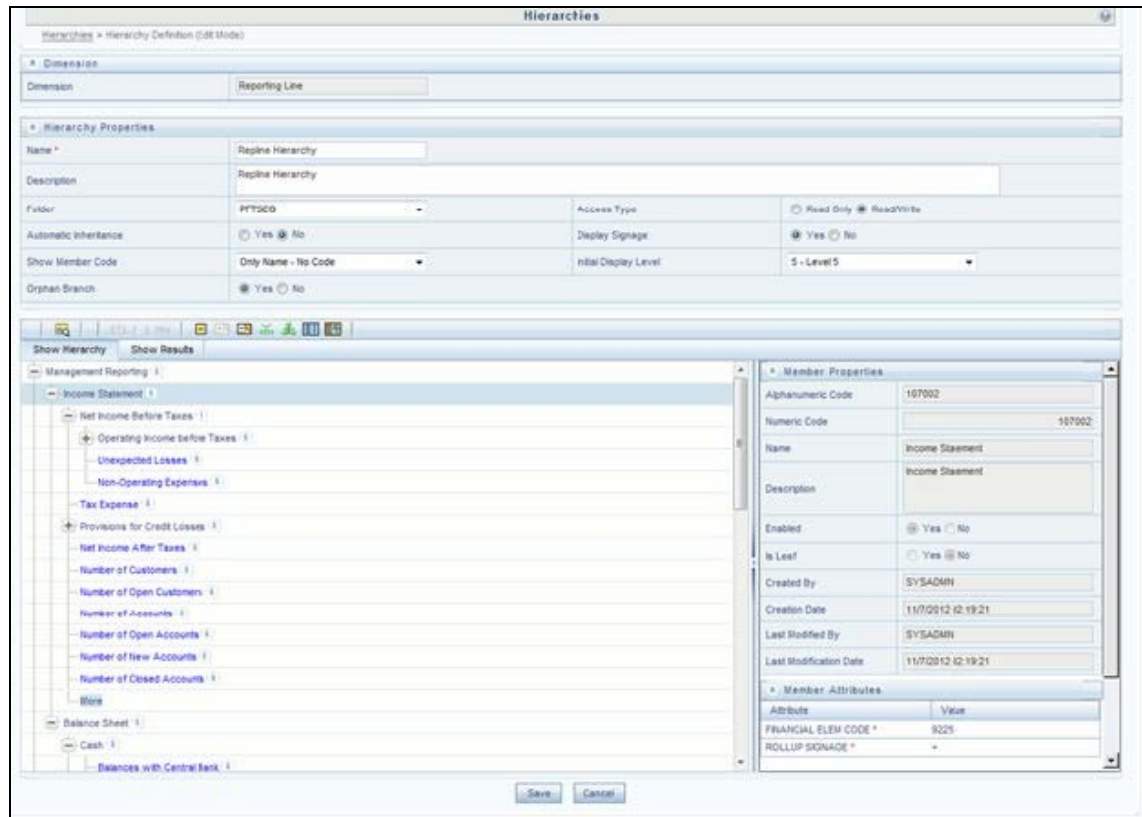
For more information, see the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.12.4 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 56: 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, see the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.12.4.1 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, see the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.12.5 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:

Figure 57: Reporting Line 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:

Figure 58: Reporting Line Hierarchy – Rollup Signage

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, the Rollup Signage of Deposit Insurance is -1 (or negative).

However, about 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.

The Operational Signage of a reporting line is defined as a parent reporting line. However, the Rollup Signage is always in relation to the immediate parent reporting line.

NOTE	Rollup Signage and Operational Signage of Deposit Insurance should be either 1 or -1 and cannot be any other value.

12.12.5.1 Prerequisites

The following are the lists of tables used in the population of Fact Account Profitability and these tables are required to be loaded before 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 Dimension Loading Process.

12.12.6 Executing the Fact Account Profitability Population DT

To execute the DT component from the OFSAAI ICC framework (accessed through Operations module), a seeded process, PR_PROFITABILITY_IPA has to be executed for the required MIS Date.

Figure 59: 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 60: Task Definition

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

The batches PR_PROFITABILITY_IPA and PR_ACCOUNT 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 an error while executing the batch.

For more details, see the Operations chapter in the Oracle Financial Services Analytical Applications Infrastructure User Guide.

12.12.6.1 Checking the Execution Status

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

For more comprehensive coverage of configuration and execution of a batch, see the Operations chapter in the 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.

12.12.7 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 has 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 has 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 61: Seeded Run Rule Framework
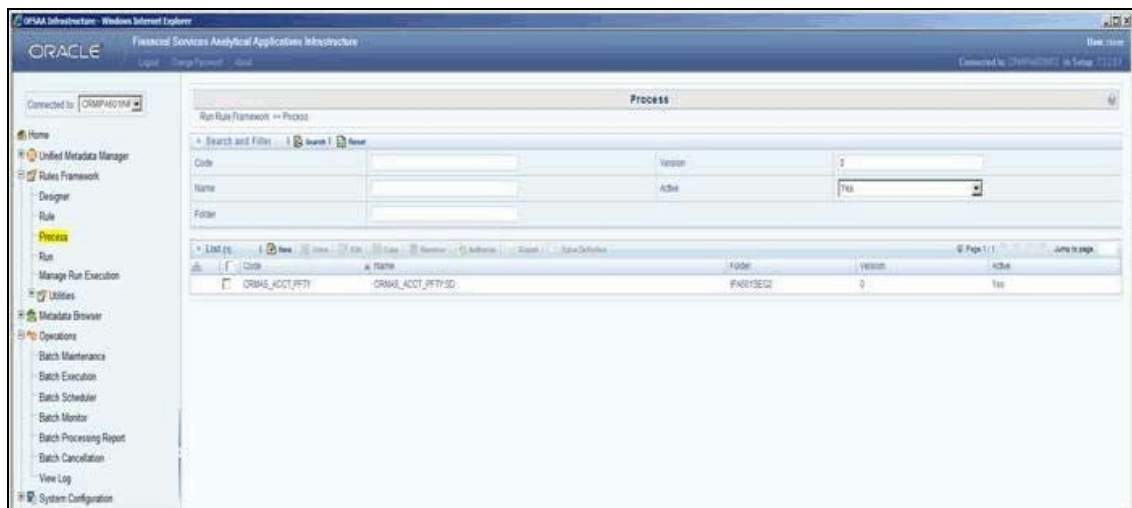
2. Edit the process and click the "component" option.

Figure 62: Seeded Run Rule Framework
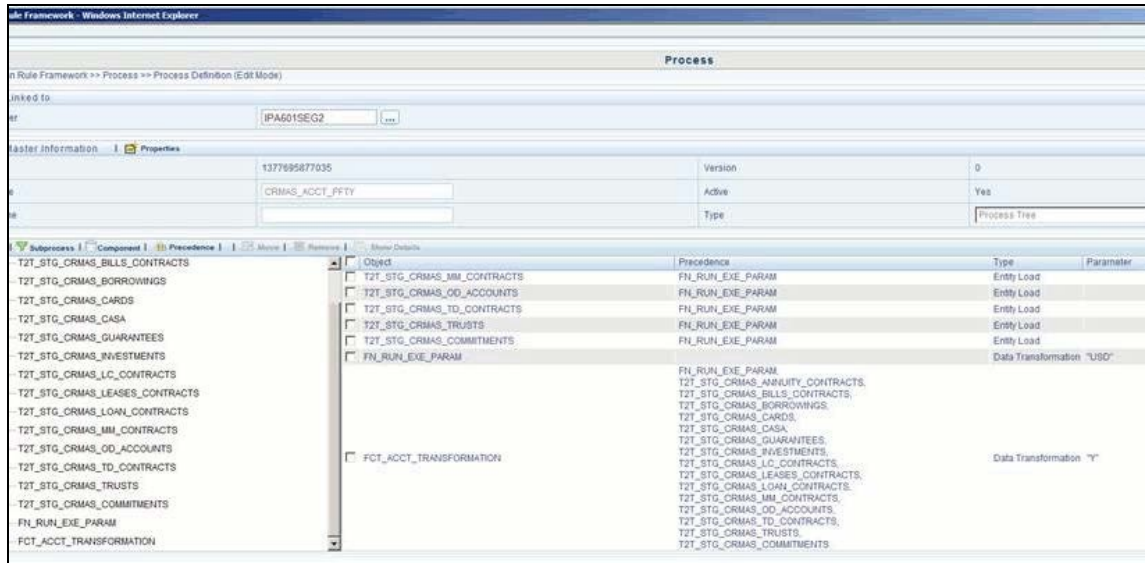
3. From the list of tasks available in the right pane, click the arrow present near the "FN_RUN_EXE_PARAM" task.

4. Feed-in the currency code of the Reporting Currency.

Figure 63: Reporting Currency
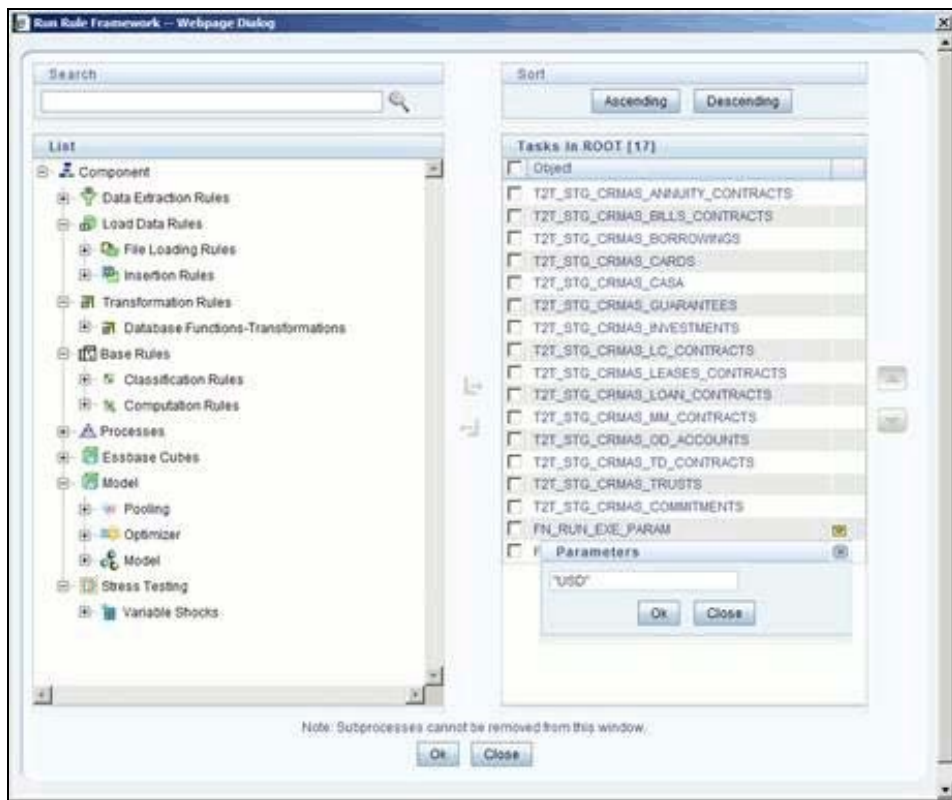
5. From the list of tasks available in the right pane, click the arrow present near the task by the name "FCT_ACCT_TRANSFORMATION".

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 the "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.

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

13   Cube Build Process

Reports of OFS RPA application can be configured to work on a Relational database or Essbase cubes. The 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 an optional source of data for the Retail Performance application.

OFS RPA application has the following seeded cubes:

- Cube for Financial Reporting - ADPAFNRE

  - Purpose: The purpose of this cube is to store data for reports belonging to Financial Reporting Dashboard.

  - Dataset: DSPAFR

    This cube is based on the FCT_LEDGER_STAT fact table.

- Cube for Management Reporting - ADPAMNRE

  - Purpose: The purpose of this cube is to store data for reports belonging to Management Reporting Dashboard.

  - Dataset: DSPAMR

    This cube is based on the FCT_MGMT_REPORTING fact table.

Topics:

- [List of Cubes in RPA Applications](#)
- [Creating Configuration Files](#)
- [Building Cubes](#)

13.1   List of Cubes in RPA Application

OFS RPA application is seeded with the following cubes:

Table 40: Seeded Cubes for OFS RPA Application

Cube	Purpose	Dataset
Institutional Analysis	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.	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.

Cube	Purpose	Dataset
RM P and L Cube	The purpose of this cube is to provide details of the Profit and Loss statement of a Relationship Manager across dimensions like Line of Business, Product, Organizational Unit, and so on.	This cube is based on the FCT_ACCOUNT_PROFITAIBILTY and FCT_ACCOUNT_MGR_REL fact Tables.

13.2 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 the 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. To trigger the change event, type a blank space in the 'Long Description' textbox and remove the same. Or a dimension can be removed from the selected list, again the same dimension re-selected, variation applied for the dimension and saved.

13.3 Building Cubes

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

- 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.

- Creating the cube outline on the Essbase server. This is performed by the CREATE CUBE component task within the batch definition.

- Loading the data to the cube. This is performed by the CREATE CUBE task within the batch definition

13.3.1 Prerequisites

Following are the prerequisites for creating a cube:

1. All the post-installation steps mentioned in the Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration 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.

    ▪ Go to **Home > Administration > Save Metadata**.

    ▪ Choose all the available metadata under Hierarchy and move it to the right by using the '>>' button.

    ▪ Click **Save** and might take a few minutes for the saving to complete.

    ▪ Click **Show Details** to view the log for the Save operation.

5. Ensure that the following services are running on the application server before doing a cube build:

    ▪ Iccserver

    ▪ Router

    ▪ AM

    ▪ Messageserver

    ▪ Olapdataserver

6. Batches need to be created for executing, which is explained in Executing the Cube build section.

7. All the required tables for the 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.

8. 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:

    a. Navigate to **Home > Unified Metadata Manager > Business Metadata Management > Data Sets**.

    b. Click **Search**.

    c. Click any dataset in the pop-up which opens and click **Ok** to return to the data set screen.

    d. Click the button on the right of the ANSI Join text box. Enter the required expression or click the below button to define an expression using the Expression screen.

    e. Click **OK** to return to the data set screen.

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

10. 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 a record count greater than zero when you fire this from the SQL prompt in the database.

11. Essbase Reserved Keywords: Essbase has few reserved keywords and 'Missing' is one of them. So in the case of cube building using dim_bands, it fails as it tries to create a cube with the reserved keyword in the table data. None of the reserved keywords should be used while creating data.

> **NOTE**   Essbase does not identify/ support the newline character. Ensure that you do not use them.

For the list of reserved keywords, please see the following link:

https://docs.oracle.com/cd/E51367_01/financialsop_gs/FASVU/apbs02s02s03.html

12. For the ADIPARM2 cube, check all the Variation Dimension checkboxes for the MCRMSIAM and MCRMDIMV measures from the Essbase cube details> Variation tab and resave the ADIPARM2 cube.

13.3.2 Tables Used by the Cube Build Component

Tables that are part of the dataset need to be populated before executing the cube build component. Also, the REV_BIHIER table in the 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 the Save Metadata screen.

13.3.3 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.

13.3.4 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.

13.3.5 Create a Cube Task

1. In the batch created in the 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 follow or the SQL query in the 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 OFSPA seeded batches:

13.3.5.1 RPA Cubes

- **Retail Analysis**: Seeded process **PR_CUBES_RPA** is provided with the installer. Execute the batch for the required MIS Date.

- **RM P and L Cube**: Seeded process **PR_CUBES_RPA** is provided with the installer. Execute the batch for the required MIS Date.

Figure 64: Batch Execution
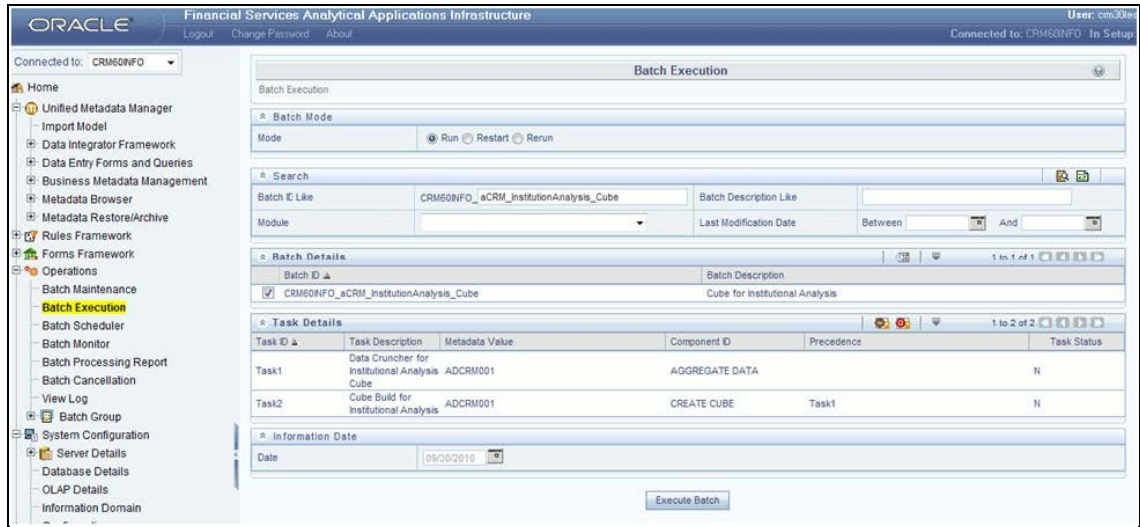
13.4   Checking the Execution Status

The status of execution can be monitored using the Batch Monitor screen. To access the Bath Monitor screen, select **Home** from the **LHS Menu**, select **Operations**, and then select **Batch Monitor**.

For more comprehensive coverage of configuration and execution of a batch, see the 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 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 Task 2 above (Create Cube). The file name will have the Batch Execution ID.

NOTE	See How to Develop a New Cube for information 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.

14   Time Series Forecasting

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 "arima()". This function enables the usage of the ARIMA technique for time series forecasting.

Projected data is generated through statistical modeling. ARIMA/ARIMAX modeling is used to create the projected data for up to five years. Historical data for the last two years is used for creating the projections. The projections are 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.

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

- If the data provided is not for the chronological end-of-month dates, the 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 the Moving Average process, and d is the number of differences after which the data can be considered stationary with the desired confidence level. It has more parameters that can be customized.

    Detailed documentation of the technique can be found at:

    - http://www.dms.umontreal.ca/~duchesne/chap7.pdf

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

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

- Logging for the individual groups' Arima model summaries happens in a file named "out.log". This log file is generated in the $FIC_DB_HOME/bin folder. If a particular group has unstable data and prediction fails, the 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 the $FIC_DB_HOME/bin folder and need to 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. The model is trying to apply an algorithm on a dataset that is returning a 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 and if a condition is receiving null and is unable to evaluate true or false.

15   Segmentation

Segmentation is the procedure of grouping together a set of customers based on certain similar features. These customers grouped 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. Based on the characteristic around which segmentation is to be created, the dimensions used for segmentation can vary.

Following are the segmentation types that are being supported for the OFS RPA application:

- Corporate Tracker Segmentation

- Profitability Segmentation

- Risk-Based Segmentation

- Behavioral Segmentation

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.

15.1.1.1   Institutional Analytics Segmentation

Segmentation is done using the following dimensions for Institutional Analytics:

- Year of incorporation

- Status of listing

- Income

- Industry

- Country of incorporation

- Group asset size

Following is the list of Product Types used in Institutional Analytics segmentation:

- CASA for Current and Saving Accounts

- TD for Term Deposits

- LOAN for Loan Contracts

- CARDS for Credit Cards.

15.1.1.2   Retail Analytics Segmentation

These scores and segments will be stored in a new fact table. The inputs for this model will come from the following tables:

- Location

- Credit Rating

- Currency

- Geography

- Industry

- Product

- Age

- Income

- Gender

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

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

Table 41: 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 42: 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

Column Name	Logical Name
RUN_SKEY	Run Key
MONTH_ON_BOOK	Month on Book
ACCT_SEGMENT	Account Segment

DIM_SEGMENT_TABLE will be populated using SCD Process (Map reference number 267) and the source would be a view of 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 a 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.

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.

15.2   Creating a rule

To define a rule, follow these steps:

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

Figure 65: Run Rule Framework - Rule
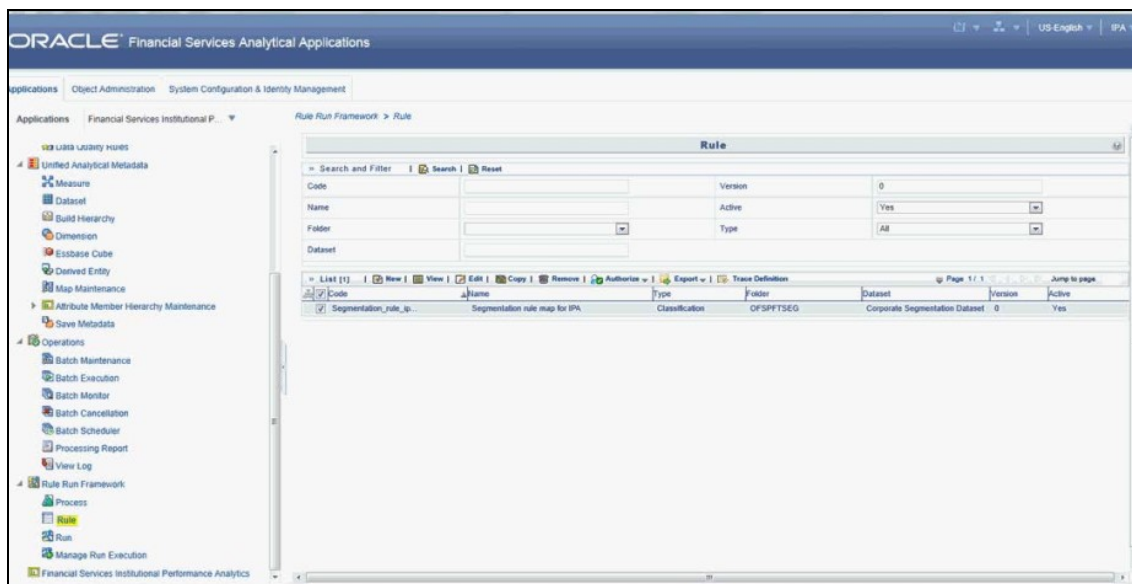
2. Select the rule and click **View**. The following screen appears.

Figure 66: Rule Definition - View Mode
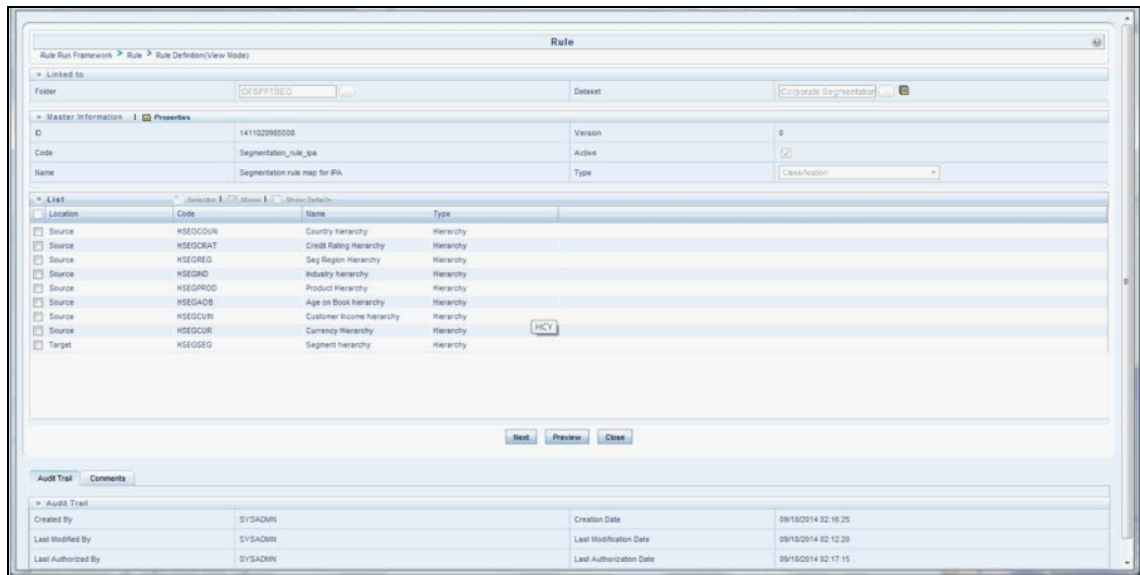
3. On clicking **Next**, the rule defined comes up. For the first time when the rule is not defined, only the default seeded node rule should appear as shown in the following screen:

Figure 64: Rule Definition – View Mode
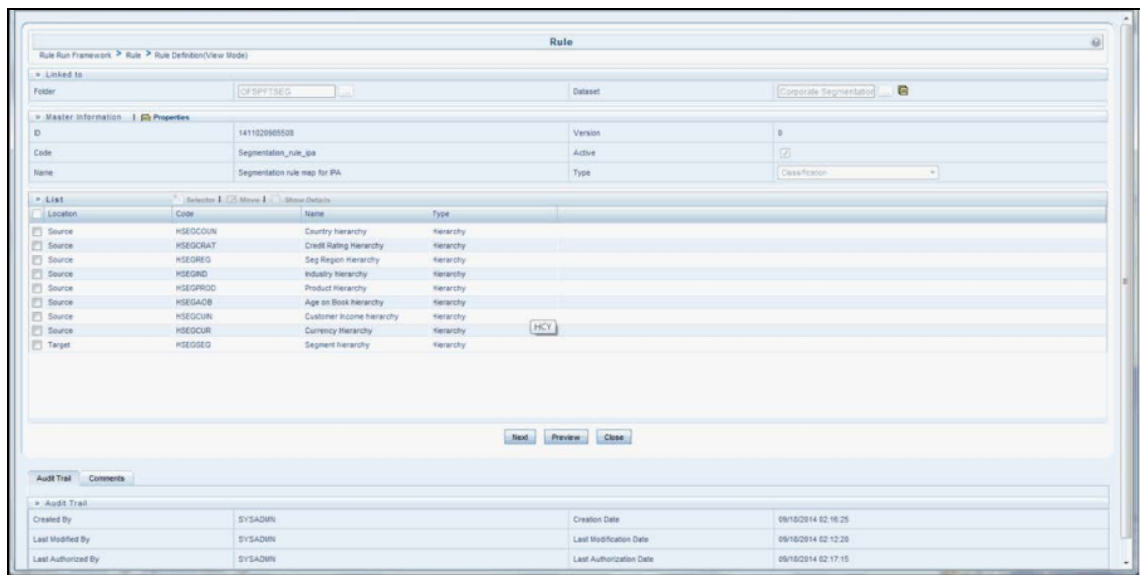
15.3 Editing a rule

To edit a rule, follow these steps:

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

Figure 67: Rule Definition – Edit Mode
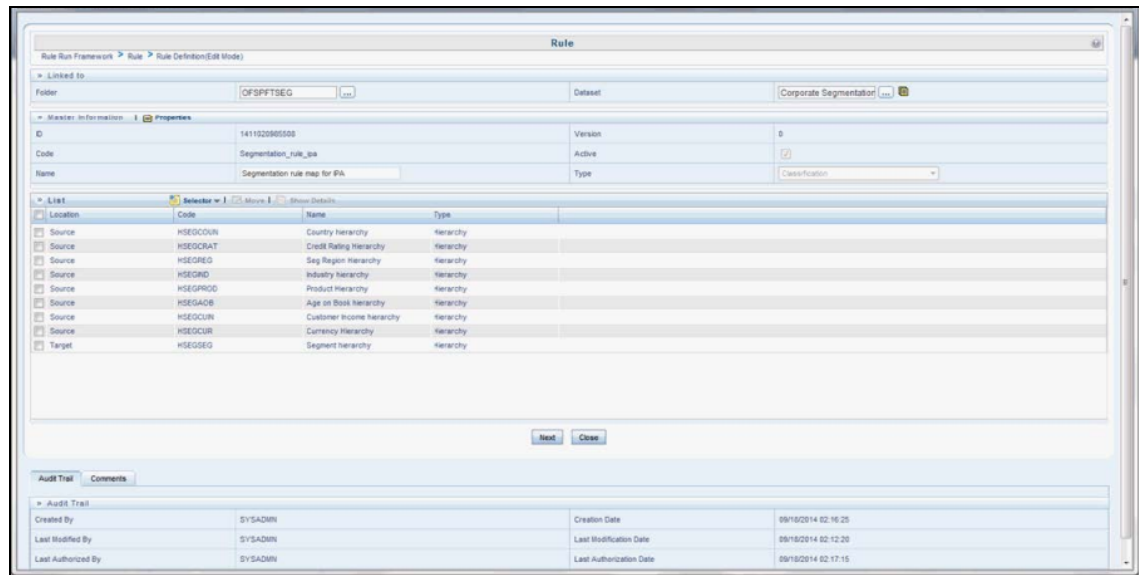
2. Click **Next**. The first-time default node defined as the rule will show up in the following figure:

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

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

Figure 68: Hierarchy Selection
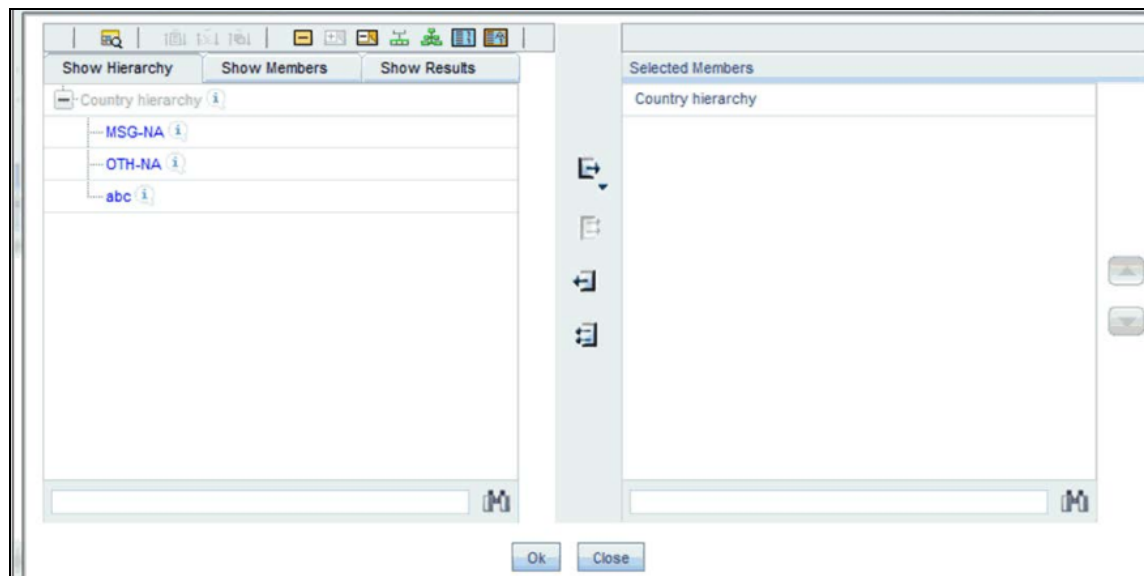
5. The selected node appears in the rule.

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

7. Click **Close**.

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

16 Overview of Retail Performance Analytics Reports

Oracle Financial Services Retail Performance Analytics (OFSRPA) offers the following dashboards that organize different kinds of reports by subject area.

These reports provide the following:

- Gain deep insight into customer engagements across target segments and products/LOB including lending, credit cards, and so on.

- Perform Wallet share analysis and Customer Profitability.

- Understand the efficiency of investments (like marketing, branch, and channel, and so on) over time.

- Monitor customer distribution across credit and delinquency bands and related exposures.

- Perform an enterprise-wide revenue analysis across customer segments, products, and reporting lines including fee income, interest, and interchange.

- Summary performance of the LOBs, overall Profitability, and Portfolio mix.

- Customer trends across performance drivers like Sales, Balances, Deposits, Product subscriptions (revenue services), Credit scores, Delinquency bands, Losses, and so on.

- LOB-specific performance reports can be analyzed against key dimensions like customer segments, product family, region, branch, risk scores, and so on.

- Analyze expenses across customer segments, products, and channels to understand ROI.

For all dashboards, there are filters available. You can filter data based on year, region, line of business, and currency type.

16.1 Dashboards

OFSRPA has the following tabs present in the Retail Performance dashboard:

- Business Analysis
  - Performance Summary
  - New Business Analysis
  - Revenue Analysis
  - Expense Analysis
  - Credit Loss Summary
  - Margin Reports
  - Customer Summary
  - What-If Analysis
- Customer Central
  - Customer 360

- ▪ Customer Performance
- Product Summary
  - ▪ All Product
  - ▪ Cards
  - ▪ Retail Bank
  - ▪ Mortgage
- Relationship Manager Insights

The following sections display the essential nature of the available reports as per each tab.

16.1.1 Business Analysis

Business Analysis Dashboard provides in-depth analysis on enterprise-wide performance evaluation based on critical business parameters. The dashboard contains various tabs containing analytical reports on business performance, business acquisition, revenues, expenses, margins, credit loss, customers, and sensitivity analysis like what-if analysis.

16.1.1.1 Performance Summary

16.1.1.1.1 Portfolio Mix

This report provides details about customers distributed among various region along with the Line of Business. The report can be analyzed through various business parameters like; the number of new and open accounts, revenues, spends, and so on. This is a zoomable map report.

Figure 69: Portfolio Mix
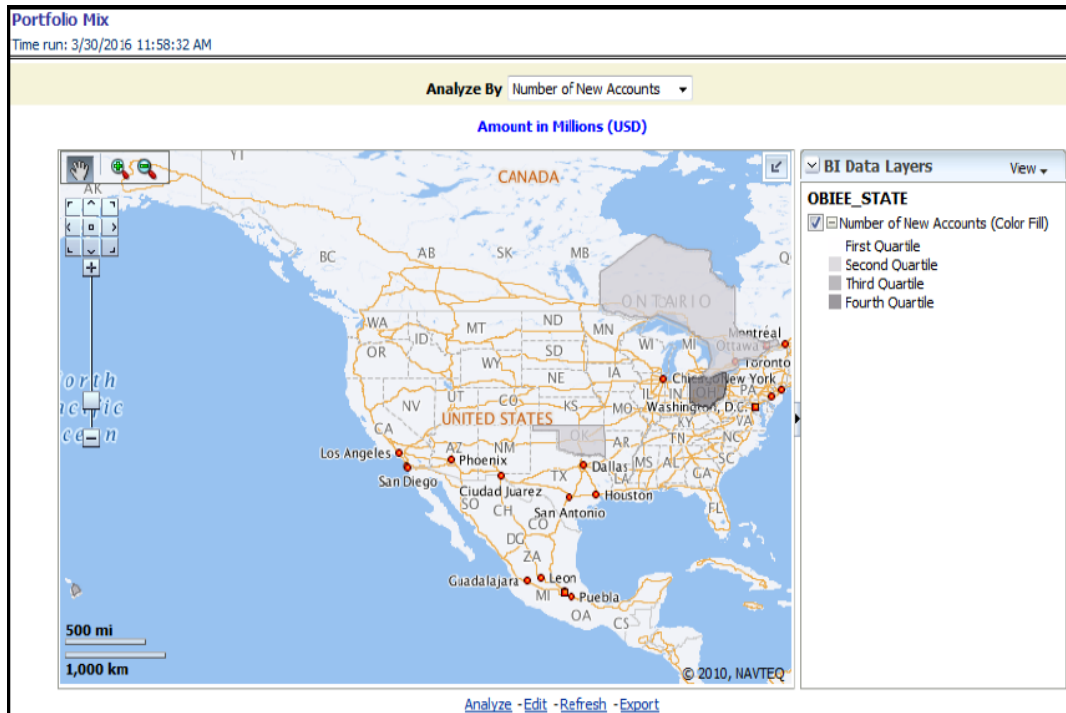
The map can be zoomed in and out by clicking on the zoom scale or the zoom in/zoom out button. You can navigate through the map by dragging the map or clicking on the Zoom icon.

16.1.1.1.2   Portfolio Mix

This report displays the growth of key metrics such as No. of New Accounts, End of Period balance, and Sales across various Lines of Businesses within the bank. This is a sunburst report.

On clicking a particular distribution, a consolidated view of that particular distribution is displayed. For example, if you need to get a consolidated view of 2011 distribution, click on 2011. You get consolidated data of 2011 distribution on open customers, Line of Business, constituent products, and corresponding revenue. Further, you can get a consolidated view of open customers, Line of Business, constituent products, and corresponding revenue, by clicking any particular distribution. To view the overall report, click the center of the circular graph.

Figure 70: Portfolio Mix – Sunburst Report
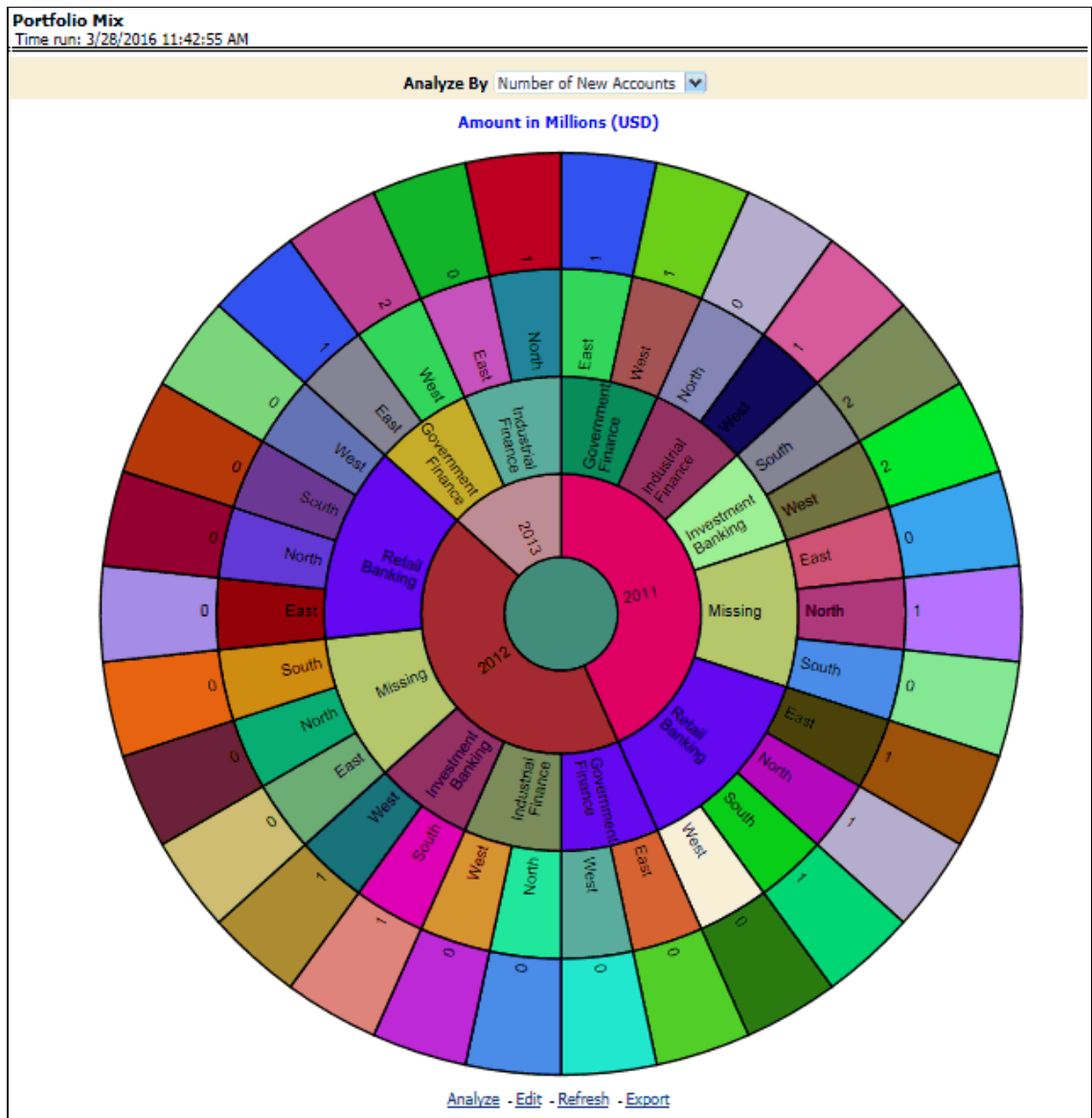
16.1.1.1.3 Revenue Distribution

This report provides the Revenue spread across the different Lines of Businesses within the bank.

Figure 71: Revenue Distribution
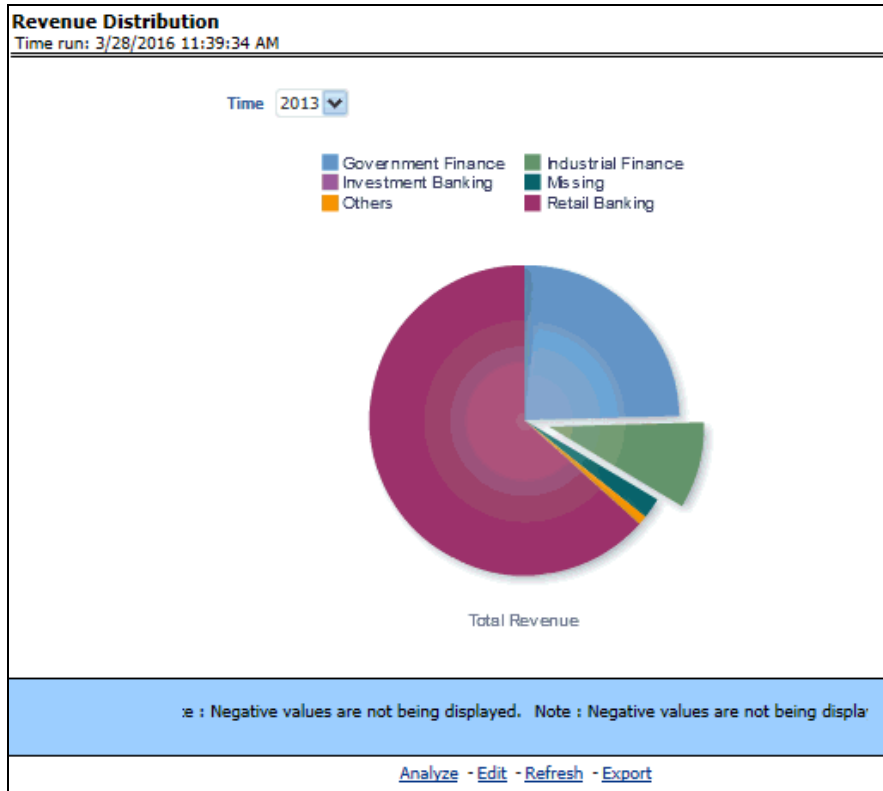
16.1.1.1.4 Cross LOB Holding

This report outlines the product holding patterns of bank customers across Lines of Business. It shows the relationships the customer has across the enterprise.

Figure 72: Cross LOB Holding

16.1.1.1.5   Top 10 Products by Revenue

This report outlines the top 10 products ranked by Revenue and percentage contribution to the total business. This report also lists top-performing products belonging to the line of businesses and the number of customers each product is holding.

Figure 73: Top 10 Products by Revenue

16.1.1.1.6   No. of Accounts by Region and Product

This report displays the concentration of Accounts across various Regions and Products within the bank.

Figure 74: No. of Accounts by Region and Product
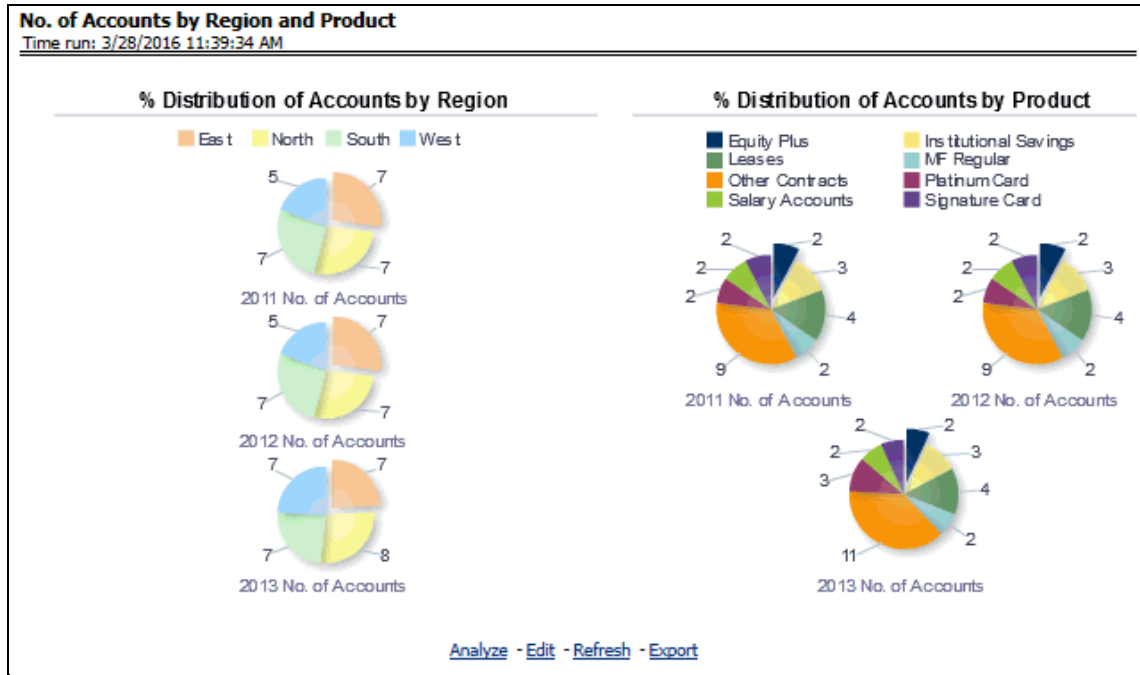
16.1.1.1.7   Summary of New Customers

This report displays the growth in customer base across the various products over time.

Figure 75: Summary of New Customers
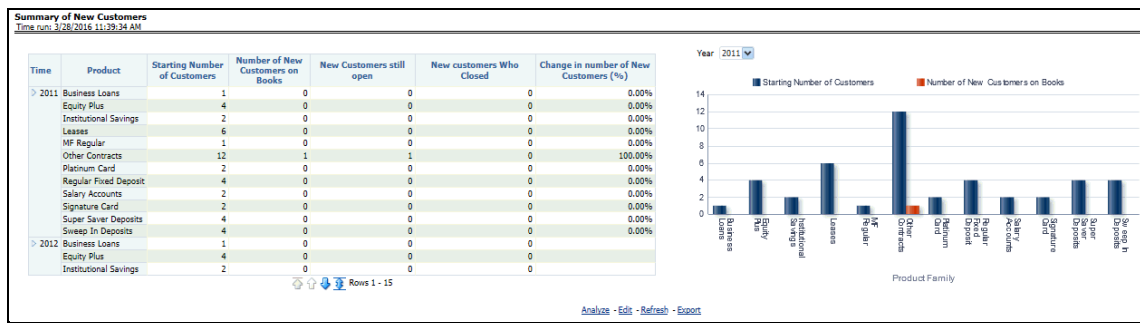
16.1.1.1.8   New Business Summary by Channel

This report displays a summary of new accounts opened across various bank channels.

Figure 76: New Business Summary by Channel
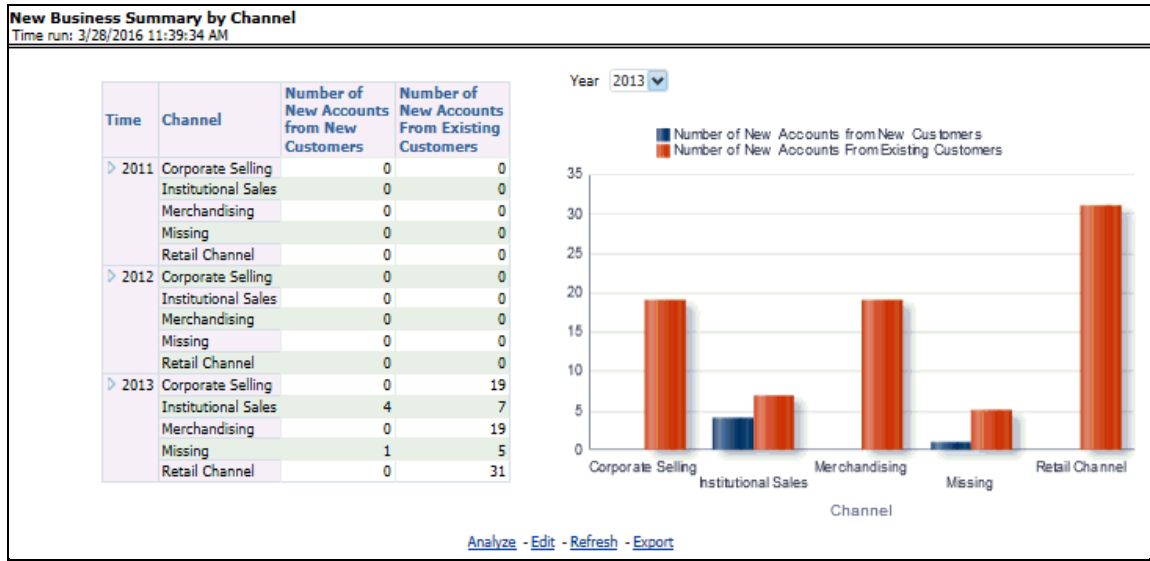
16.1.1.1.9 New Business Summary by Product

This report displays a summary of new customers on-boarded by product.

Figure 77: New Business Summary by Product
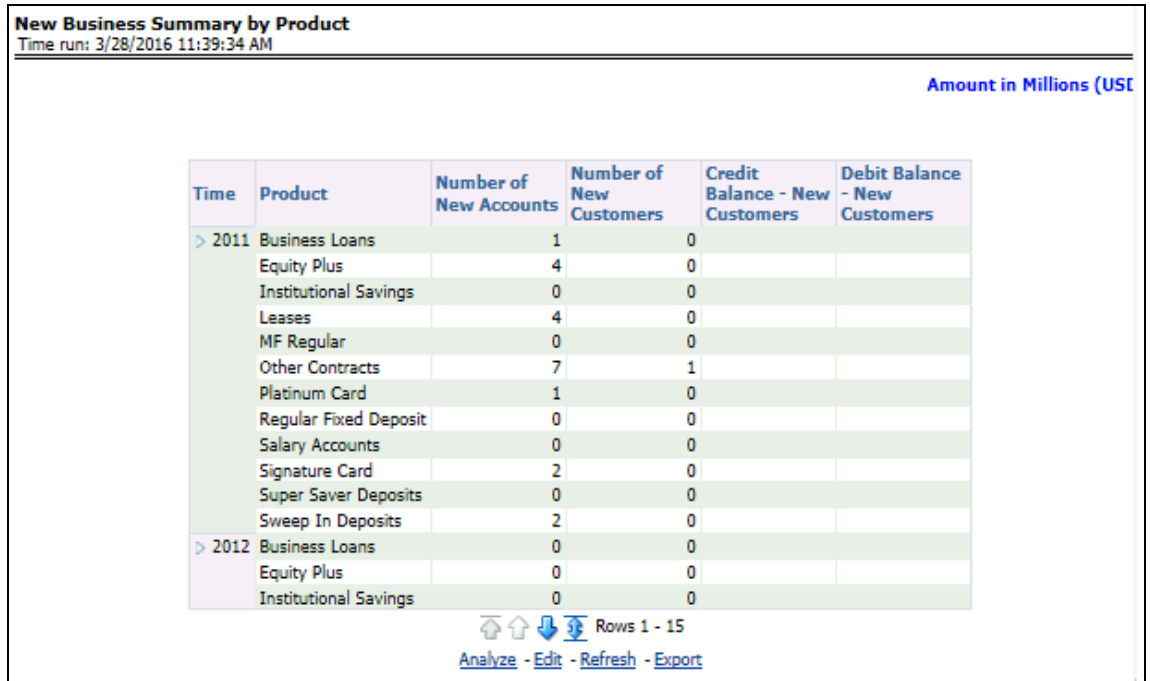
16.1.1.1.10 Summary of Closed Accounts

This report displays the number of accounts closed across products and attrition reasons.

Figure 78: Summary of Closed Accounts
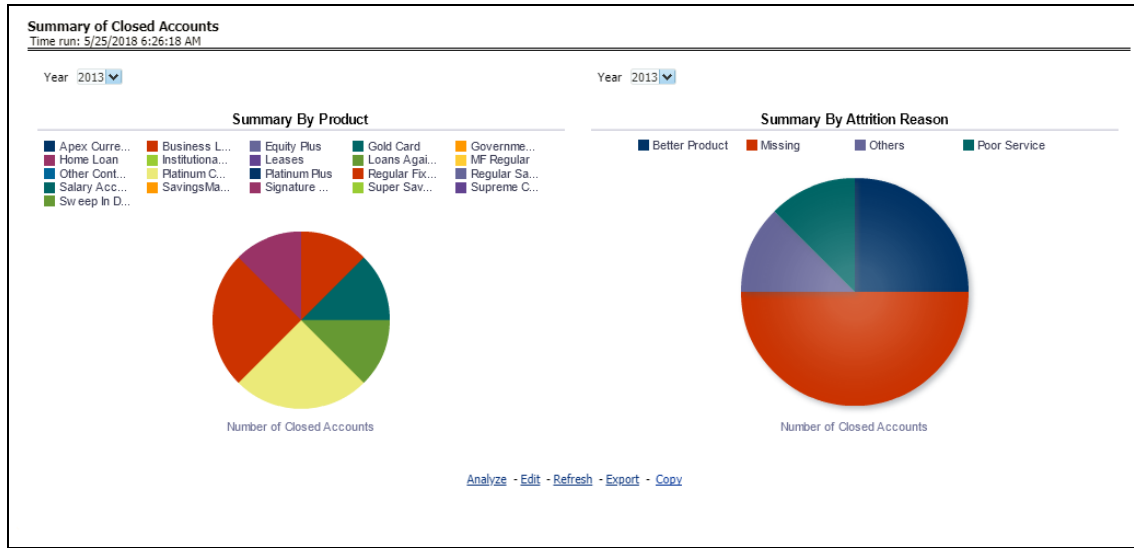
16.1.1.1.11   Origination Channel Performance

This report displays Net Income generated by different sales and business acquisition channels.

Figure 79: Origination Channel Performance
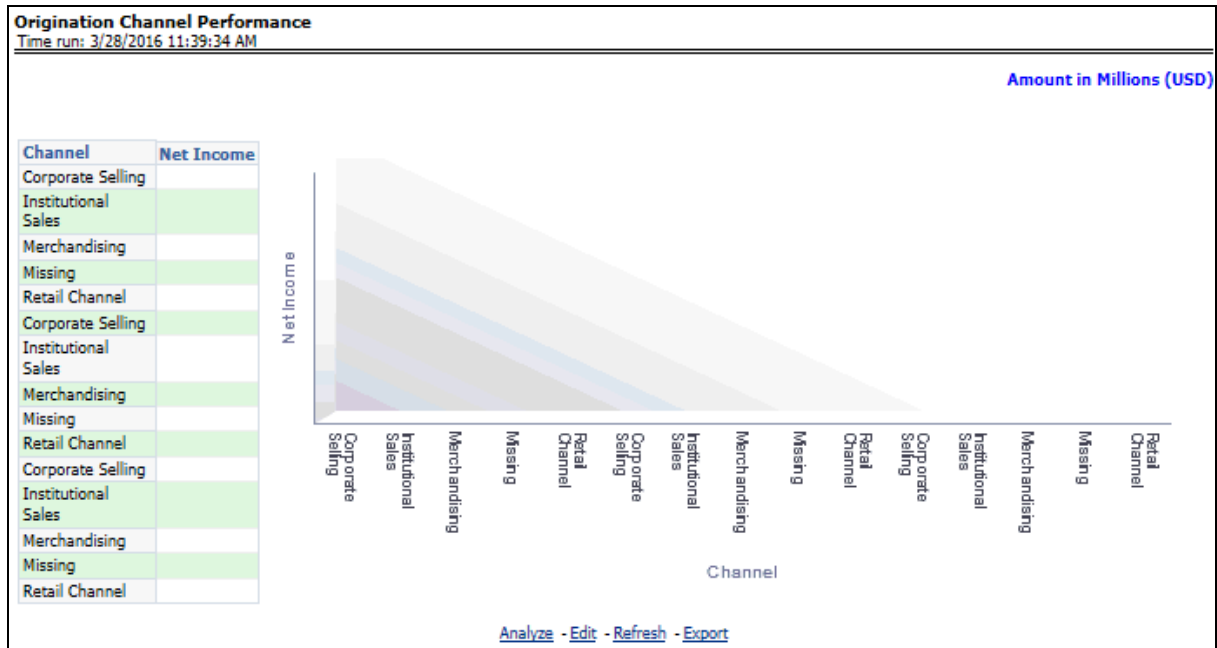
16.1.1.2   New Business Analysis

16.1.1.2.1   Acquisitions Over last 5 Years

This report displays the growth of accounts and customers over a period. The dashboard prompts allows the user to narrow this analysis down to a specific LOB, Product Family, or Product.

Figure 80: Acquisitions Over Last 5 Years
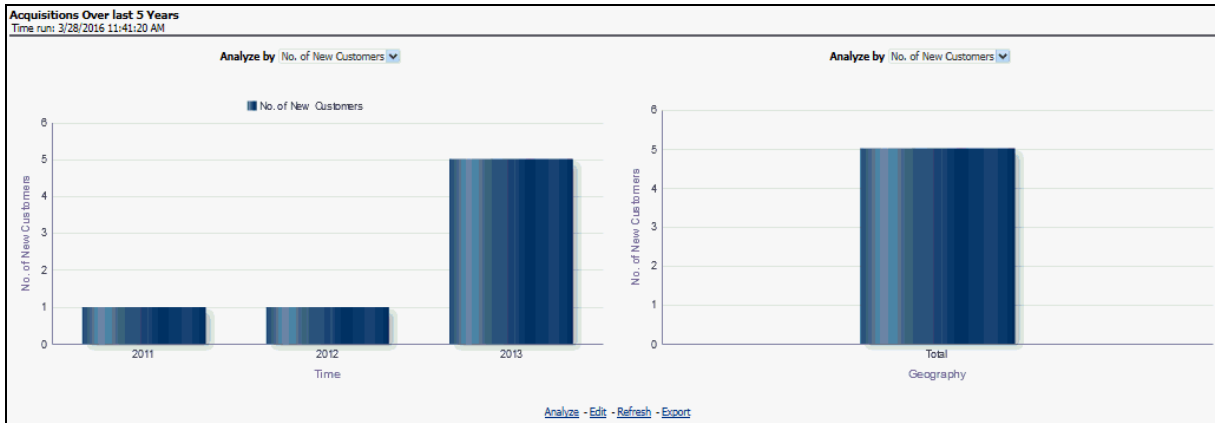
16.1.1.2.2   Approval Trends across Channels

This report displays variance in approval rates over time across the various bank channels and enables you to track how these rates have gone up or down in specific channels or compare a rise/fall in one channel vs performance in others. The dashboard prompts allows the user to narrow this analysis down to a specific LOB, Product Family, or Product.

Figure 81: Approval Trends across Channels
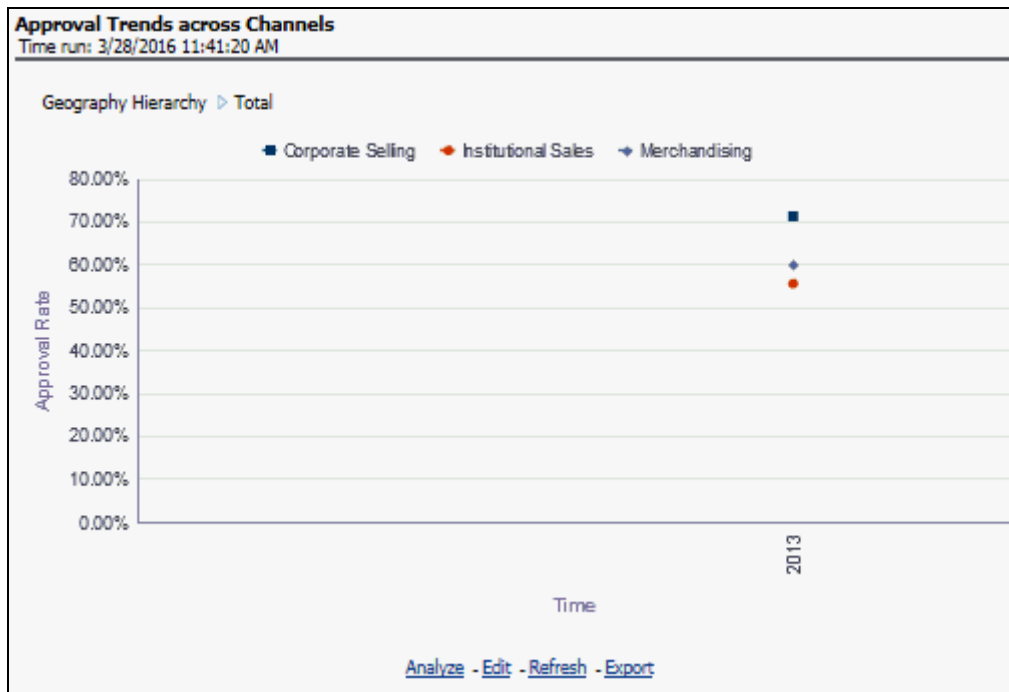
Line of Business Filter prompt selection is not applicable for this report.

16.1.1.2.3   New Account Distribution by Customer Segment

Provides a composition of the customers across key customer segments like Age, Gender, and Income.

Figure 82: New Account Distribution by Customer Segment
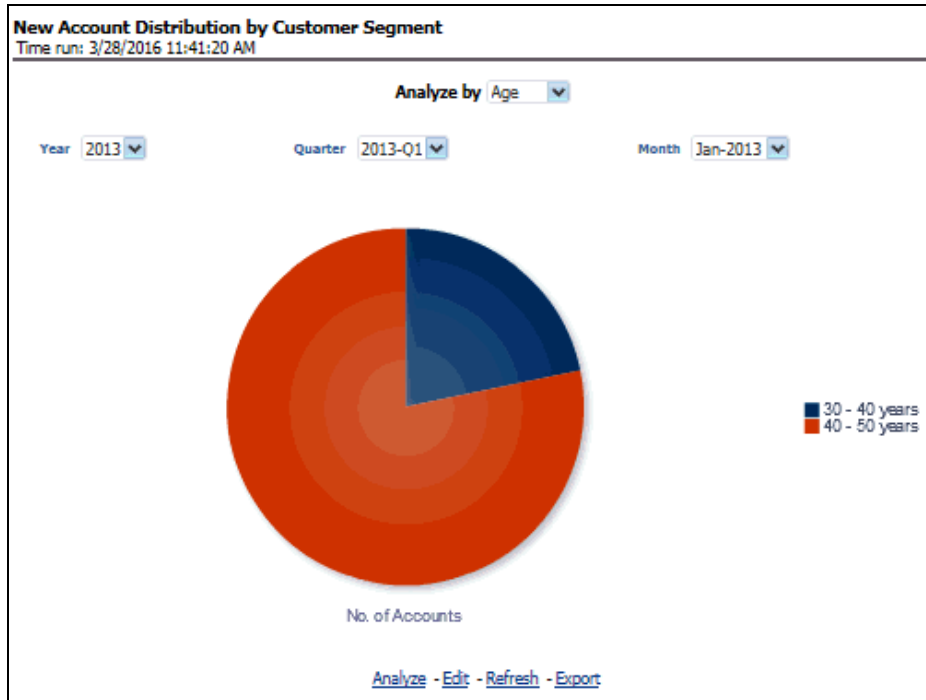
Product filter prompt selection is not applicable on this report.

16.1.1.2.4   Acquisition by Channel

This report displays how acquisitions have migrated from one channel to another over a period.

Figure 83: Acquisition by Channel
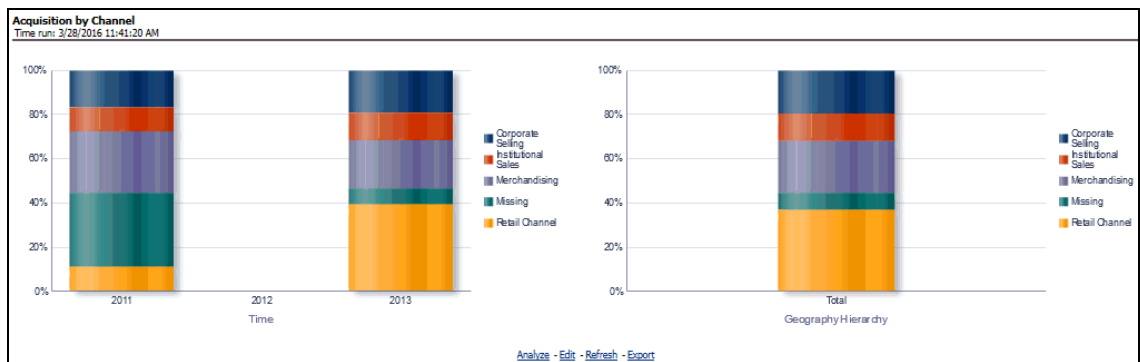
16.1.1.2.5   Overall Attrition Over last 5 Years

This report enables the user to have a view of the outflow (closed customers) or attrition over the last 5 years to fully understand how customers are flowing in and out of a certain product or product family or LOB.

Figure 84: Overall Attrition Over Last 5 Years

16.1.1.2.6   Top 10 Reject Reasons

This report displays a variance analysis of the various reasons why an application was rejected. This can also be viewed as a time series to see trends over a period.

Figure 85: Top 10 Reject Reasons

Rejection Reason	No. of Applications	% Total
NO PROPER DOCUMENTATION	4	50.0%
NOT ELIGIBLE TO APPLY	4	50.0%
Grand Total	**8**	**100.0%**

Line of Business Filter prompt selection is not applicable for this report.

16.1.1.2.7   Account Distribution by Credit Band

This report outlines the number of accounts booked across various credit score bands. The credit score corresponds to the customer holding the account.

Figure 86: Account Distribution by Credit Bond
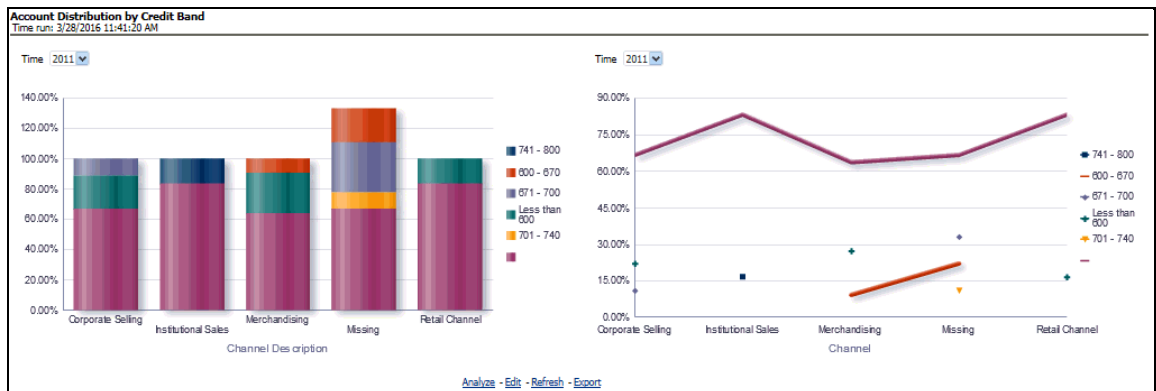
16.1.1.3　Revenue Analysis

16.1.1.3.1　Revenue Analysis

This report displays variance analysis of the various Revenue components like Fee Income, Net Interest Revenue, and so on by Product with the ability to further understand this distribution across key customer segments like Gender, Age, and Income.

Figure 87: Revenue Analysis
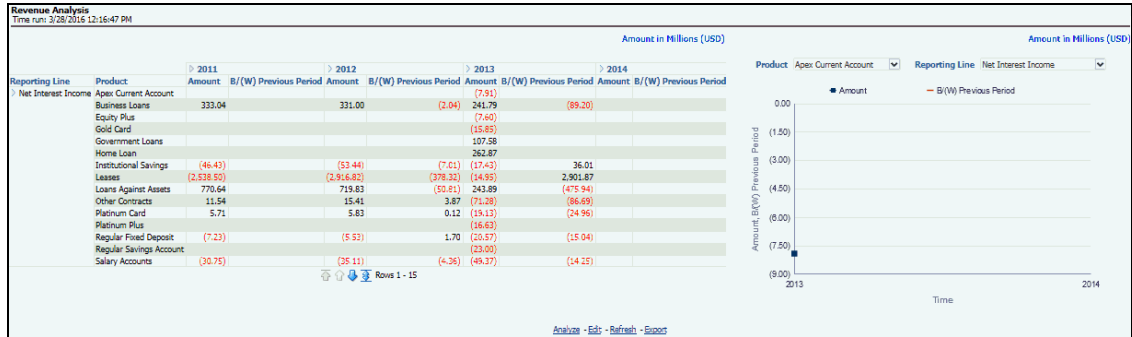
16.1.1.4　Expense Analysis

This report displays variance analysis of the various Expense components like Direct Expenses, Operating Expenses, and so on by Product with the ability to further understand this distribution across key customer segments like Gender, Age, and Income.

Figure 88: Expense Analysis
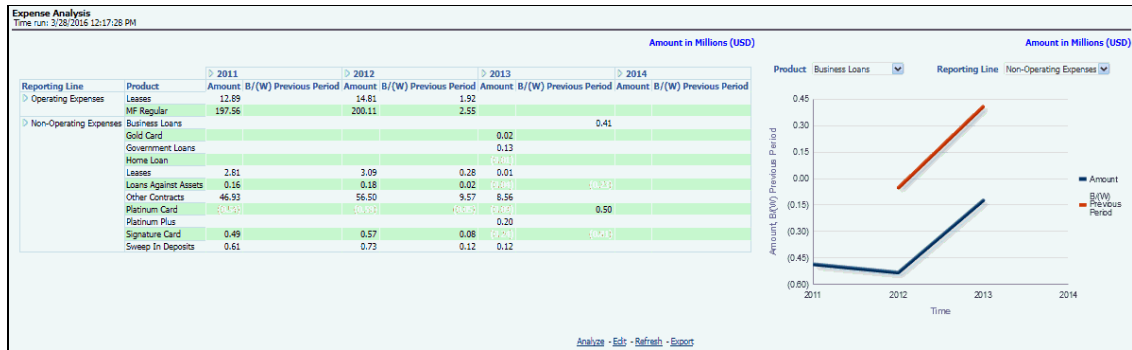
16.1.2　Credit Loss Summary

16.1.2.1　Risk Summary

This report gives a snapshot of the number of accounts in each delinquency bucket and the corresponding balance in each of these buckets. This can further be refined to limit this data to a specific LOB or a product within a LOB.

Figure 89: Risk Summary
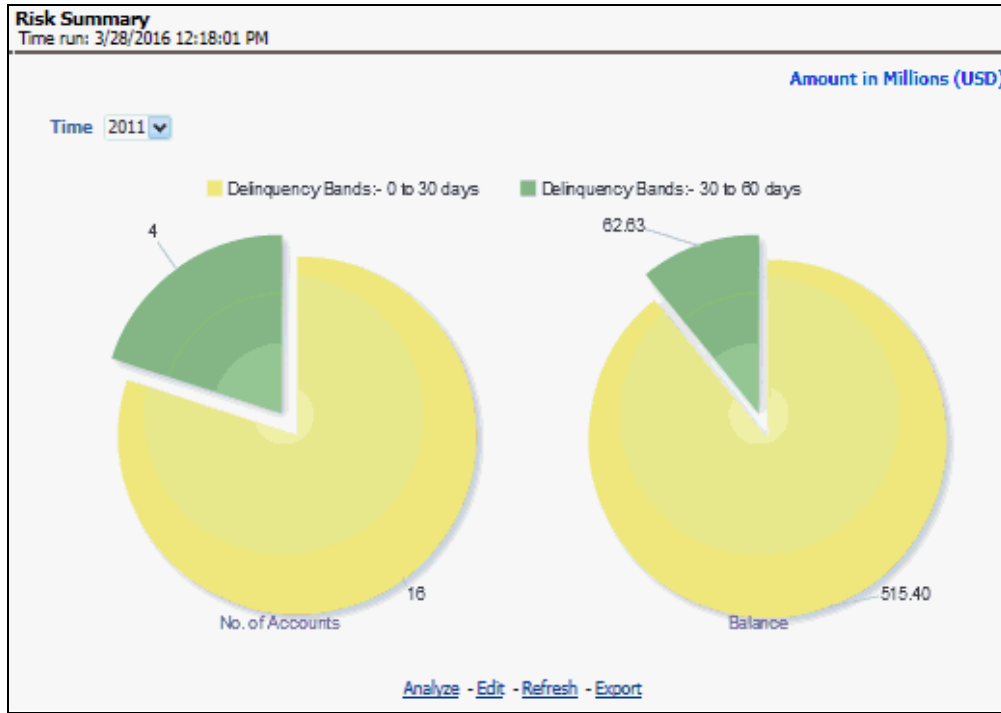
16.1.2.1.1 Credit Loss Statement

This report displays Net Credit Loss (Gross Credit Loss adjusted for any recoveries) booked by the bank.

Figure 90: Credit Loss Statement
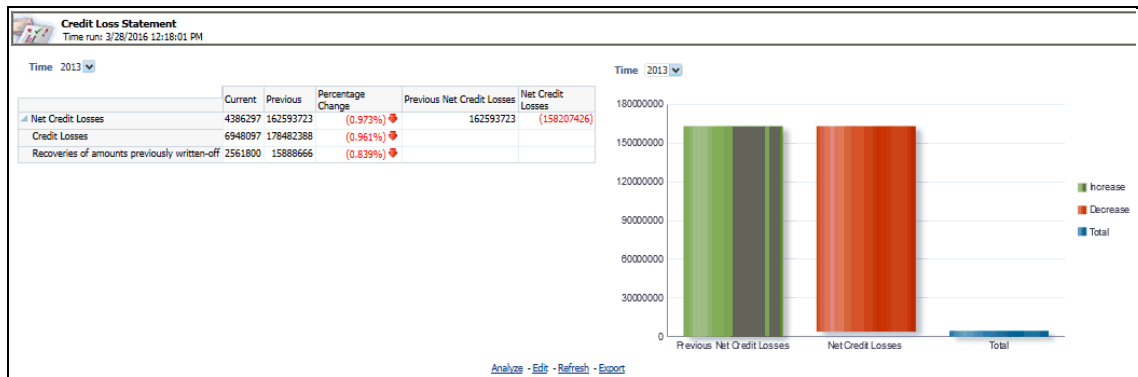
16.1.3 Margin Report

This report tracks the margin of profitability that has been achieved by the customer over a period.

Figure 91: Margin Report
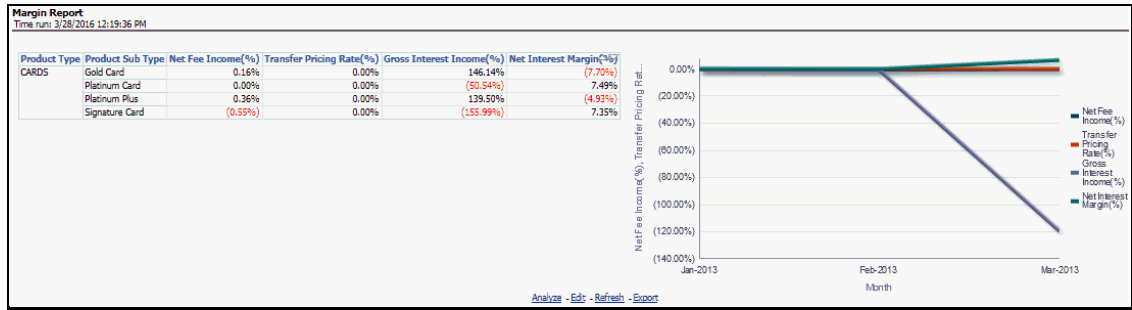
16.1.4 Customer Summary

16.1.4.1 Customer Distribution by Income

This report displays the Distribution of Open Customers and Open Accounts across Income bands.

Figure 92: Customer Distribution by Income
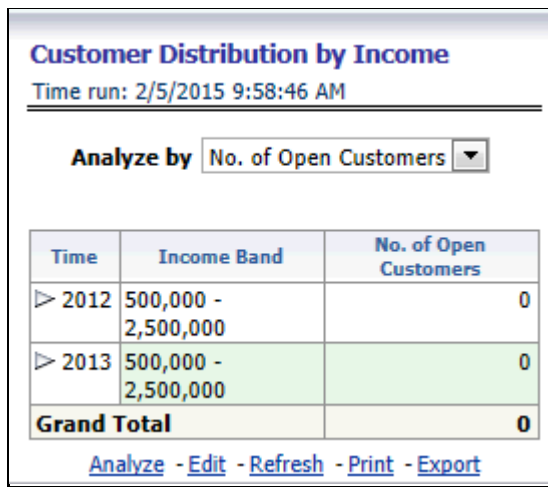
16.1.4.2 Customer Distribution by Age

This report displays the Distribution of Open Customers and Open Accounts across Age bands.

Figure 93: Customer Distribution by Age
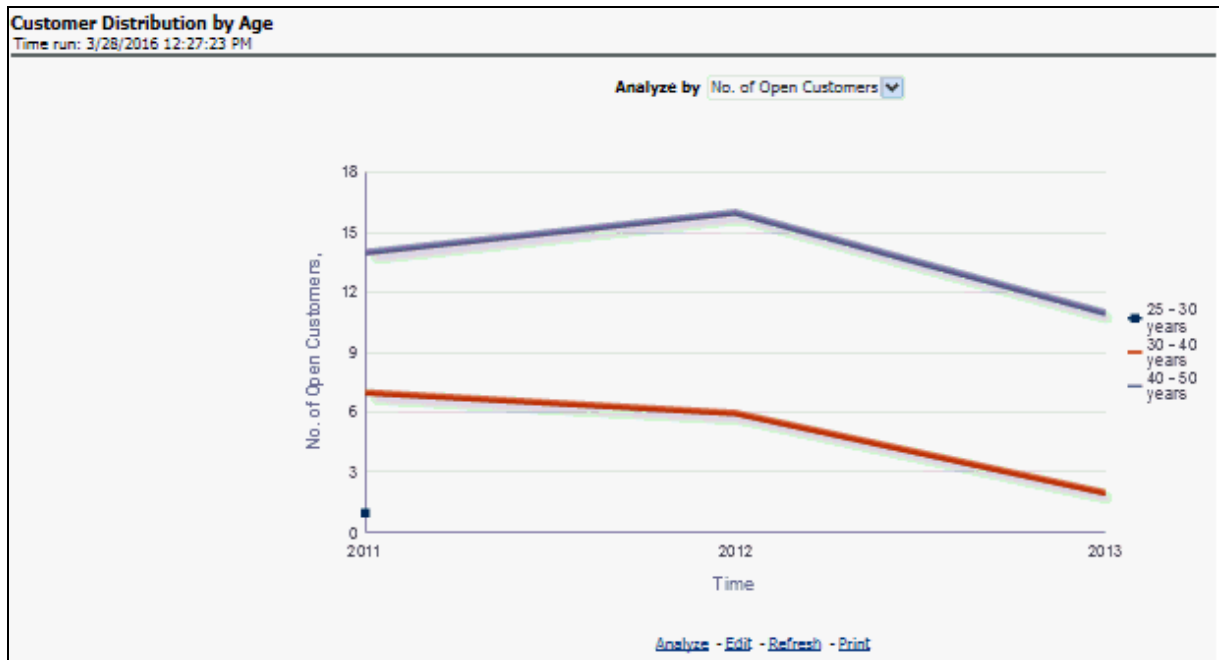
16.1.4.3 Customer Distribution by LoB

This report displays the distribution of Open Customers and the corresponding Revenue across each Line of Business, its constituent products, and year. This is a sunburst report.

On clicking a particular distribution, a consolidated view of that particular distribution is displayed. For example, if you need to get a consolidated view of 2011 distribution, click on 2011. You get consolidated data of 2011 distribution on open customers, Line of Business, constituent products, and corresponding revenue. Further, you can get a consolidated view of open customers, Line of Business, constituent products, and corresponding revenue, by clicking any particular distribution. To view the overall report, click the center of the circular graph.

Figure 94: Customer Distribution by LOB

Customer Distribution by LOB
Time run: 3/28/2016 12:27:23 PM

Analyze - Edit - Refresh - Print
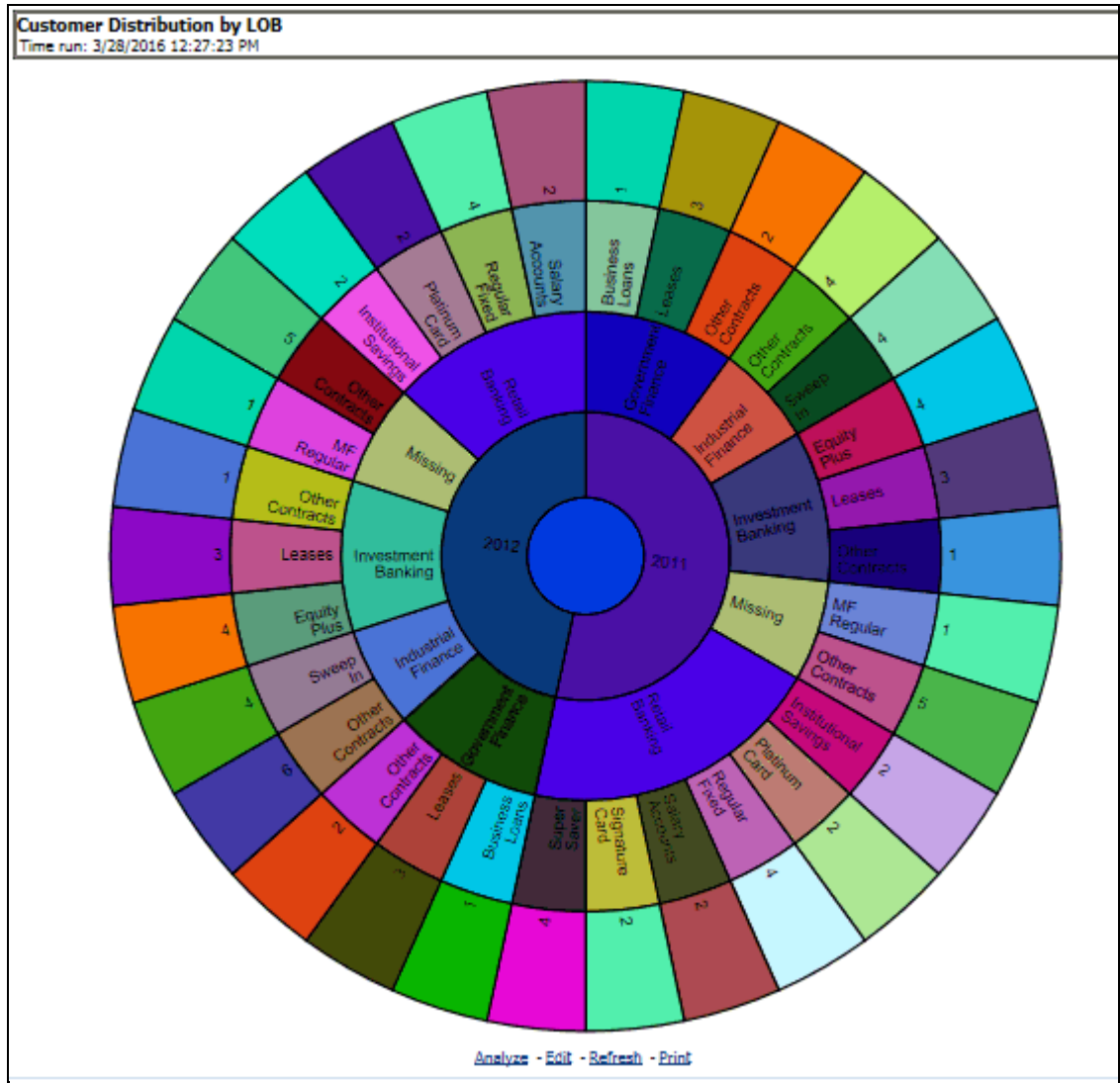
16.1.4.4   Customer Distribution by Region

This report displays the Top 10 Products by Customers and Revenue Distribution of Open Customers and Open Accounts across Regions and Products.

Figure 95: Customer Distribution by Region
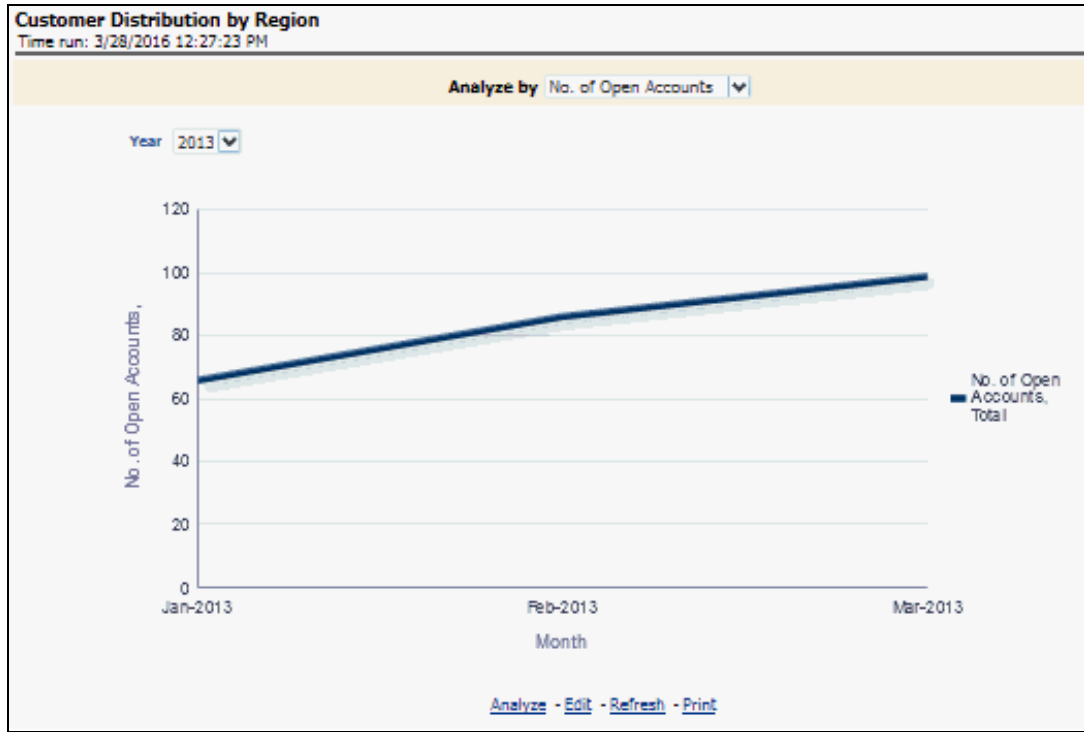
16.1.4.5   Product Penetration Report

This report displays the movement/sales of products over a period.

Figure 96: Product Penetration Report
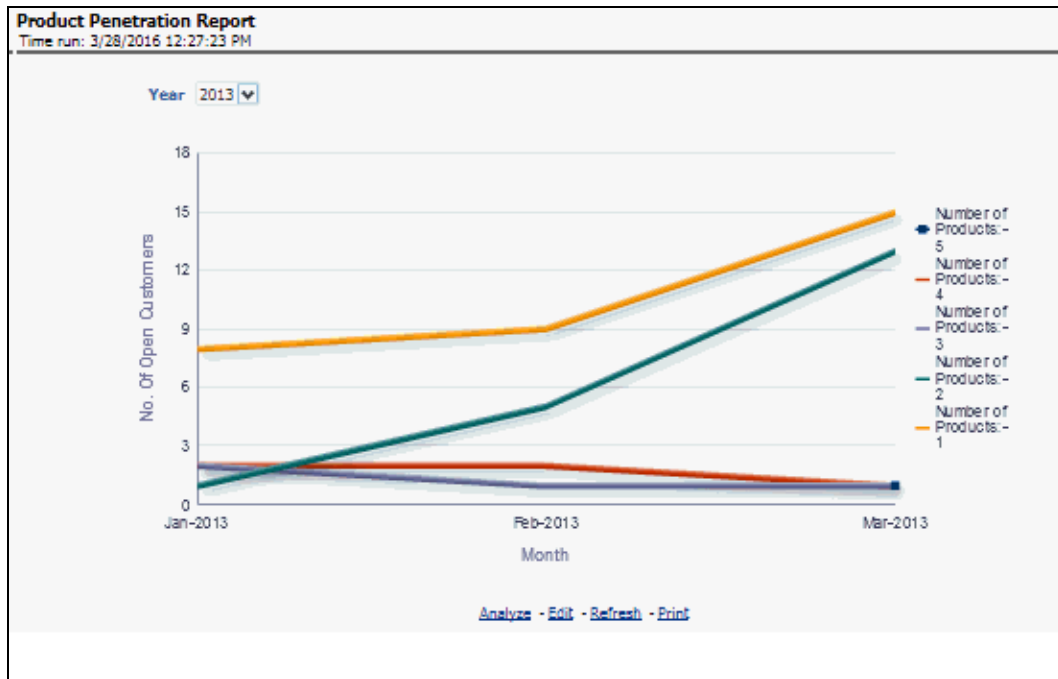
16.1.5 Customer Central

The purpose of this Dashboard is to provide detailed information about the customer, information related to the accounts of the customer, and other behavioral attributes. It enables the user to analyze a customer in its entirety. The report is specific to a customer and the selection of customer for which the report is to be viewed is done through the dashboard prompt. The search is enabled either by Customer or Account.

This Dashboard provides complete demographic details of the customer as well as the engagements of the customer with the bank. The engagement with the bank is specified in terms of the accounts held by the customer as well as the other services/activities through which the customer interacts with the bank.

All accounts of the customer (current as well as previous) are reported along with their specifics such as the start date, balance, peak balances, net income, relationship manager, and so on. Other reports include the specifics of the subscriptions and enrollments of the customer and the various offers that are provided to the customer and the accounts to which those offers have been provided. It also displays the details of transactions of the customer which can be viewed by classification into monetary or non-monetary transactions. Any predictive modeling scores that have been computed or are available for the customer are also reported.

Based on the profitability of the accounts, the future behavior of accounts is predicted and this predicted value is used to compute Customer Life-Time Value (CLTV). The CLTV can be analyzed for different periods of projections and accordingly the projected data to be considered for reporting CLTV is selected. Various reports available under this tab are discussed in the following sections:

The following Tabs and Reports are present in the Customer Central tab:

16.1.5.1 Customer 360

Based on the Segment filter prompt selected the reports are generated. The options available under the Segment filter prompt are as follows:

- Risk-Based
- Demographics
- Behavioral
- Profitability

Figure 97: Customer 360 Filters

The maximum number of measures/dimensions that can be selected is 15. On selecting more than 15 measures/dimensions, the first selected dimensions in the hierarchy are deselected.

16.1.5.1.1 Retail Profile

This report provides the basic demographics of a particular customer.

Figure 98: Retail Profile

Customer Central (Sunburst-Wheel): This report displays a circular graphical representation that is divided into several sectors. Each sector represents the value of the dimension or measure, that is, Turnover, Customer Since, Total Asset Balance, Total Liability Balance, No. of Asset Product, No. of Liability Product, No. of Products Held (currently), No. of Products Held (Since the inception of the customer), Debit Turnout, Mitigant Value, Total Spent, and so on, of the customer that has been selected. This is a sunburst report. On clicking a particular segment, the selected segment rotates and appears on the top part of the circle for better visualization.

The radial axis on the anti-clock side of every sector represents the scale for that sector. The following values are represented in each sector:

- **Customer Value**: This represents the dimensional value of customers across the scale.

- **Segment Average**: This represents the average value of the dimension of the segment that the customer belongs to.

- **Enterprise Average**: This represents customers from all the segments considered to compute the average value of dimension or measure.

Hyperlinks are provided under the Customer, Scores, and Rating section. On clicking these hyperlinks, you are navigated to the respective detail reports.

**Figure 99: Customer Central**
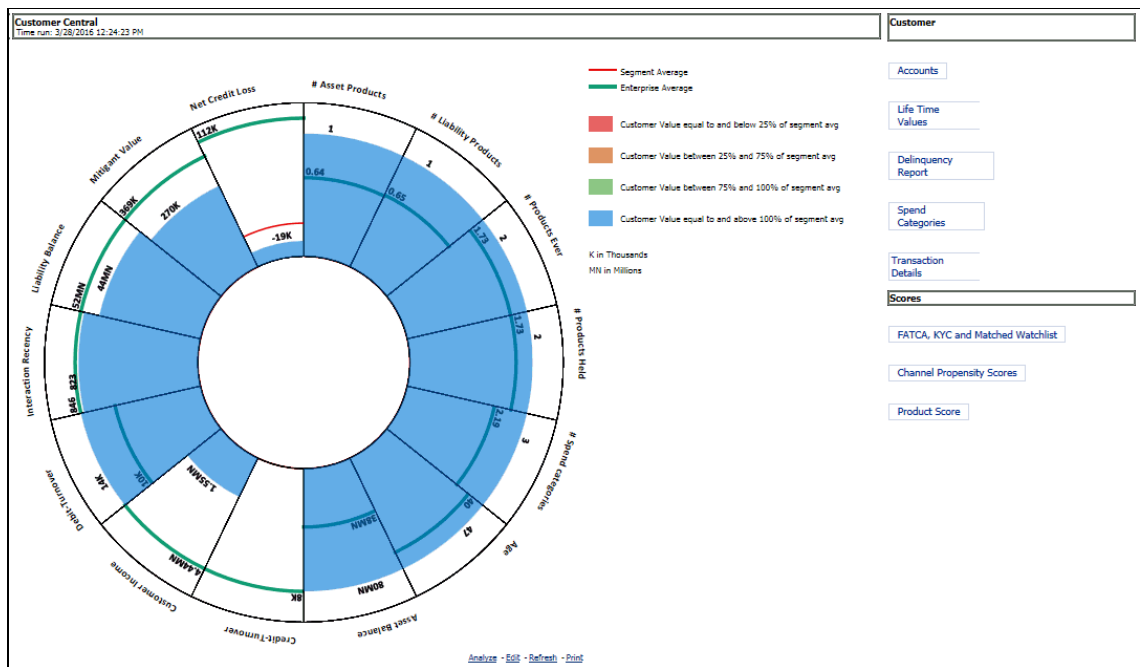
16.1.5.2   Customer Performance

16.1.5.2.1   Risk-Adjusted Performance Metrics

This report helps you to determine the ratio of risk-adjusted Net Income against the Economic Capital.

This metric is also called Risk-Adjusted Return On Capital (RAROC). It helps in determining the efficiency of Economic Capital corresponding to every customer. This Report shows a snapshot of measures against various reporting lines, for example, Total Revenue, Total Expenses, Net Income, Return on Total Asset RAROC-Economic Capital, and Return on Equity.

Figure 100: Risk-Adjusted Performance Metrics
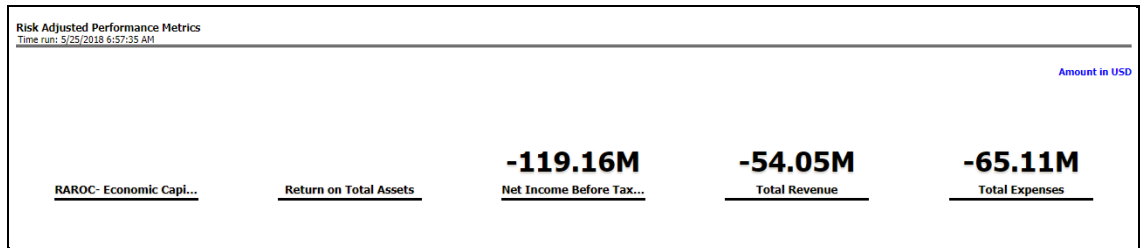
16.1.5.2.2   Income Statement

This report displays Revenues and Expenses under various reporting lines. The numbers are displayed for the current period selected and in comparison with numbers recorded during the previous same period and move in terms of the percentage change.

Figure 101: Income Statement
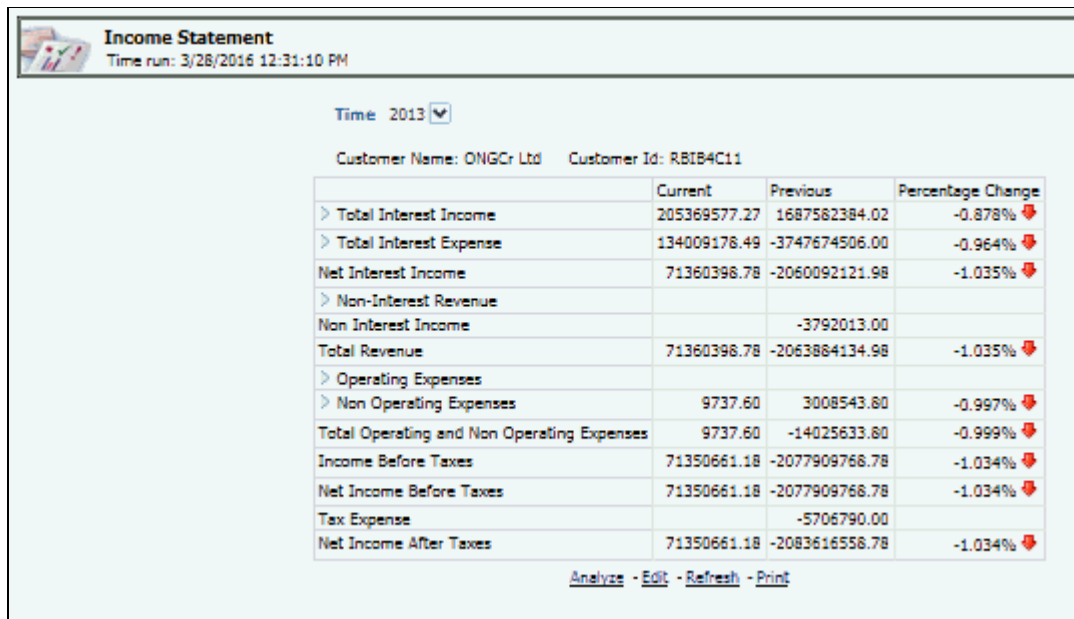
16.1.5.2.3   Profit and Loss Summary

This report displays the Profit and Loss statement for all products which the user can then limit to a specific Line of Business (LOB) or slices of the P&L for a specific LOB across individual customer segments like age, gender, and Income.

Figure 102: Profit and Loss Summary
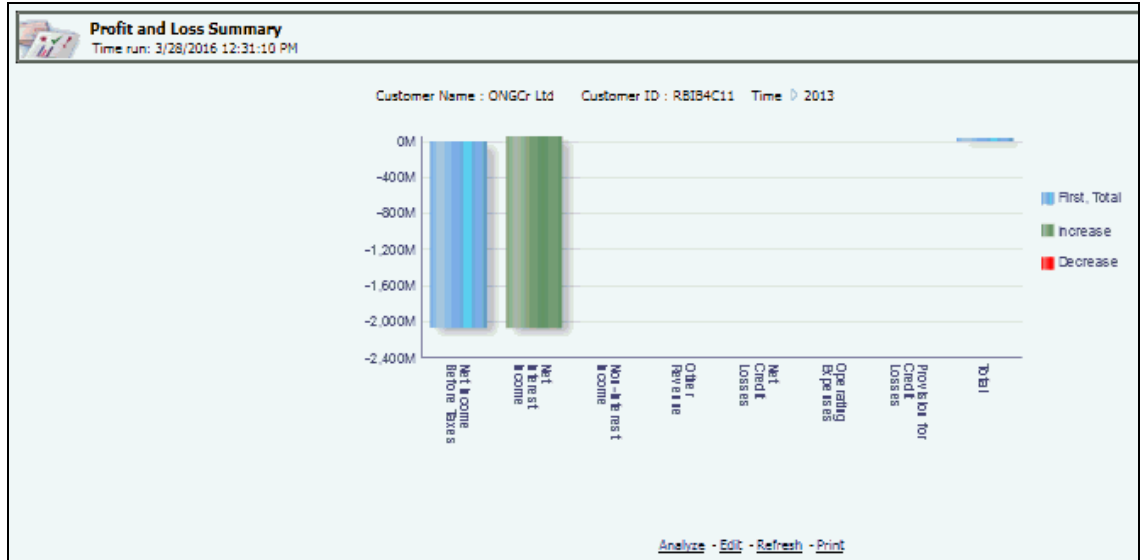
16.1.6   Product Summary

16.1.6.1   All Product

16.1.6.1.1   Risk-Adjusted Performance Metrics

This report helps you to determine the ratio of risk-adjusted Net Income against the Economic Capital.

This metric is also called Risk-Adjusted Return On Capital (RAROC). It helps in determining the efficiency of Economic Capital corresponding to every customer. This Report shows a snapshot of measures against various reporting lines, for example, Total Revenue, Total Expenses, Net Income, Return on Total Asset RAROC-Economic Capital, and Return on Equity.

Figure 103: Risk-Adjusted Performance Metrics
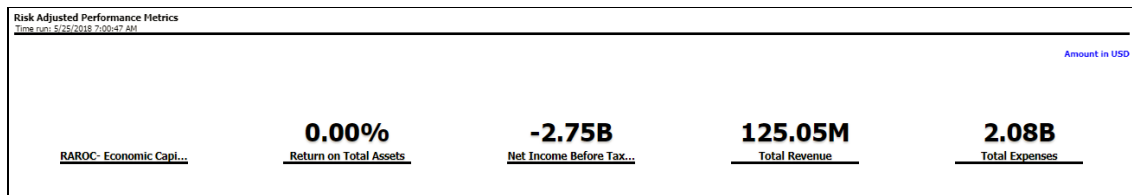
16.1.6.1.2   Income Statement

This report displays Revenues and Expenses under various reporting lines. The numbers are displayed for the current period selected and in comparison with numbers recorded during a previous same period and move in terms of the percentage change.

Figure 104: Income Statement

16.1.6.1.3 Profit and Loss Summary

This report displays the Profit and Loss statement for all products which the user can then limit to a specific Line of Business (LOB) or slices of the Profit and Loss for a specific LOB across individual customer segments like age, gender, and Income.

Figure 105: Profit and Loss Summary

16.1.6.1.4 Profit and Loss - Scenario Comparison

This report provides a comparison of the details of the income generated to predefined scenarios.

Figure 106: Profit and Loss Scenario Comparison

Some of the dashboard filters, that is, Vintage, The Age On Book, Customer Gender, Customer Age/Customer Income, Geography Filter prompt selection do not apply to this report.

16.1.6.2 Cards

16.1.6.2.1 Risk-Adjusted Performance Metrics

This report helps you to determine the ratio of risk-adjusted Net Income against the Economic Capital.

This metric is also called Risk-Adjusted Return On Capital (RAROC). It helps in determining the efficiency of Economic Capital corresponding to every customer. This Report shows a snapshot of measures against various reporting lines for example, Total Revenue, Total Expenses, Net Income, Return on Total Asset RAROC-Economic Capital, and Return on Equity.

Figure 107: Risk-Adjusted Performance Metrics
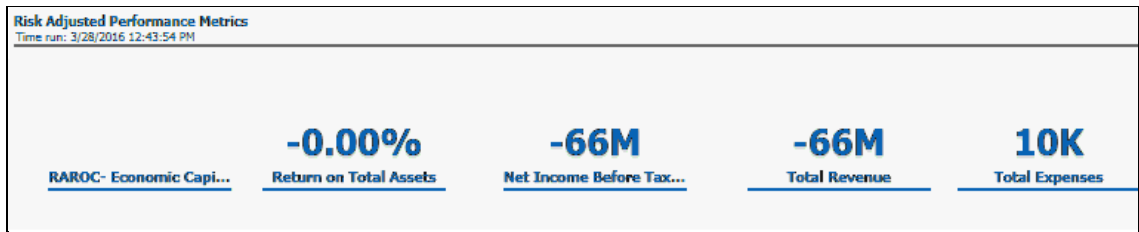
16.1.6.2.2 Income Statement

This report displays Revenues and Expenses under various reporting lines. The numbers are displayed for the current period selected and in comparison with numbers recorded during the previous same period and move in terms of the percentage change.

Figure 108: Income Statement

Income Statement
Time run: 3/28/2016 12:43:54 PM

	Current	Previous	Percentage Change
> Total Interest Income	1740190	24004812	(0.928%)
> Total Interest Expense	313200	(6206826)	(0.950%)
Net Interest Income	1426990	17797986	(0.920%)
> Non-Interest Revenue	1423431	46088200	(0.969%)
Non Interest Income	1423431	46088200	(0.969%)
Total Revenue	2850421	63886185	(0.955%)
> Non Operating Expenses	(28244)	(154259)	(0.817%)
Total Operating and Non Operating Expenses	(28244)	154259	(0.817%)
Income Before Taxes	2878665	64040444	(0.955%)
Net Income Before Taxes	2878665	64040444	(0.955%)
Net Income After Taxes	2878665	64040444	(0.955%)

Analyze - Edit - Refresh - Export

16.1.6.2.3   Profit and Loss Summary

This report displays the Profit and Loss statement for Cards products that can be viewed as slices of individual customer segments like age, gender, and Income.

Figure 109: Profit and Loss Summary
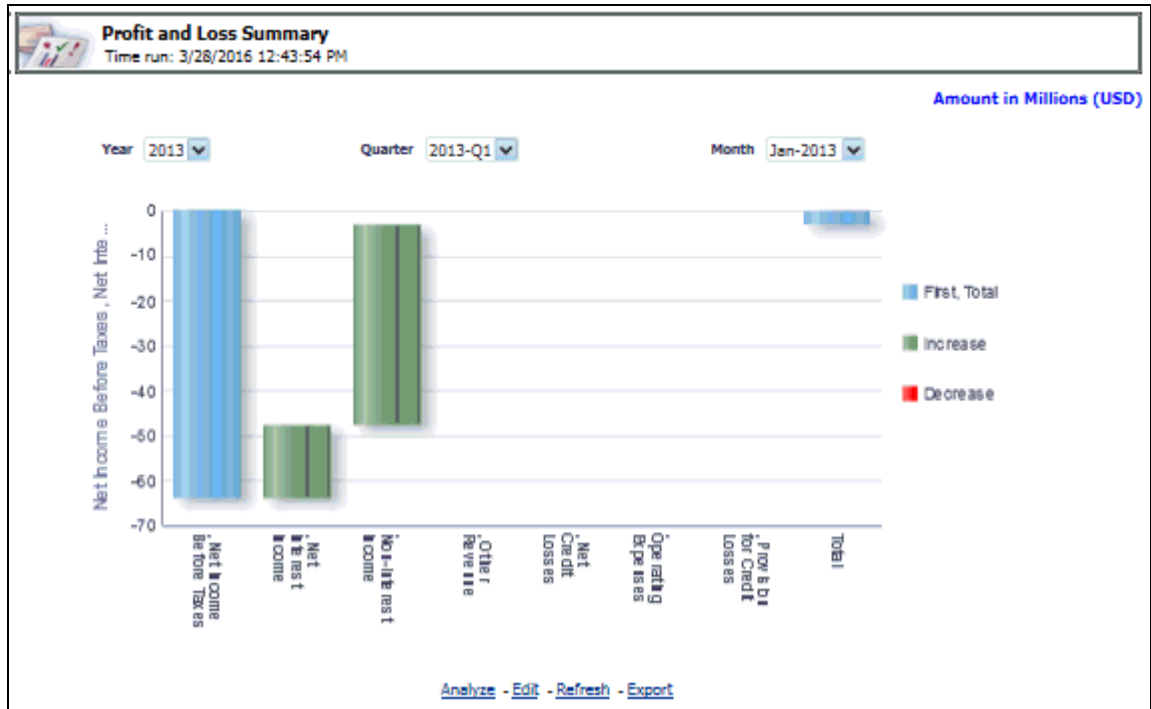
16.1.6.2.4   Performance by Card Type

This report displays Key Metrics such as the Number of New accounts, Number of Open customers, and so on reported across various card types demonstrating the mix across the card types.

Figure 110: Performance by Card Type

Performance by Card Type
Time run: 3/28/2016 12:43:54 PM

Amount in Millions (USD)

| Time	Card Type	No. of Accounts	No. of New Accounts	No. of Open Customers	Fee Revenue	Total Payments	Total Revenue
2011	Platinum Card	2	1	2	14.15	0.13	142.42
	Signature Card	2	2	2	14.45	0.16	136.89
2012	Platinum Card	2	0	2	12.09	0.13	148.79
	Signature Card	2	0	2	15.43	0.16	127.22
2013	Gold Card	9	4	2	4.33	0.05	53.67
	Platinum Card	13	6	5	8.62	0.07	77.43
	Platinum Plus	12	7	6	6.12	0.09	92.60
	Signature Card	9	5	4	6.03	0.07	55.35

Analyze - Edit - Refresh - Export

16.1.6.2.5   Standard Non-Cash Balance

This report displays the growth (or decline) of customer non-cash balance over time.

Figure 111: Standard Non-Cash Balance
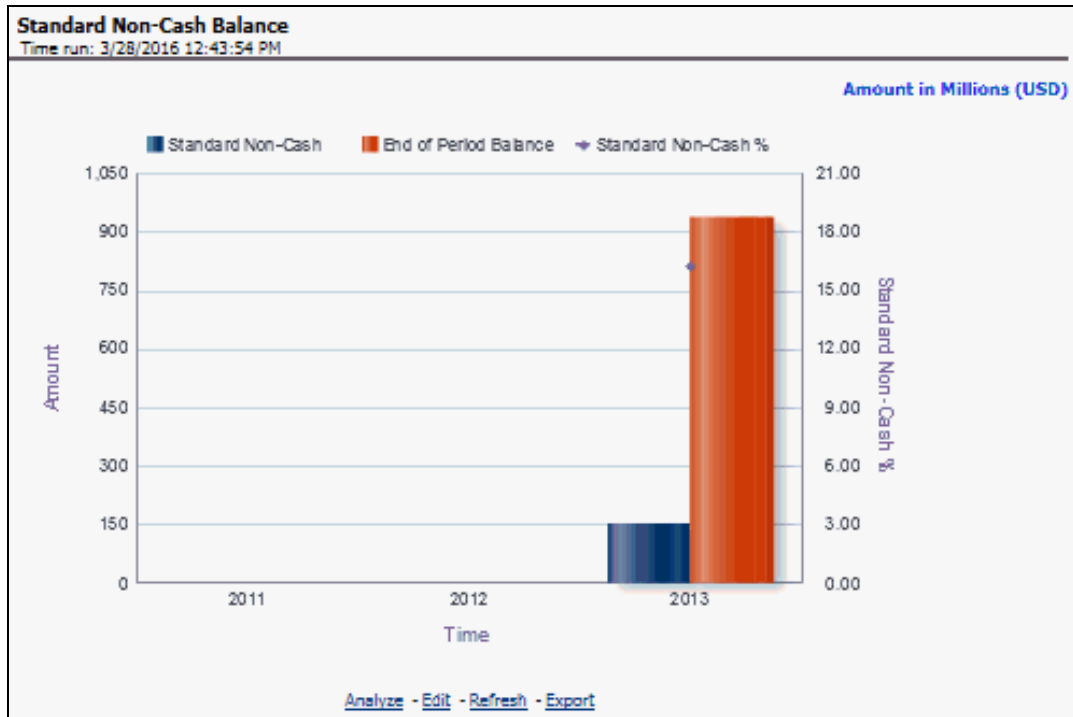
16.1.6.2.6   Balance by Expiry Month

This report displays balance distribution across offer expiry dates gives the user an indication of the balance that is scheduled to flip to a different balance bucket.

Figure 112: Balance by Expiry Month
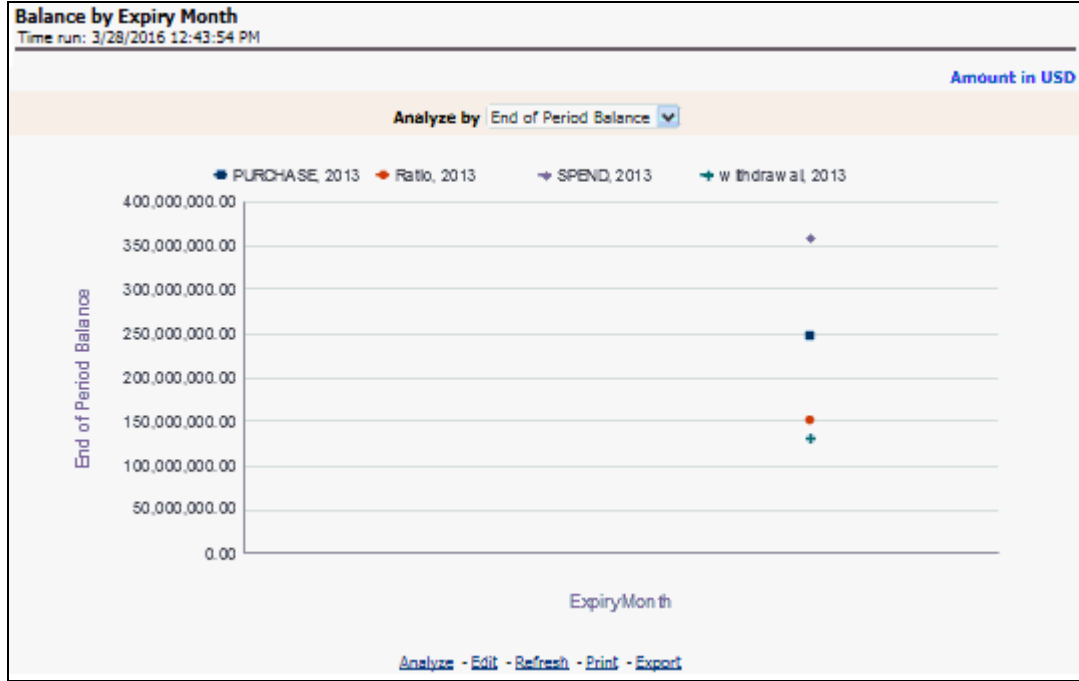
16.1.6.2.7   Active Net Receivables Summary

This report displays the average growth (or decline) of balances per open customer in relation to how the standard non-cash balance is growing (or declining).

Figure 113: Active Net Receivables Summary
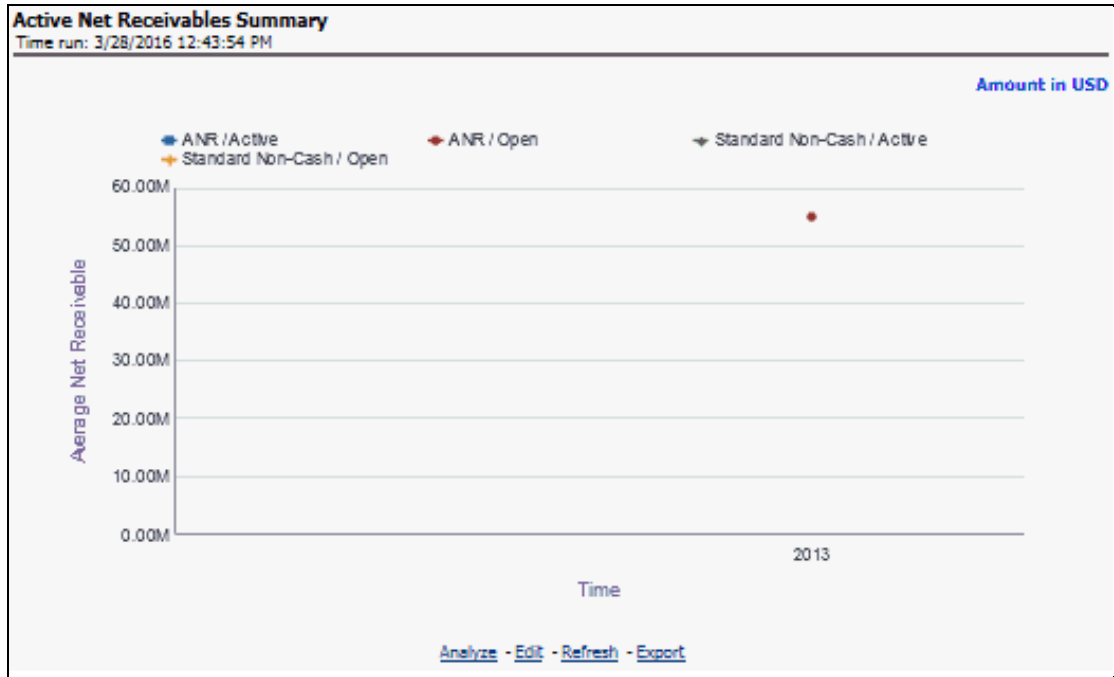
16.1.6.2.8 Summary of Balance (Receivables) Breakdown

This report summarizes the Average Net Receivable (ANR) mix across the different balance buckets available within the bank and the effective interest rate across each balance bucket.

Figure 114: Summary of Balance (Receivables) Breakdown
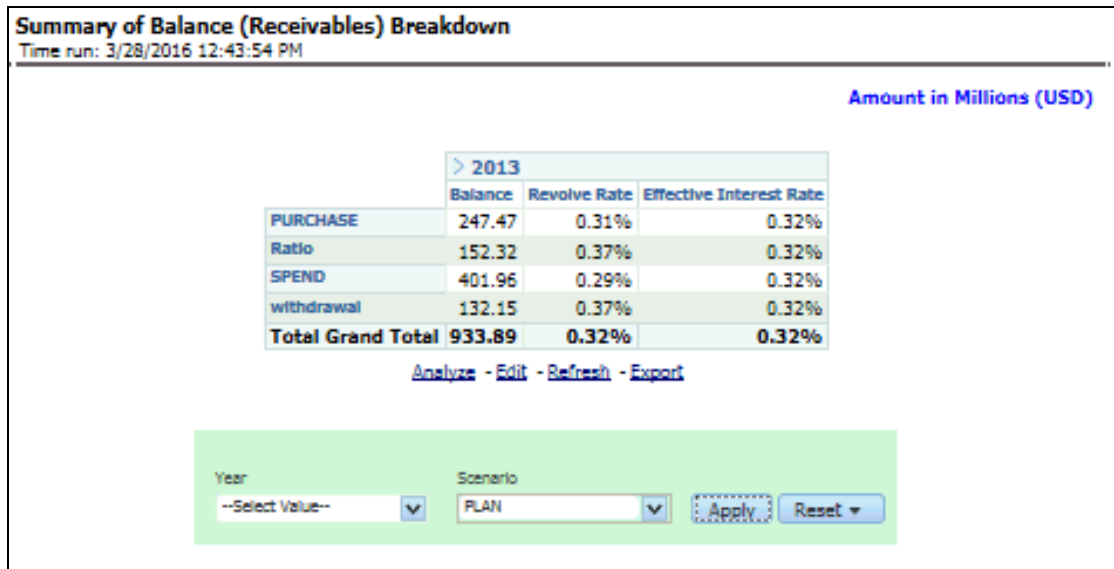
16.1.6.2.9 Profit and Loss - Scenario Comparison

This report provides a comparison of the details of the income generated to predefined scenarios.

Figure 115: Profit and Loss - Scenario Comparison

Some of the dashboard filters, that is, Vintage, The Age On Book, Customer Gender, Customer Age/Customer Income, Geography Filter prompt selection do not apply to this report.

16.1.6.3   Retail Bank

16.1.6.3.1   Risk-Adjusted Performance Metrics

This report helps you to determine the ratio of risk-adjusted Net Income against the Economic Capital.

This metric is also called Risk-Adjusted Return On Capital (RAROC). It helps in determining the efficiency of Economic Capital corresponding to every customer. This Report shows a snapshot of measures against various reporting lines eg; Total Revenue, Total Expenses, Net Income, Return on Total Asset RAROC-Economic Capital, and Return on Equity.

Figure 116: Risk-Adjusted Performance Metrics

16.1.6.3.2   Income Statement

This report displays Revenues and Expenses under various reporting lines. The numbers are displayed for the current period selected and in comparison with numbers recorded during the previous same period and move in terms f percentage change.

Figure 117: Income Statement
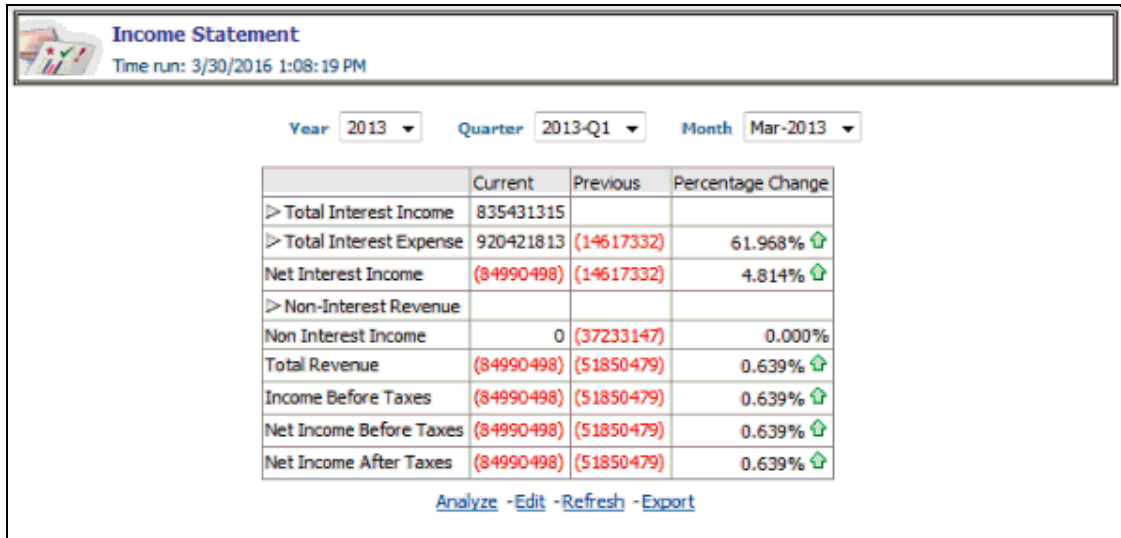
16.1.6.3.3 Profit and Loss Summary

This report displays the Profit and Loss statement for Cards products that can be viewed as slices of individual customer segments like age, gender, and Income.

Figure 118: Profit and Loss Summary
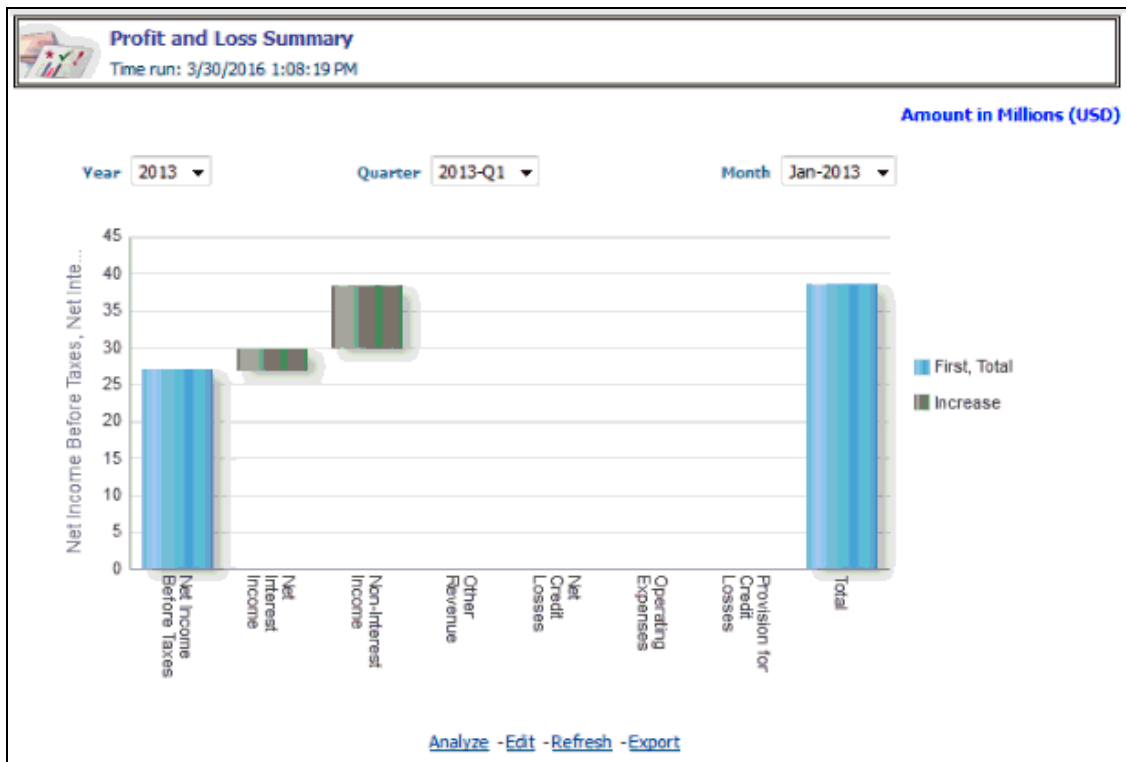
16.1.6.3.4 Total Deposit Analysis

This report displays the distribution of deposits booked by the bank across various product categories.

Figure 119: Total Deposit Analysis
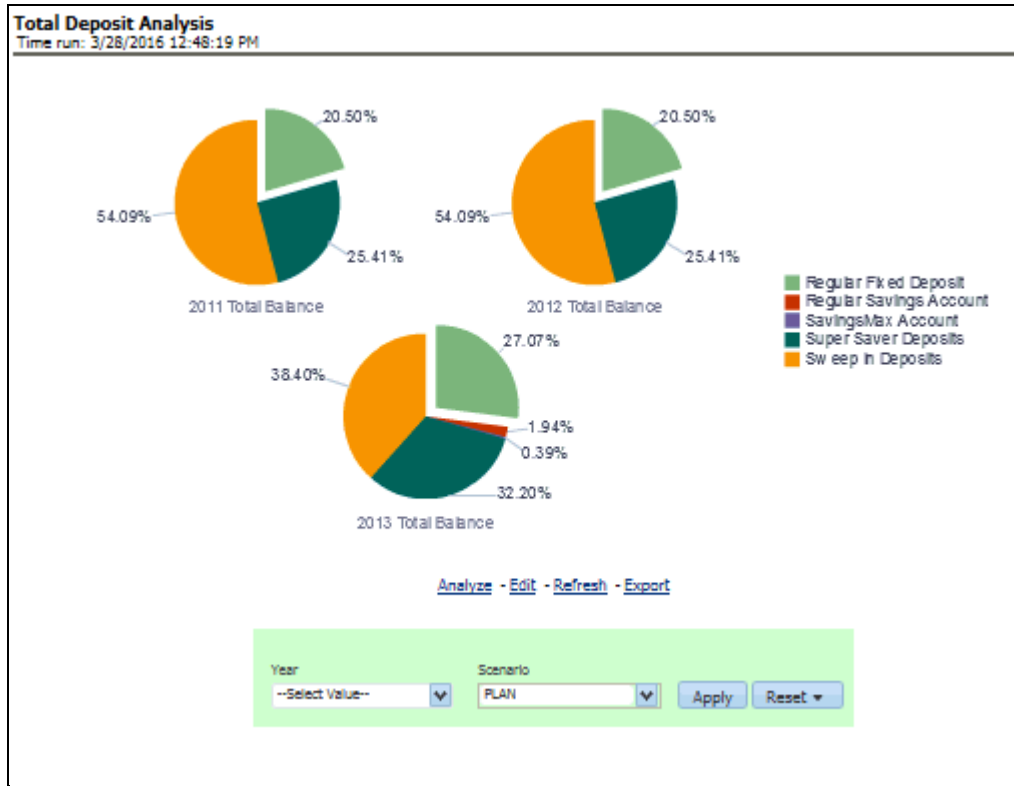
16.1.6.3.5   Profit and Loss - Scenario Comparison

This report provides a comparison of the details of the income generated to predefined scenarios.

Figure 120: Profit and Loss - Scenario Comparison

Some of the dashboard filters, that is, Vintage, The Age On Book, Customer Gender, Customer Age/Customer Income, Geography Filter prompt selection do not apply to this report.

16.1.6.4   Mortgage

16.1.6.4.1   Risk-Adjusted Performance Metrics

Figure 121: Risk-Adjusted Performance Metrics

This report helps you to determine the ratio of risk-adjusted Net Income against the Economic Capital.

This metric is also called Risk-Adjusted Return On Capital (RAROC). It helps in determining the efficiency of Economic Capital corresponding to every customer. This Report shows a snapshot of measures against various reporting lines for example, Total Revenue, Total Expenses, Net Income, Return on Total Asset RAROC-Economic Capital, and Return on Equity.

16.1.6.4.2   Income Statement

This report displays Revenues and Expenses under various reporting lines. The numbers are displayed for the current period selected and in comparison with numbers recorded during a previous same period and move in terms f percentage change.

Figure 122: Income Statement

16.1.6.4.3   Profit and Loss Summary

This report provides a comparison of the details of the income generated to predefined scenarios.

Figure 123: Profit and Loss Summary
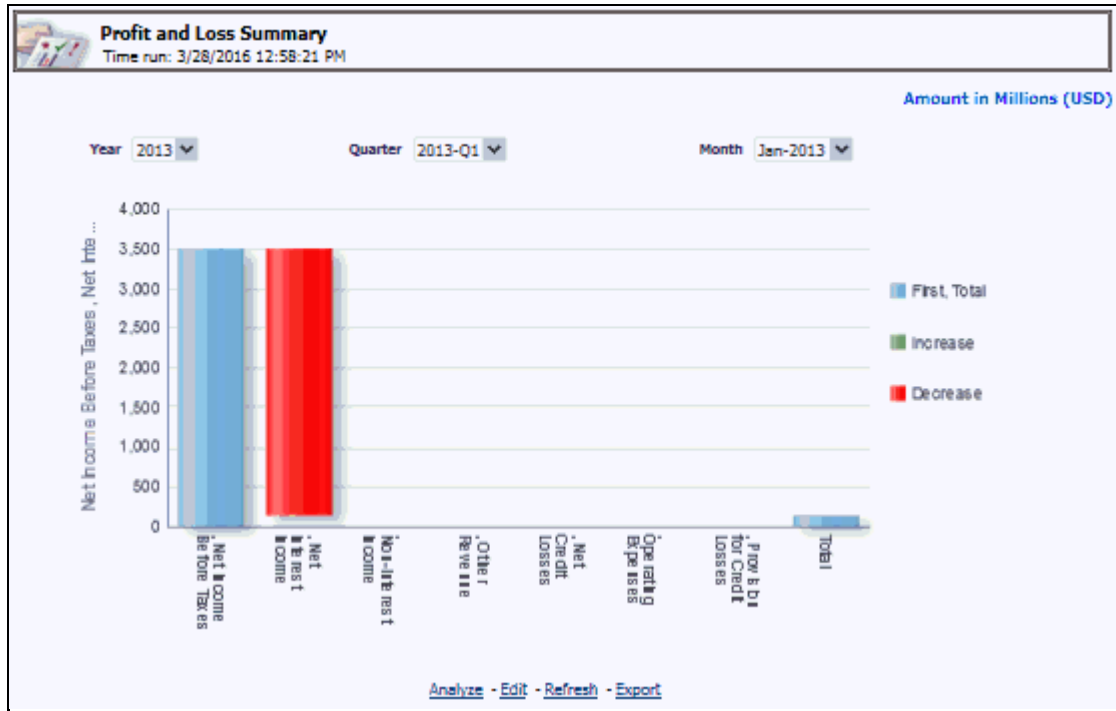
16.1.6.4.4 Profit and Loss - Scenario Comparison

This report provides a comparison of the details of the income generated to predefined scenarios.

Figure 124: Profit and Loss - Scenario Comparison

Some of the dashboard filters, that is, Customer Gender, Customer Age/Customer Income, Geography Filter prompt selection do not apply to this report.

16.1.7 Relationship Manager Insights

16.1.7.1 Relationship Manager Portfolio

This report displays the various assets of a Relationship Manager. This report provides an account-wise product portfolio with the Revenue of customers held by each RM.

Figure 125: Relationship Manager Portfolio

16.1.7.2 Relationship Manager Org Performance

This report provides details of each Relationship Manager's performance at an Account Manager's level with details of customers, holding, total revenue, percent contribution to the business, and both direct and indirect revenue generation.

Figure 126: Relationship Manager Org Performance

17   What-If Analysis

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'. The 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 the 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 the account.

Users 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 a 'Scenario'. The projections are by default created for 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 a 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 the 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 that 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 get 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 the profitability of any probable change in the future. Certain users should have the authority to save a scenario that 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 the income statement in the scenario.

Figure 127: Income Statement Variation

The following input parameters are applicable for OFS PA:

Table 43: Input Parameters for OFS Performance Analytics

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 a limitation on the graph. Default upper limit for the graph will be based on underlying data in the 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 the graph, it is not possible through the graph and hence, it should be applied through the 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. |
|---|---|

17.1.1   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 dimensions to display the **Scenario - Basic** screen.

Figure 128: Basic Scenario
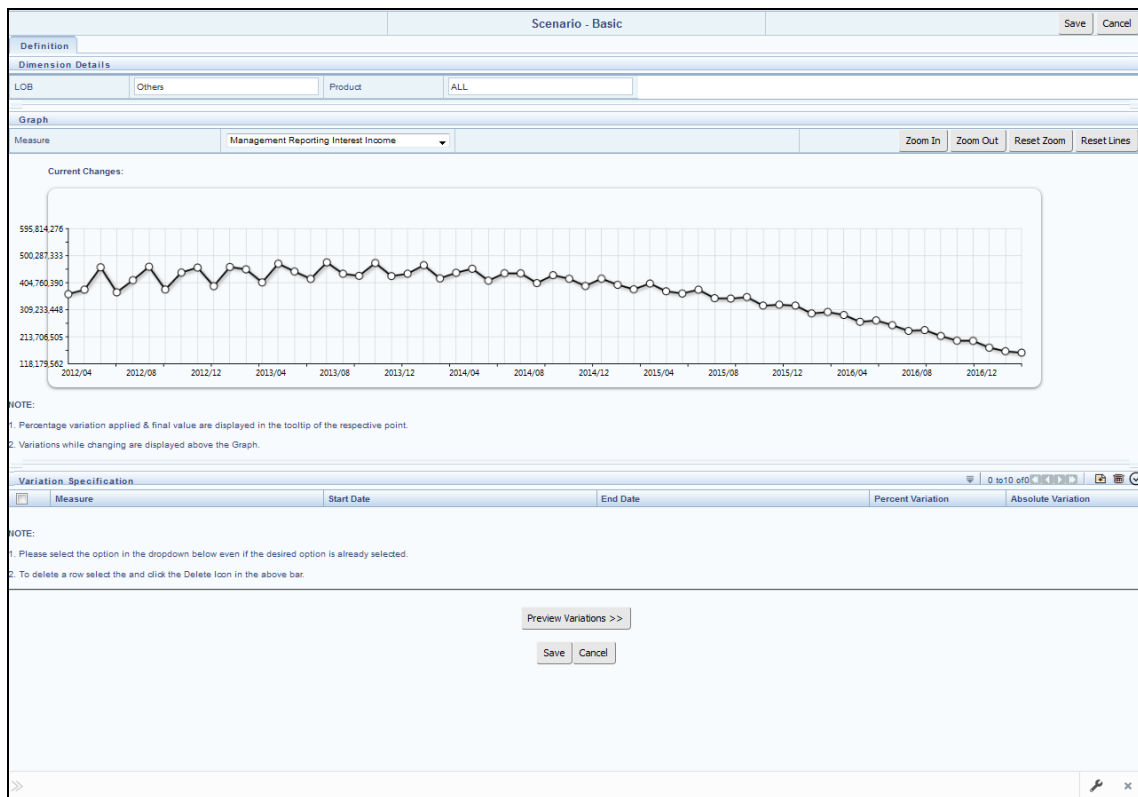
3. Apply the necessary **Dimension Details** for the following:

   - 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 the **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:

Figure 129: Business Summary
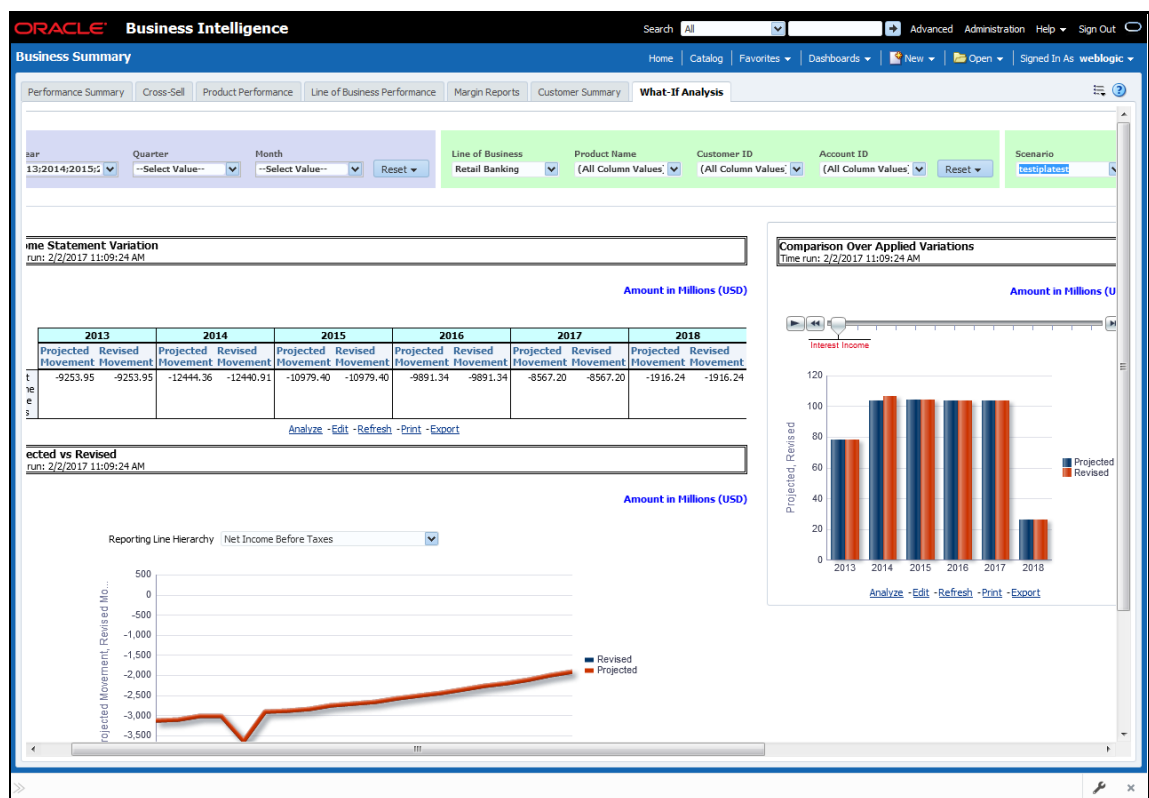

17.1.2 Advanced Scenario

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.

Figure 130: Advanced Scenario
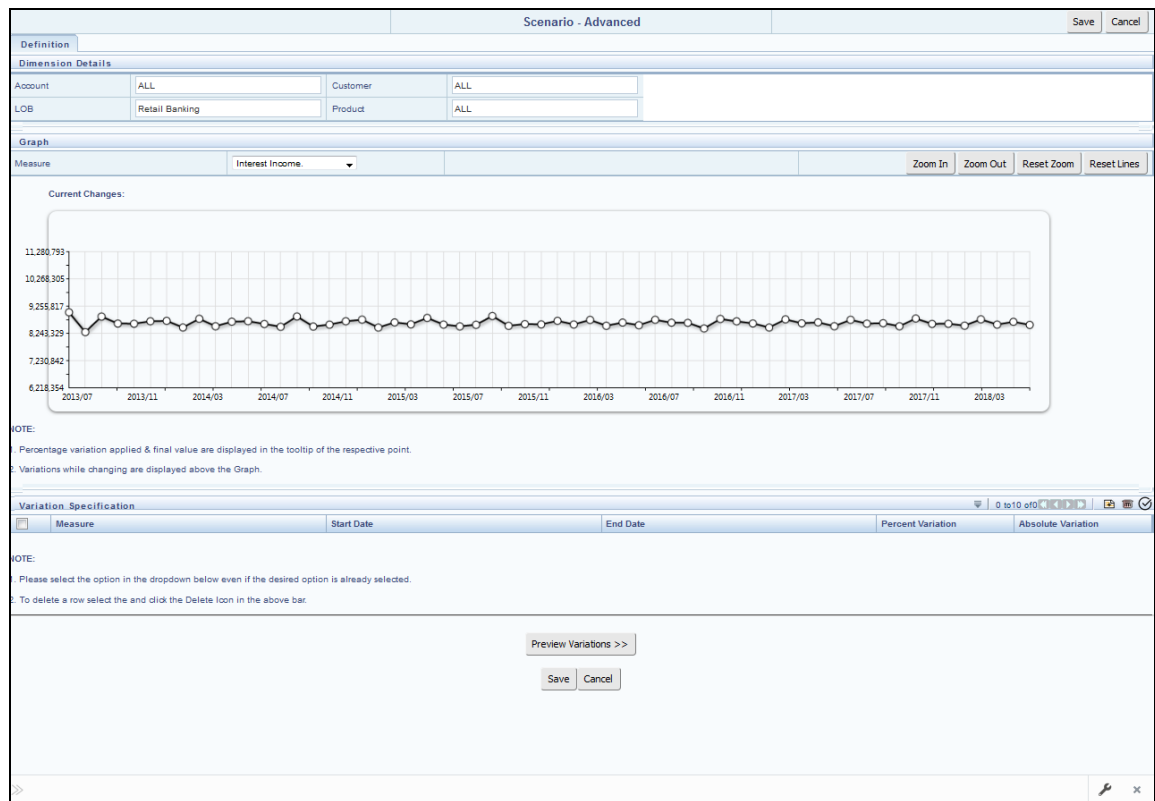
2. Apply the necessary **Dimension Details** for the following:

   ▪ 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:

Figure 131: Business Summary
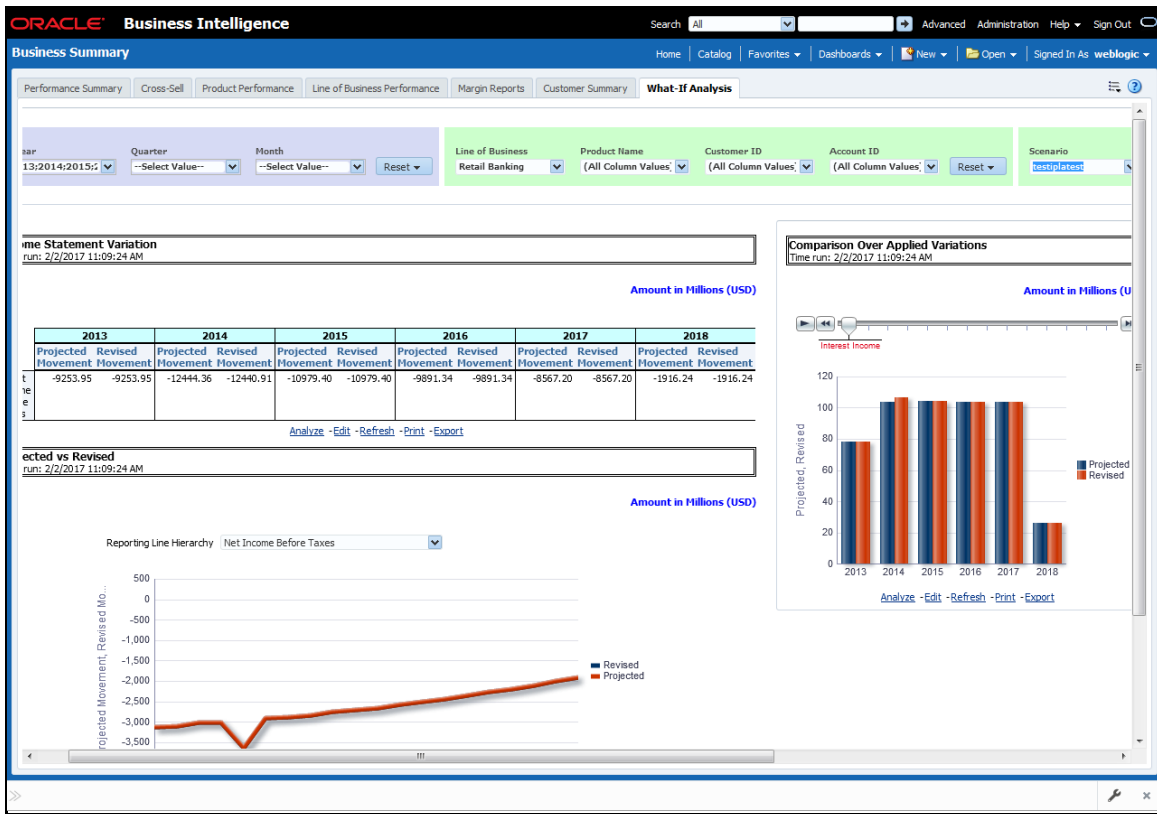
This chapter discusses the following topics:

- Configuration for What-If Analysis

- PACS Integration with Price Creation and Discovery (PCD)

- Reporting Line Correlation Calculations

- What-If Analysis Limitation

17.2 Configuration for What-If Analysis

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

Example: Assume that the OFSAA hostname is xx.xxx.xxx.xxxx and the OBIEE analytics port is xxxx. Hence, the

OBIEE analytics access URL would be http://xx.xx.xxx.xxx:xxxx/PFT810.

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 OFS IPA and Retail Performance for OFSRPA.

Figure 132: What-If Analysis Dashboard Page
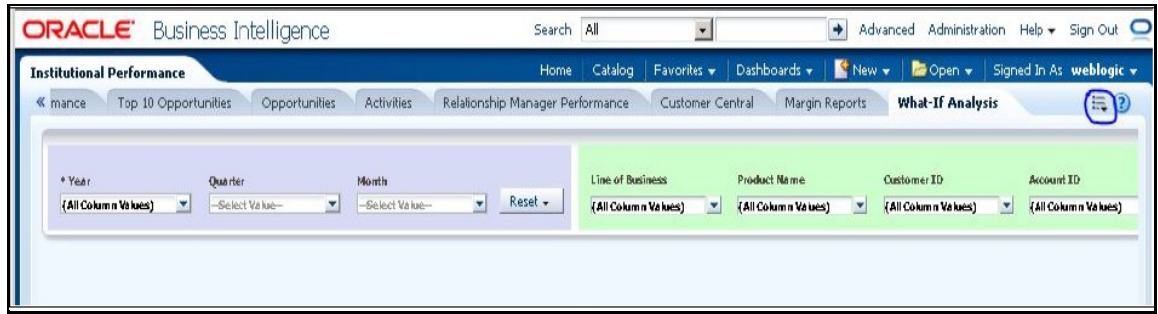
Edit the analysis Create Scenario.

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

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

Figure 133: Business Intelligence – Advanced XML Page

3. 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).

4. Click **Apply XML** and save the analysis after the occurrences of placeholders have been replaced and the XML contents have been pasted.

Figure 134: Business Intelligence – Advanced Tab
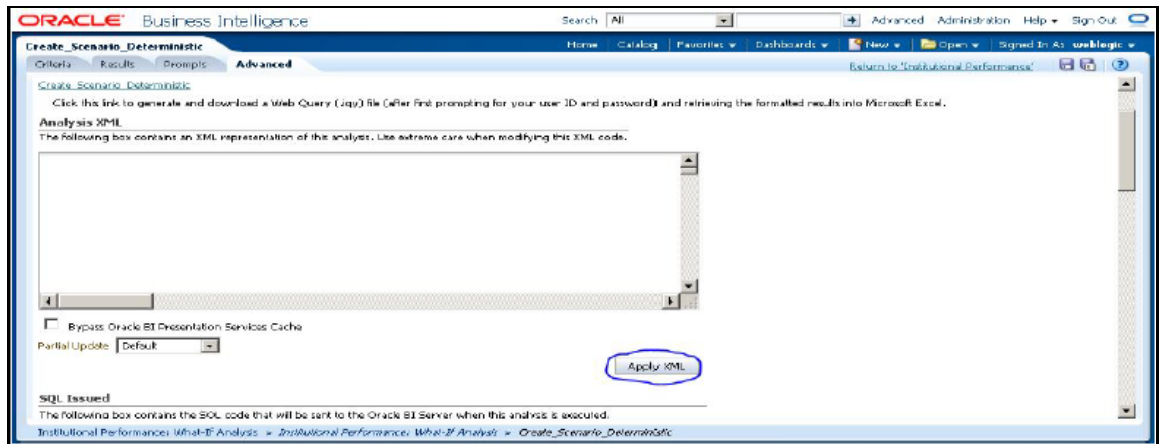
5. Configure the OBIEE URL in the What-If Model Definition setup tables to be able to navigate between the OFSAAI and OBIEE screens.

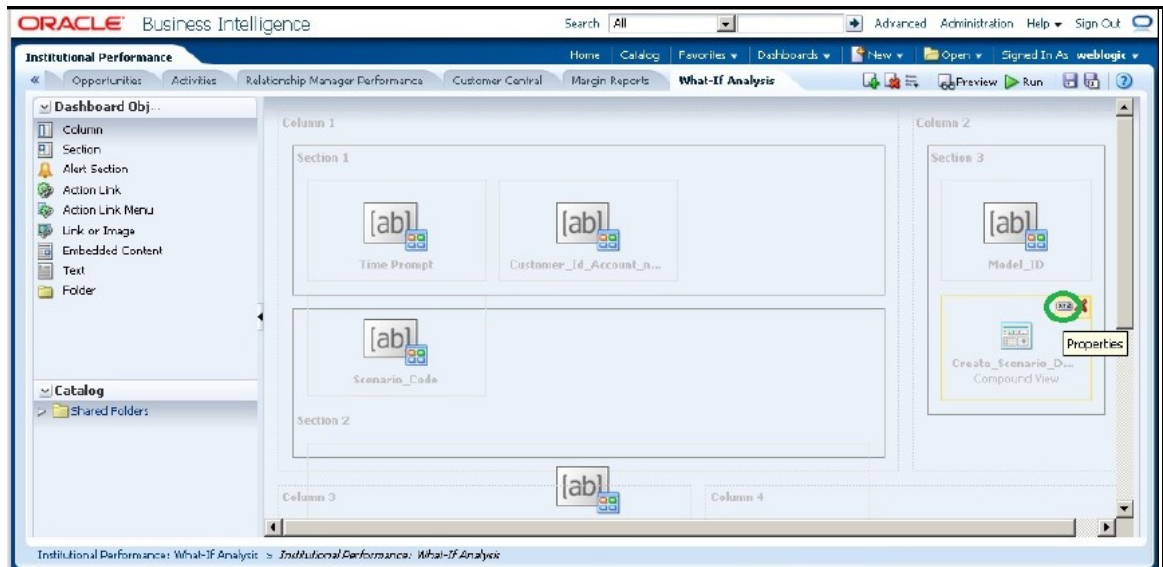   Example: Assume that the user hostname is – 10.184.150.107 and the OBIEE analytics port is 7001. Hence, the OBIEE analytics access URL would be:

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

6. 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.

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

Figure 135: Oracle BI Administration Tool – Variable Definition
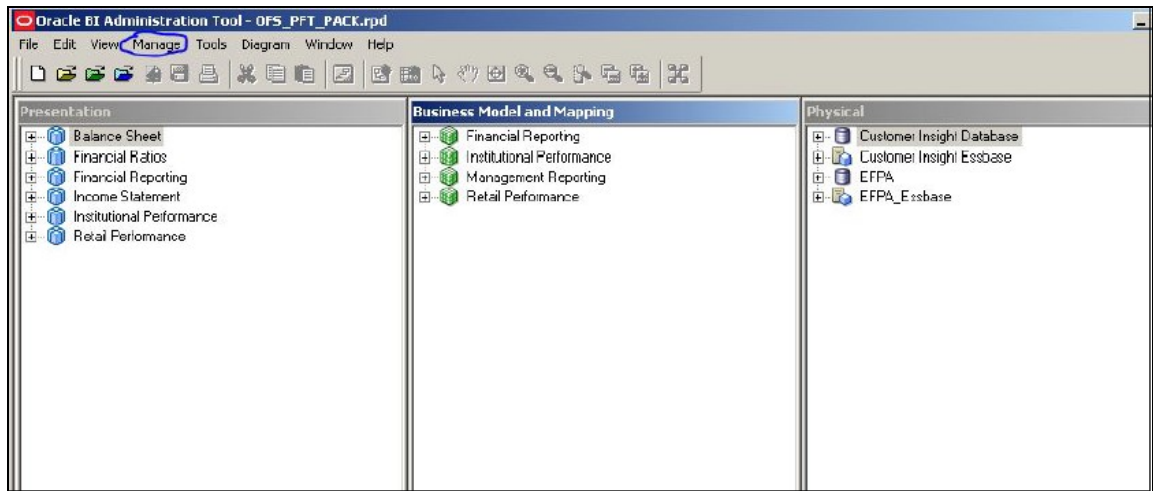
8. Modify the WIF_ADMIN_CI Repository Dynamic Variable:
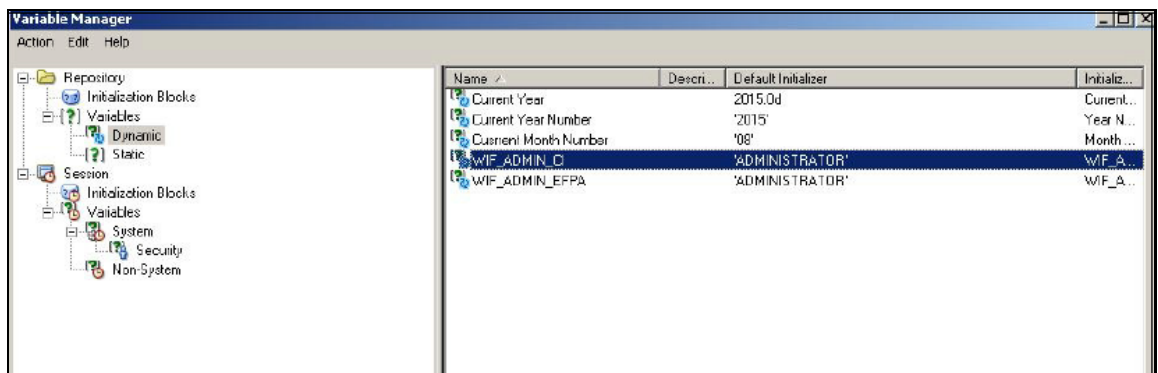
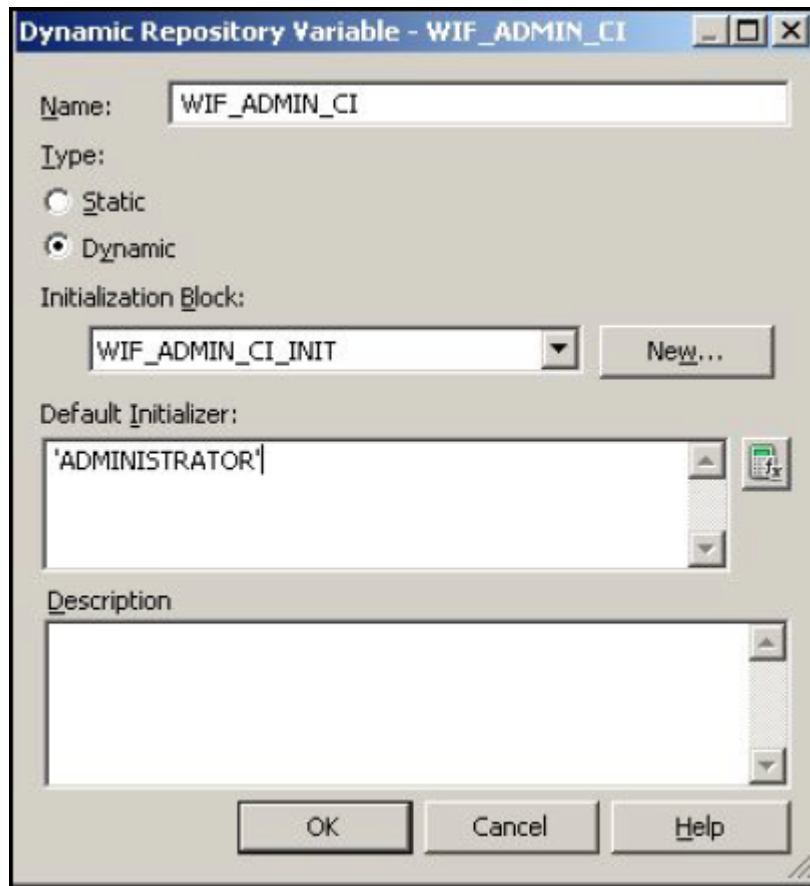9. Edit the default initializer to enter the desired What-If Administrator role. The user with this role will have the privilege to create and save a scenario. Users without this privilege will only be able to create a scenario, but not save it.

10. If the webserver is Tomcat of version >= 8.0.18, the 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"/>
```

17.3 OFS PA Integration with OFS Price Creation and Discovery (PCD)

The input from OFS PA to OFS PCD is 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.

17.4 Reporting Line Correlation Calculations

For the reporting lines, regression coefficients are calculated using the R-model based on the threshold values. It is considered that a pairwise relationship exists between independent and dependent reporting lines.

In what-if analysis, you can make variations to the value of a variable. Variations can be applied only to the following 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:

- The start date of the reference period

- The end date of the reference period

- Percentage of values that lie within the threshold

- Percentage of outliers that need to be removed

The following steps are used in repline correlation calculation:

- **Excluding Outliers**: For each variable, the sigma and mean are calculated within the reference period as defined in the database. If the value of the variable lies outside 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.

    add.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 co-integration test.

- **Testing for Co-integration:** After the stationary test is done, the causal relations between regression variables are checked. Then co-integration is done.

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

Table 44: Action performed Pairwise Stationary and Pairwise Co-integration

Pairwise Stationarity	Pairwise Co-integration	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

17.5   What-If Analysis Limitation

There is a limitation on the graph. Default upper limit for the graph will be based on underlying data in the 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 the graph, it is not possible through the graph and hence, it should be applied through the 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.

18 Visibility

Visibility is implemented to restrict the user's access to the data. The user can view based on the role and the privileges assigned to the user.

18.1 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 OFS Performance Analytics Installation and Configuration Guide, Release 8.1.1.0.0. 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 OFS Performance Analytics Installation and Configuration Guide, Release 8.1.1.0.0).

The following diagram depicts a hierarchy of Managers:
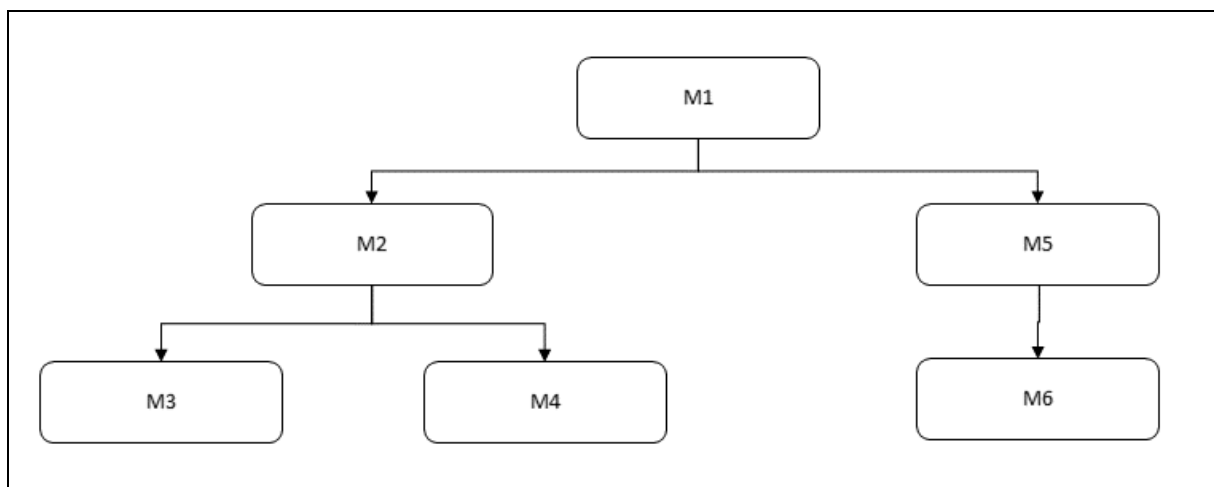
The data visibility for each of the Managers, starting from the top of the hierarchy is as follows:

- 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.

See the OBIEE documentation about *Setting Up Row-Level Security (Data Filters) in the Repository* if data visibility must be extended.

19   Appendix A: How to Define a Dimension

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

19.1   Create a Business Hierarchy

To create a business hierarchy, follow these steps:

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

Figure 136: Business Hierarchy

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

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

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

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

3. Select the **Hierarchy Sub Type**. The Hierarchy Sub Type can be:

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

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

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

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

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

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

   The components for the hierarchy definition differ for each subtype of the hierarchy.

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

   - If the subtype is Business Intelligence Enabled, then the user can specify the levels and SQL expressions for each level within the hierarchy.

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

For more details, see the [OFS Analytical Applications Infrastructure User Guide](#).

19.2   Create a Business Dimension

To create a business dimension, follow these steps:

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

Figure 137: Business Hierarchy Details Page

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

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

19.3   Modify a Dataset

To modify a dataset, follow these steps:

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

Figure 138: Data Set Details Page

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

3. Edit the data set definition.

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

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

6. Save the data set.

19.3.1 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:

Figure 139: Parent-Child Hierarchy

Figure 140: Relationship Manager Hierarchy
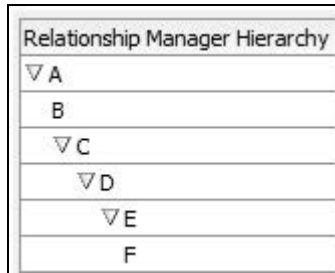
The first hierarchy is generated by the RDBMS source and the second is generated by the ESSBASE source. The additional parents are the Hierarchy Name and the Dimension Name of the metadata bearing the hierarchy.

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 the Relationship Manager Hierarchy is selected along with Direct Movement, no results are displayed.

Figure 141: Selected Columns
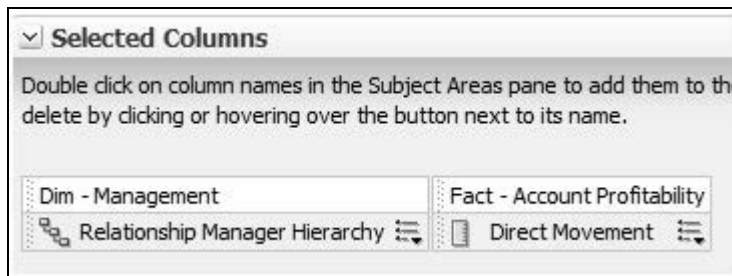
**Figure 142: Compound Layout**
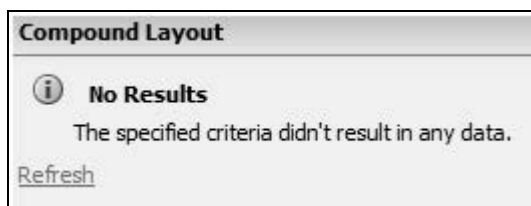
3. To view results for the logged-in Relationship Manager, the user must choose the Relationship Manager who is mapped to the user. In this case, the logged-in user is WebLogic. From FSI_M_USER_MANAGER_MAP, the following is seen:

Figure 143: FSI_M_USER_MANAGER_MAP

	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.

Figure 144: Edit Member Setup
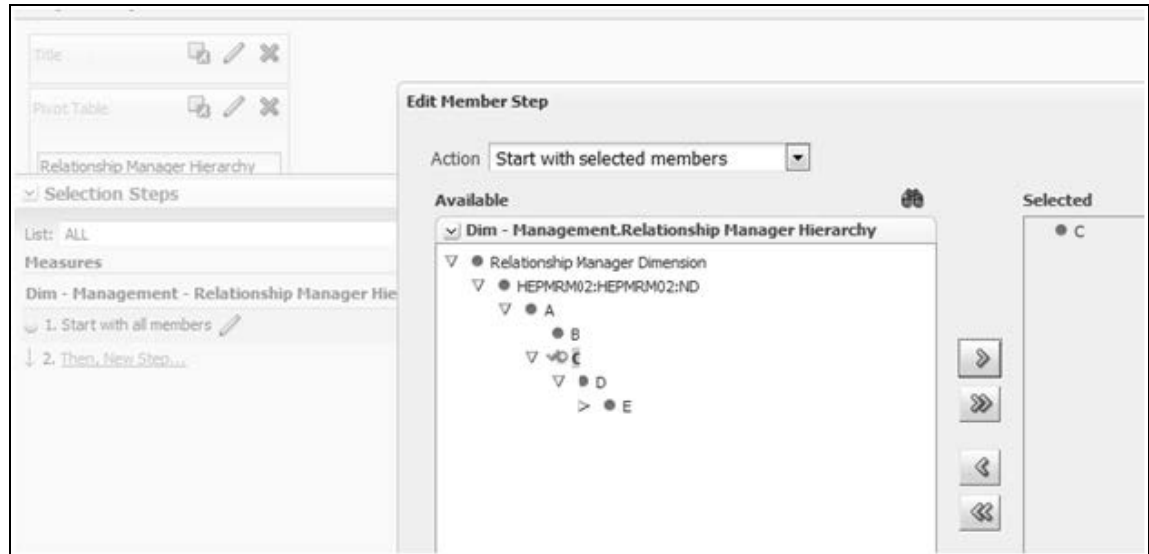
As a result, the user will be able to see the data related to the manager.

Figure 145: Result after Edit Member Setup

Relationship Manager Hierarchy	Direct Movement
▽ C	-827.25
▽ D	-827.25
▽ E	-827.25
F	-1611.25

19.4 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.

19.5 Optional Metadata

- PFT Acc_Sum_tech.xlsx lists the technical metadata related to the PFT account summary.
- FTP Acc_Sum_tech.xlsx lists the technical metadata related to the FTP account summary.

19.6 Business Metadata

- OFS RPA Business metadata.xlsx lists the Oracle Financial Services Retail Performance Analytics BI 6.0 Business Metadata.

19.7 Reporting Metadata

- Customer Attributes - RPA.xlsx lists Customer Attributes.
- RPA-RPD_webcat.xlsx lists the RPA-RPD-Webcat metadata.

20 Appendix B: How to Define a Measure

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

To create a business measure, follow these steps:

1. From the Unified Metadata Manager, select Business Metadata Management, then select Business Measures.

Figure 146: Business Measure Details Page

2. Click the **Add** icon to create a Business measure definition. In the **Business Measure Definition (Add mode)** window, select the **Aggregation Function**. The Aggregation Function can be:

   - **SUM**: for summing up the values in the column of the fact table.
   - **COUNT**: for determining the number of records in the fact table.
   - **MAXIMUM**: for identifying the maximum value of a column in the fact table.
   - **MINIMUM**: for identifying the minimum value of a column in the fact table.
   - **COUNT DISTINCT**: for determining the distinct count of records in the fact table.

3. Specify if this measure needs to be rolled up against hierarchies.

4. Select the fact table as part of the Entity.

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

6. Specify **Business Exclusions** and **Filters**, if required.

7. Save the measure.

21   Appendix C: How to Develop a New Cube

This section details the steps to be performed by the user for developing a new cube. Ensure that the existing cubes do not provide the required analytics or reporting coverage before deciding to define a new cube. In case the 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, a user should have defined datasets, measures, hierarchies, and dimensions before defining a cube.

21.1   Procedures to Develop a New Cube

1. Add Cube

    From the **Unified Metadata Manager**, select **Business Metadata Management**, and then select **Cube**. Specify the MDB details that will be created in ESSBASE.

2. Include Dimensions

    Include the dimensions that are part of the cube definition. Users mandatorily need to include TIME and MEASURE dimensions.

3. Specify Variations

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

4. Specify Dataset

    Specify the dataset corresponding to the selected dimensions and measures. The data set supplies the required data to the cube.

5. Specify Node Level Formula

    If node level formulae are required for the nodes within the hierarchy, then they can be specified on this screen.

6. Save and Build

    Save the cube. Define and execute batch in ICC to build the cubes.

For more information on Cubes, see the *Cubes* section under the *Unified Metadata Manager* chapter in the [OFS Analytical Applications Infrastructure User Guide](#).

22 Appendix D: Define a Batch

Batch refers to a set of executable processes based on a specified rule. The 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 a Data Centre Manager function role mapped to access the Operations framework within OFSAAI. You can access Batch Maintenance by expanding the Operations section within the tree structure of the 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.

22.1 Batch Creation

To create a batch, follow these steps:

1. From the **OFSAAI Home** menu, navigate to **Operations**, and then select **Batch Maintenance**.

2. In the Batch Maintenance window, select the + icon from the Batch Name toolbar.

   The New Batch Definition window is displayed.

3. Enter the following Batch details:

   - 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.

   - Duplicate Batch: (Optional) Select the check box to create a new Batch by duplicating the existing Batch details.

     On selection, the Batch ID field is enabled.

   - Batch ID (If duplicate Batch is selected): It is mandatory to specify the Batch ID if the 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 the Batch Maintenance window with the specified Batch ID.

For more comprehensive coverage of configuration and execution of a batch, see the *Operations* Chapter in OFS Analytical Applications Infrastructure User Guide.

23   Appendix E: List of Hardcoded Members

The following are the dimension members that are hardcoded within the application.

Table 45: Hardcoded 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

24   Appendix F: Run

The Run feature in the Rules Run Framework helps you to combine various components and/or processes together and execute them with different underlying approaches. Further, run conditions and/or job conditions can be specified while defining a run.

Two types of runs can be defined namely Base Run and Simulation Run.

- **Base Run** allows you to combine different rules and processes as jobs and apply run conditions and job conditions.

- **Simulation Run** allows you to compare the resultant performance/ calculations with respect to the baseline runs by replacing an existing job with a simulation job (a job can be a rule or a process). This comparison provides useful insights into the effect of anticipated changes on the business.

- **Instance Run** allows you to combine Base Runs and Simulation Runs in addition to other components from multiple information domains as Jobs. This eliminates the need for having different Run definitions if some Jobs are available in Hive Information Domain and some are present in RDBMS Information Domain.

The Roles mapped for the Run module are Run Access, Run Advanced, Run Authorize, Run Read Only, Run Write, and Run Phantom. Based on the roles mapped to your user group, you can access various screens in the Run module.

Figure 147: Run Parameters

The *Run* window displays the runs created in the current Information Domain with the metadata details such as Code, Name, Type, Folder, Version, and Active status. For more information on how

object access is restricted, see the *Object Security* section in the OFS Analytical Applications Infrastructure User Guide.

You can search for specific runs based on Code, Name, Folder, Version, Active status, or Type. The **Folder** drop-down list displays all Public folders, shared folders to which your user group is mapped, and Private folders for which you are the owner. The Pagination option helps you to manage the view of existing runs within the system.

24.1   Create Run

You can create run definitions using the existing metadata objects. The various components that can be used to form run definitions are mentioned in the section *Process Hierarchy Members* in the OFS Analytical Applications Infrastructure User Guide. The Write role should be mapped to your user group, from the *User Group Role Map* window.

The following filter conditions can also be applied to a run definition:

Table 46: Filter Conditions applied for Run Definition

Condition Type	Description
Run Condition	A Run Condition is defined as a filter and all hierarchies (defined in the current information domain) are available for selection. You can select up to 9 run conditions. A Run condition is defined for all Jobs. But it will be applied to a Job only if the underlying target/destination entities of both Job and Hierarchy are common.
Job Condition	A Job Condition is a further level of filter that can be applied at the component level. This is achieved through a mapping process by which you can apply a Job Condition to the required job. You can select only one Job Condition and the hierarchy that you have already selected as a run condition cannot be selected as the Job Condition again.

> **NOTE**   Filter conditions are not applicable for Instance Runs.

To create a run definition in the *Run* window, follow these steps:

1.   Click the **Add** icon from the toolbar. The *Run Definition (New Mode)* window is displayed.

Figure 148: Run Definition (Edit Mode)

2. Click the **View Attributes** icon adjacent to the **Folder** field in the *Linked to* the pane. The *Folder Selector* window is displayed. The folders to which your user group is mapped are displayed.

   a. Select the checkbox adjacent to the required folder. Click **OK**.

   b. Click the **Add** icon from the List toolbar to create a new folder/segment. For more information, see the section *Segment Maintenance* in the OFS Analytical Applications Infrastructure User Guide.

   c. Search for a folder by specifying any keyword and clicking the **Search** button.

3. Enter the details of the Master information pane as tabulated below:

Table 47: Master Information

Field Name	Description
ID	Refers to system-generated ID for a newly created run. When you create a rule, it is displayed as **<<New >>.**
Code	Enter a valid code for the run. Ensure that the code value specified is a maximum of 30 characters in length and it does not contain any special characters except "_". The code is unique and case sensitive. It is used to identify a run definition during execution. **Note**: You cannot use the same code of a rule which has been deleted from the UI.
Name	Enter a valid name for the run. Ensure that Run Name is alphanumeric and does not contain any of the following special characters: #, %, &, +, ", and ~. Note that the name is not required to be unique.

Field Name	Description
Version	By default, the version field is displayed as **<<NA>>** for the new run being created. Once the run definition is saved, an appropriate version is assigned as either **-1** or **0** depending on the authorization permissions. For more information, see the *Run Definition Versioning* section in the OFS Analytical Applications Infrastructure User Guide.
Active	By default, the Active field is displayed as **<<NA>>** for the new run being created. Once the run definition is saved, the status becomes **Yes** if you are an authorizer or **No** if the created Run needs to be authorized by an authorizer.
Type	Select the type of the run from the drop-down list. The available types are **Base Run, Simulation Run,** and **Instance Run**.
Route Execution to High Precedence Node	Select the checkbox if you want to route the execution of this Process definition to the high precedence node set up in the AM server.

4. Click the **Properties** icon in the Master information grid. The *Properties* window is displayed.

Figure 149: Properties window

You can edit the following details in the *Properties* window:

- **Effective Start Date** and **Effective End Date**: Effective Dates are not implemented for the Run definition.

- **Last Operation Type**: By default, this field displays the last change done to the run definition. While creating a run, the field displays the operation type as Created.

5. Click **OK**. The properties are saved for the current Run definition.

24.1.1 Select Run Condition for Run

You can select conditions to preset the initialization mechanism of a run definition.
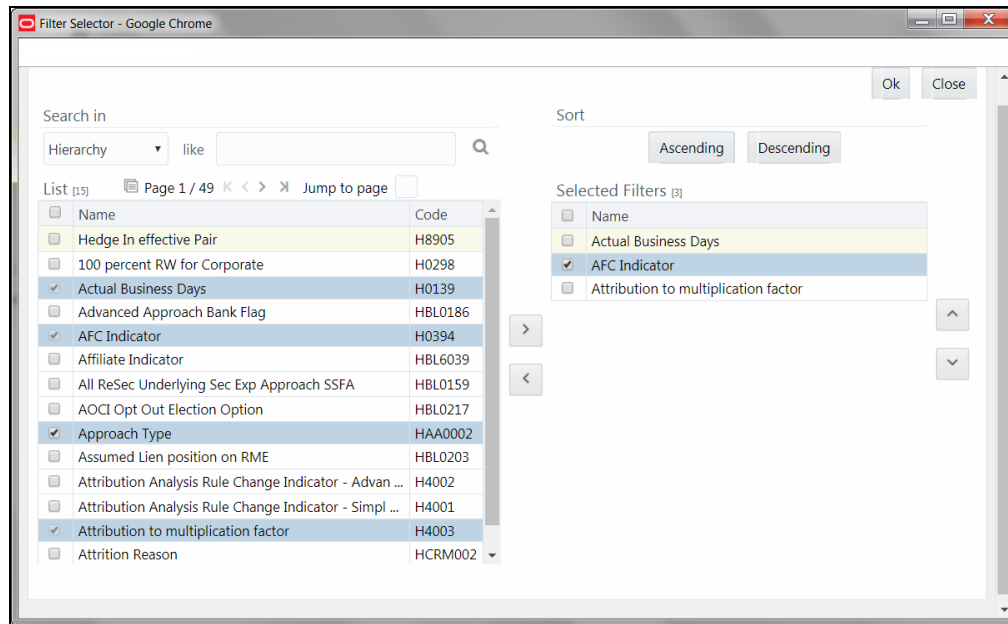
NOTE	Run Condition is not applicable for Instance Run.

To select a condition for a run in the *Run Definition (New Mode)* window, follow these steps:

1. Click the **Selector** icon from the List toolbar and select the **Run Condition** icon. The *Filter Selector* window is displayed.

Figure 150: Filter Selector

The List pane displays Hierarchies or Filters based on the option selected in the drop-down list in the *Search in* the pane. The options are:

- Hierarchy- Displays all Business Hierarchies defined in the information domain.
- Filter-Data Element- Displays all Data Element Filters defined in the information domain.
- Filter-Hierarchy - Displays all Hierarchy Filters defined in the information domain.
- Filter-Group - Displays all Group Filters defined in the information domain.
- Filter-Attribute - Displays all Attribute Filters defined in the information domain.

2. Select the checkbox adjacent to the Hierarchy or Filter that you want to select as the Run condition and click the **Move** icon.

   To know about the operations you can do in this window, see *Filter Selector Hierarchy_Selector* window in the OFS Analytical Applications Infrastructure User Guide.

3. Click **OK**. The selected Hierarchies are listed in the *Run Definition (New Mode)* window.

4. If the selected Run condition is a Parent-Child hierarchy, the **Use Descendants** checkbox is displayed. If the checkbox is selected for a hierarchy, the descendants will be automatically applied and need not be selected in node selection from the Hierarchy Browser window.

24.1.2 Select Jobs for Run

You can select the required jobs for the run definition being created.

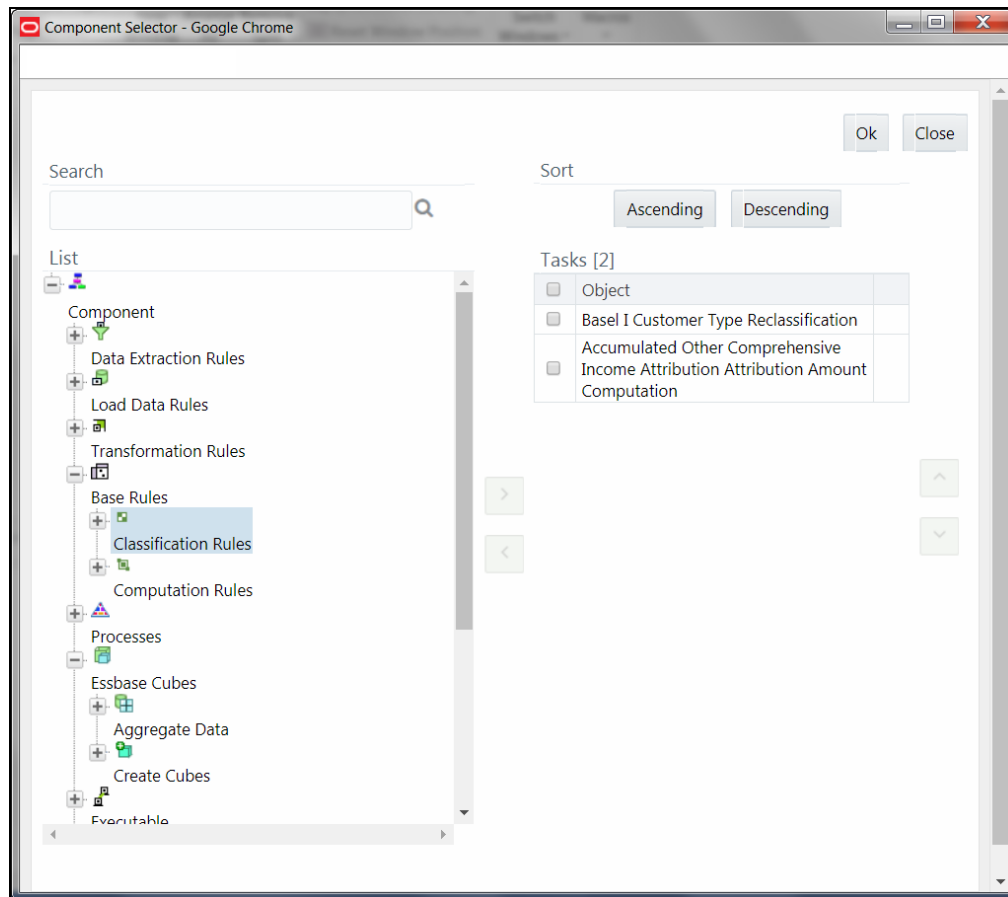To select jobs for Base and Simulation Run, follow these steps:

1. Click the **Selector** icon from the List toolbar and select the ▭ icon (**Job**). The *Component Selector* window is displayed.

Figure 151: Component Selector

On the **List** pane, you can click the **Add** button to expand the members and view the job components. For more information, see *Process Hierarchy Members* in the OFS Analytical Applications Infrastructure User Guide.

2. Select a job component and click the **Move** icon to move the component to the Tasks pane.

NOTE	You cannot select different Jobs with the same unique code in a run definition. In such cases, the Jobs should be added to a process and the process should be added to the run definition.

In the *Job Selector* window you can also:

- Search for a component by specifying the nearest keyword and clicking the **Search** icon. It may not display search results if the branch of that component has not been expanded.

- Click **Ascending** or **Descending** button to sort the selected components in ascending or descending alphabetical order.

- Click the ⌃ icon or the ⌄ icon to re-order the selected components.

- Click the ⊡ icon to add parameters for the selected components.

> **NOTE**   Parameters can be given in the format "param1","param1VALUE" or "$PARAM2","param2VALUE". Single quotes should not be used.

- Click the **Remove** icon to remove the selected components from the Tasks pane.

3. Click **OK**. The components are listed under the List pane in the *Run Definition* window.

To select Jobs for Instance Run

4. Click the **Selector** icon from the List toolbar and select the **Job** icon. The *Component Selector* window is displayed.

Figure 152: Component Selector
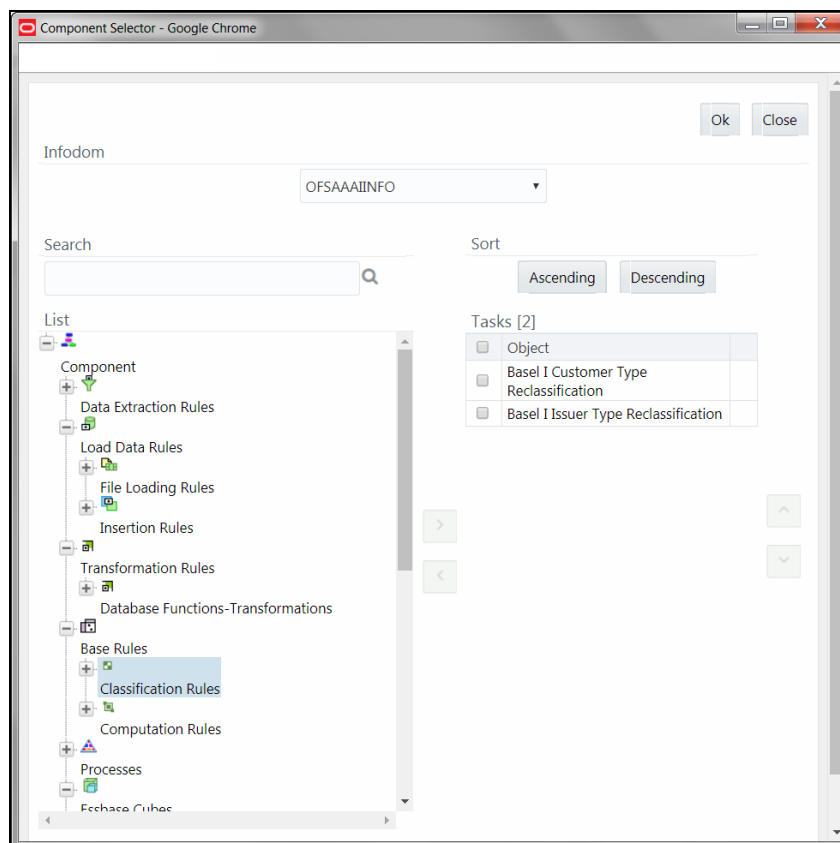
For Instance Run, you can add Base Run and Simulation Run as Jobs.

5. Select the information domain in which the job component you want to add is present, from the **Infodom** drop-down list. By default, the selected Application's Information Domain is displayed.

The drop-down list displays all information domains to which your user group is mapped except sandbox information domains.

6. Select a job component and click the **Move** icon to move the component to the Tasks pane.

   ▪ If you want to add a job component from another information domain, select the required information domain from the drop-down list. The Component list refreshes and you can add the required Job components.

   ▪ For more information see *Job Selector* in the OFS Analytical Applications Infrastructure User Guide

7. Click **OK**. The components are listed under the List pane in the *Run Definition* window.

24.1.3 Select Job Condition for Run

You can select only a single job condition for the execution of predefined jobs in a run. A hierarchy, which is already selected as a run condition, cannot be selected as a job condition.

> **NOTE** The Cumulative Node Expression for Hierarchy Nodes used as Job Condition in a Run definition should not cross 4000 characters. If it is exceeded, you will get an error while executing the Run definition.
>
> Job Condition is not applicable for Instance Run.

To select the job condition for a run, follow these steps:

1. Click the **Selector** icon from the List toolbar and select the **Job Condition** icon The *Filter Selector* window is displayed.

2. Select the checkbox adjacent to the hierarchy that you want to select as the Job condition and click the **Move** icon.

   To know about the operations you can do in this window, see *Filter Selector Hierarchy_Selector* window in the OFS Analytical Applications Infrastructure User Guide.

> **NOTE** Ensure that you have selected only one Job Condition and the same hierarchy is not selected as both Run and Job conditions.

3. Click **OK**.

   From the List grid in the *Run Definition (New Mode)* window, you can also:

   ▪ Click the **Edit** icon to change a selected run condition to job condition and conversely. For Instance Run, the **Edit** is disabled.

   ▪ Click the **Show Details** icon to view the metadata information of the selected member.

   ▪ If the selected Job condition is a Parent-Child hierarchy, the **Use Descendants** checkbox is displayed. If the checkbox is selected for a hierarchy, the descendants will be automatically applied and need not be selected in node selection from the Hierarchy Browser window.

Once all the necessary information in the first window of the Run Definition (New Mode) is populated, click **Next** to navigate to the concurrent procedures of defining a Rule.

Figure 153: Run Definition New Mode

The second window of the *Run Definition (New Mode)* window displays all the information you have provided in the Linked to and Master information grids. You can view the selected filters in the Run Condition grid and selected jobs along with the job condition in the Detail Information grid in the case of Base Run and Simulation Run. For Instance Run, only jobs will be displayed.

Expand a job which is a process, then the Object, Parent Object, Precedence, and Type columns are populated.

24.1.4 Hierarchical Member Selection

In the Run Condition grid, you can modify the run conditions by including hierarchical members.

> **NOTE**   This option will be available only if you have selected Hierarchy as the Run condition.
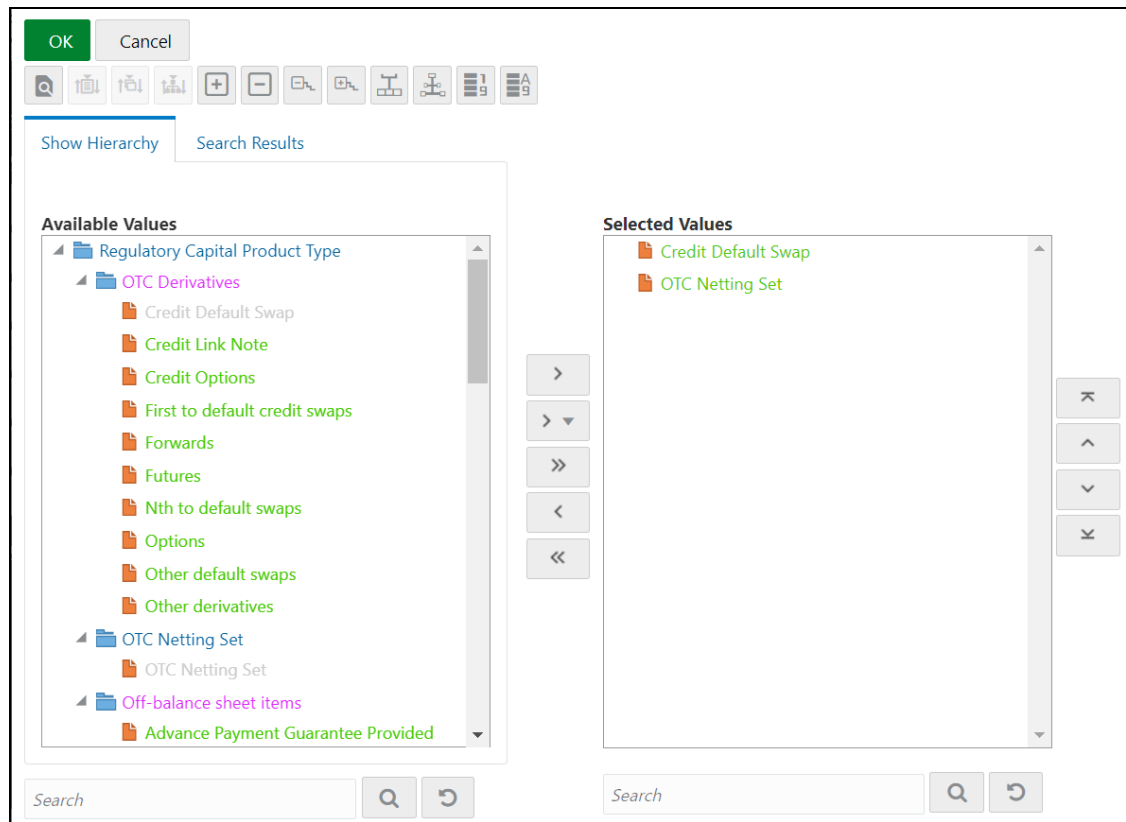
To modify a run condition, follow these steps:

1. Click the ▦ icon corresponding to the run condition you want to modify. The *Hierarchy Browser* window is displayed.

Figure 154: Hierarchy Selection

2. Select a member/node and click the **Move** icon to select the same. Click the ⯆ icon to select the member as Self, Self & Descendants, Self & Children, Parent, Siblings, Children, Descendants, or Last Descendants. For more information, see Hierarchical Member Selection Modes.

    In the *Hierarchy Browser* window you can also:

    ▪ Click ⊞↓ to sort members based on the path.

    ▪ Click ⊞↓ to sort hierarchy (top to bottom).

    ▪ Click ⊞↓ to sort based on level.

    ▪ Click ⊟ or ⊞ to collapse or expand the members under a node respectively.

    ▪ Click ⊟↵ or ⊞↵ to collapse or expand the selected branch respectively.

    ▪ Click ⟂ to focus only on the selected branch. The *Available Values* pane shows the members of the selected branch only. Click ⟂ to go back to normal view.

    ▪ Click ▤ to display member's numeric codes on the right. The icon changes to ▤ .

    ▪ Click ▤ to display member's numeric codes on the left. The icon changes to ▤ .

- Click ⬚ to show only member names. This is the default view. The icon changes to ⬚.

- Click ⬚ to display member's alphanumeric codes on the right. The icon changes to ⬚.

- Click ⬚ to display member's alphanumeric codes on the left. The icon changes to ⬚.

- Click ⬚ to display only member names. This is the default view. The icon changes to ⬚.

- Select a member and click ⬚ or ⬚ to re-arrange the members in the *Selected Values* pane.

- Select a member and click ⬚ to move it to the top or click ⬚ to move it to the bottom.

- Click ⬚ to launch the Search panel. Here you can search based on **Dimension Member Numeric Code**, **Dimension Member Name,** or **Dimension Member Alphanumeric Code**. You can also search in the grid based on member name using the **Search** field.

3. Click ⬚ to the run condition to view the SQL query. The SQL query is formed based on the hierarchical member selection mode. The *Preview SQL Query* window is displayed with the resultant SQL equivalent of the run condition.

   The *Detail Information* grid displays the jobs and job conditions defined for the run definition.

   - Click the **View** icon adjacent to the job names to re-order the selected jobs.

   - Click ⬚ beside the job condition to launch the *Hierarchy Browser* window. This option will be available only if a Hierarchy is selected as the Job condition.

   - Select the checkbox corresponding to the job if you want to apply the Job condition to that job.

   - Click a job to view its definition details. For example, if it is a Rule, the *Show Details* window displays the *Rule Definition (View Mode)* window.

   You can click **Back** to navigate back to the first page of the *Run Definition (New Mode)* window to modify any details.

   Once all the necessary details are entered, click **Save**. If you are an authorizer, the version of the run definition will be **0**, else it will be **-1**.

   The Audit Trail section at the bottom of the *Run Definition (New Mode)* window displays metadata information about the Run definition created. The User Comments section facilitates you to add or update additional information as comments.

24.2   View Run Definition

You can view individual run definition details at any given point. To view the existing Run definition details in the *Run* window, follow these steps:

1. Select the checkbox adjacent to the Run Code whose details are to be viewed.

2. Click the **View** icon in the List toolbar.

The *Run Definition (View Mode)* window is displayed with all the details of the selected Run. Click the **Next** and **Back** buttons to navigate back and forth in the Run *Definition (View Mode)* window.

24.3   Edit Run Definition

You can modify all the details except ID, Code, Version, Active Status, and Type of a run definition. To modify an existing run definition in the *Run* window, follow these steps:

1. Select the checkbox adjacent to the Run Code whose details are to be updated.

2. Click the **Edit** icon in the List toolbar. The Edit button is disabled if you have selected multiple Runs. The *Run Definition (Edit Mode)* window is displayed.

3. Edit the Run details as required. For more information, see <ins>Create Run</ins>.

4. Click **Save** to save the changes.

24.3.1   Run Definition Versioning

For an authorizer:

When you create a new run, its version will be **0.** When you edit an existing run and try to save it, you are prompted whether to save it as a new version or not. If you click **Yes**, a new run is created with the version as **0** and the run having version as **0** will be saved with the version as maximum version +1. If you click **No**, the existing run is overwritten and the version will be as it is.

For a non-authorizer:

When you create a new run, its version will be **-1**. Once the run is approved by an authorizer, the version becomes **0**. When you edit an existing run and try to save it, you are prompted whether to save it as a new version or not. If you click **Yes**, a new run is created with the version as **-1**. Once the run is approved, its version becomes **0** and the run having version as **0** will be saved with the version as maximum version +1. If you click **No**, the existing run is overwritten, and the **Active** flag of the run becomes **N** (which you can view from the *Summary* window). The version remains the same. Once the run gets approved, its **Active** flag changes to **Y**.

NOTE	• The run with version 0 is the latest one and it can have many versions say 1 to n, where 1 is the oldest run and n is the next to the latest.
	• A run with version -1 will always be in an Inactive state.

You can view all the versions of a particular rule by providing the run's name or code and clicking **Search** in the Search and Filter grid. (Ensure the **Version** field is cleared since it is auto-populated with **0**).

24.4   Copy Run Definition

This option facilitates you to quickly create a new run definition based on an existing run by updating the values of the required fields.

To copy an existing Run Definition in the *Run* window, follow these steps:

1. Select the checkbox adjacent to the Run Code whose details are to be duplicated.

2. Click the **Copy** icon in the List toolbar to copy a selected Run definition. The *Run Definition (Copy Mode)* window is displayed. Copy button is disabled if you have selected multiple Runs.

   In the *Run Definition (Copy Mode)* window you can:

   ▪ Create a new Run definition with existing variables. Specify a new **Run Code** and **Folder**. Click **Save**.

   ▪ Create a new Run definition by updating the required variables. Specify a new **Run Code**, **Folder**, and update other required details. For more information, see <u>Create Run</u>. Click **Save**.

The new Run definition details are displayed in the *Run* window. By default, version **0** is set if you have authorization rights, else the version is set to **-1**.

24.5   Authorize Run Definition

All the actions in a run definition should be approved by an authorizer. An authorizer can approve a pre-defined run definition for further execution or reject an inappropriate run definition listed within the *Run* window. To approve/ reject run definitions in the *Process* window, you need to have the Authorize role mapped to your user group.

If you are an authorizer, the run definition is auto-approved as you save it and the **Active** status is set to **Yes**. Otherwise, the **Active** status is set to **No** and an authorizer needs to approve it to change the **Active** status to **Yes**.

To approve/reject runs, follow these steps:

1. Select the checkbox(s) adjacent to the required Run Codes.

2. Do one of the following:

   ▪ To approve the selected run definitions, click **Authorize,** and select **Approve**.

   ▪ To reject the selected run definitions, click **Authorize** and select **Reject**.

A run is made available for use only after approval. For a rejected definition a comment with the rejection details will be added.

24.6   Export Run to PDF

This option allows you to export multiple run definitions to a PDF file. You have the option to export only the rules or processes in the run definition to PDF by selecting the required Trace Options. In the case of Instance Run, you can select Runs that you want to export, apart from Rules and Processes.

To export the run definitions in the *Run* window, follow these steps:

1. Select the checkbox(s) adjacent to the required Run Codes.

2. Click the icon (Export) in the List toolbar and click the PDF button in the pop-up window. The Export dialog box is displayed.

Figure 155: Run Rule Framework – Export Options
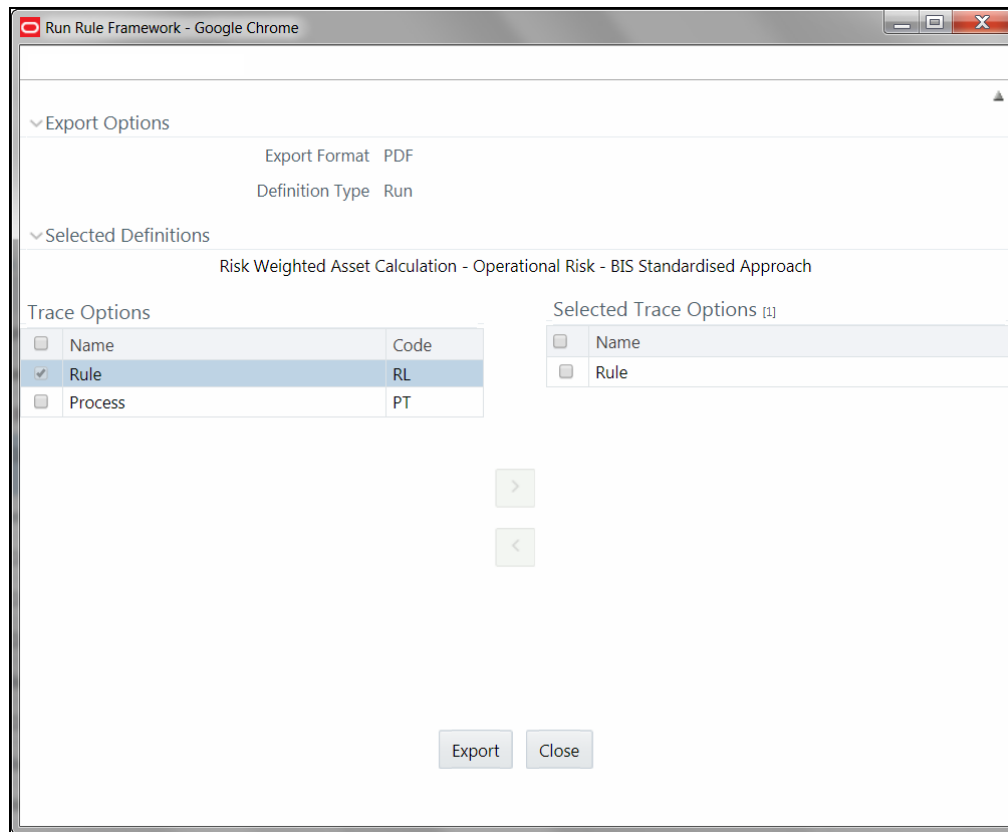
The Export dialog displays the Export Format, Definition Type, the names of the Selected Definitions, and the Trace Options.

- Select the checkbox adjacent to Rule or Process if you want to export only the rule details or Process details respectively. If you do not select any checkbox, all details of the selected run definitions will be exported.

- Click the **Move** icon. The selected options are displayed in the Selected Trace Options pane. You can also select a trace option and click the **Remove** icon to deselect it from the Selected Trace Options pane.

3. Click **Export**. The process is initiated and is displayed in a pop-up specific to the current download. Once the PDF is generated, you can open/save the file from the File Download dialog.

You can either save the file on the local machine or view the file contents in a PDF viewer. The downloaded PDF displays all the details such as Linked to, Properties, Master info, Audit Trail, List, and Comments of all the Run definitions selected.

24.7   Fire Run

This feature facilitates you to execute a previously created Run. You can execute the run definition as a batch from the Operations module.

To execute a run definition, follow these steps:
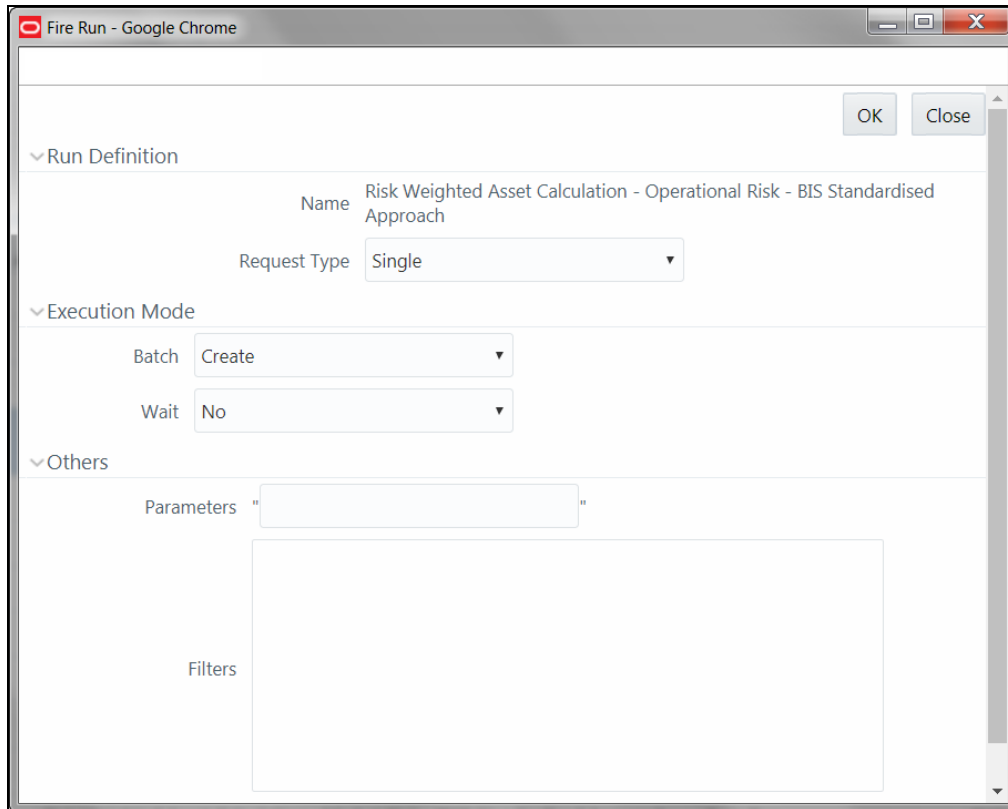
1. Select the checkbox adjacent to the Run Code which you want to execute and click ![icon] **Fire Run** in the List toolbar. The *Fire Run* window is displayed.

Figure 156: Fire Run

2. Enter the field details as below:

    a. **Name**: This field displays the name of the selected run.

    b. **Request** Type: Select the request type either as **Single** or as **Multiple** from the drop-down list.

        — **Single Request:** You need to provide the MIS Date during Batch execution from the Operations module.

        — **Multiple Request**: You can run the batch with the same MIS date multiple times from the Operations module.

    c. **Batch**: Select the Batch either as Create or as Create & Execute from the drop-down list

   — **Create**: The batch will be created and needs to be executed from the Operations module.

   — **Create & Execute**: The batch will be created and executed. You can monitor it from the Operations module.

d. MIS Date: Click the 📅 to display the Calendar. Select the MIS Date from the calendar.

   This field is displayed only if you have selected Request Type as **Multiple** with any of the Batch modes or Request Type as **Single** with Batch mode as **Create & Execute**.

e. Wait: Select **Yes** and provide the Duration in seconds after which the run definition should be executed.

   Select **No** to execute it immediately.

f. Parameters: Enter the required parameters in the field provided.

   The parameter provided in this field is considered for Run execution.

g. Filters: Enter the filter details in the field provided.

   The filters provided in this field are considered for Run execution.

3. Click **OK**. The details are saved and the run definition is executed as per the Fire Run details. For information on runtime parameters supported during run execution, see the Passing Runtime Parameters section.

24.8   Delete Run Definition

You can remove the Run definition(s) which are no longer required in the system by deleting them from the *Run* window. However, it is a soft deletion only. An authorizer has to approve the deletion.

To delete a run definition, follow these steps:

1. Select the checkbox(s) adjacent to the Run Codes whose details are to be removed.

2. Click the **Delete** icon from the List toolbar.

3. Click **OK** in the information dialog to confirm the deletion.

An information dialog is displayed confirming the deletion of the Run definitions and asking the authorization of the same.

25 Appendix G: Loading Multiple Load Runs in OFSAA

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 varies 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.

The load run is enabled only in the model and is defaulted to '0' in the model. This would not impact the data previously available.

FSDF staging model provides customers the 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. For customers who would like to leverage this design release, the following mentioned changes will need to be done as a workaround to load multiple snapshots 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 edit the file excludeURLList.cfg and add the following entry at the end of the file [SQLIA]./pr2

There should not be any blank line in the file.

This chapter discusses the following topics:

- Features
- Design Details
- Data Transformations

25.1 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. The 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 the past dates.

25.2 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 a 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 tables. Since OFSAA processing is on a snapshot of data, this entity helps users to identify a set of records that are the 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 the set of data is a 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 the set of data is incremental:

- Register load run information in a 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 snapshots and some entities incremental load. The decision of snapshot vs incremental depends on the above use case.

25.3   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 the uniqueness of the load run name. The 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 the user will query this table to get the load run id given a name).

5. Log messages accordingly.

6. Return success/failure.

25.3.1   Execution

Execute this DT before loading any fact for intra-day load. Use the registered load run identifier as a value to map to the 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 the load run name is used as a filter instead of the 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. A load run name or ID can be provided. If the load run name is provided, we can look up into the load run master for retrieving the ID. Check if the table name exists.

3. Register the information in the load run details table.

4. Log messages accordingly.

5. Return success/failure.

25.3.2   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 tables from a setup table.

3. If the record does not exist in the 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 the record exists in the load run map table for the functional key in staging, then update the 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.

25.3.3   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 approaches:

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 are four sources, then four distinct run skeys are generated for the load date. Some of the out-of-the-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 the Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary is 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 definition would be set to have a data source as the filter using the hierarchy defined earlier.

   For example, if there are three 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 runs are executed for the required MIS date. This approach requires as many run definitions for each of the sources which have 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 the atomic schema.

Following are the details of the approach:

11. 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 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
  ```

12. The Existing process "ACCOUNT PROFITABILITY PROCESS" is modified to have all the T2T's which are loading data into the Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary is 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 that 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 the 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 is 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 the same version.

13. 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.

14. New run definitions will be created for each data source which would be based on the same process "ACCOUNT PROFITABILITY PROCESS". Each run definition would be set to have a 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 wants to process, by repeating the above-mentioned process.

15. 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 the 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##
/
```

16. Create Database views in the 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##'
/
```

17. Modify the PL/SQL function "FN_FCT_ACCT_PFT_DT." Execute the script " FN_FCT_ACCT_PFT_DT.sql" in atomic schema.

18. Modify the existing DT "FN_FCT_ACCOUNT_PFT" for passing additional parameters to PL/SQL function "FN_FCT_ACCT_PFT_DT".

- Navigate to the Post Load transformation screen.

- Edit the definition "FCT_ACCT_TRANSFORMATION".

- Navigate to the Stored Procedure section.

- Copy and paste the contents of the file "fn_fct_acct_pft.sql" into the Stored Procedure Editor Section and save the definition.

- Execute the Run's as required.

26   Appendix H: 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.

26.1   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 48: 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)

Attribute Name	Datatype	Description and Acceptable Values
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.
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"` `}`

Attribute Name	Datatype	Description and Acceptable Values

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.

27   Appendix I: PA Metrics Computation Logic

Topics:

- [Projection Logic](#)
- [Data Treatment for Months](#)
- [Aggregation and Projection Logic for Forecasting of Replines under PA](#)

27.1   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 the last MIS Date, there is a projection on every projected date (generally the last date of every month)

PA Metrics calculation

The following table list the available web services (APIs) for OFSPA Cloud Service:

Table 49: Web Services (APIs) Available for OFS PA

Web Service	Input	Output	Version
Account Level Metrics (without UDM)	Account Number, Date	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income	
Customer Level Metrics (without UDM)	Customer Number, Date	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income	
Account Segmentation	Account Number, Date	Account Segment Code	
Customer Segmentation	Customer Number, Date	Customer Segment Code	
Metrics on existing Customer/Account with UDM	Account Number, Date of Report, {Date of UDM, UDM} repeated for all UDMs	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income, Customer ROTA, Customer RAROC, Customer ROE, Customer Total Expenses, Customer Total Income, Customer Net Income	
Metrics for New Account of existing Customer	Customer Number, Date of Report, Account Schedule	ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income, Customer ROTA, Customer RAROC, Customer ROE, Customer Total Expenses, Customer Total Income, Customer Net Income	
Metrics for New Account of New Customer	Date of Report, Account Schedule	Account Level ROTA, RAROC, ROE, Total Expenses, Total Income, Net Income	

NOTE	Segment Metrics are done through the batch process. The segment metrics considering the lifetime calculations of all the accounts in that segment.
	At present, Customer level segment metrics are based on the segment of the first account identified (by the system) of the customer. The segment averages as computed against the identified account segment would be reported.

27.2   Data Treatment for Months

Data to be considered for the 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 the 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 the first-month end date to the 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 purposes, to arrive at the Net Present Value (NPV) of the future stream of NIBT; a discount factor be applied to arrive at the expected value of NIBT.

1. **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 that can be used for forecasting.

2. **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 - 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, if there is no requested currency code, the customer would be showing in reporting currency and the Accounts are shown in its natural currency. In case of any requested currency code, both customer and currency would be shown in the 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 overall asset accounts)/(sum of ROTA denominator overall asset accounts)

  — RAROC: (Sum of RAROC Numerator overall asset accounts)/(sum of RAROC denominator overall asset accounts)

  — ROE: (Sum of ROE numerator overall accounts)/(Sum of ROE denominator overall 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.

27.3   Aggregation and Projection Logic for Forecasting of Replines under PA

Current Behavior/Functionality:

At present, the following Replines are considered for forecasting (@ (fsi model parameters table):

Table 50: Replines for Forecasting

N_REP_LINE_CD	Repline Name	Whether Forecasted
82100	Interest Income	Yes
82200	Transfer Pricing Credit	Yes
83300	Interest Expenses	Yes
90000	Transfer Pricing Charge	Yes
91900	Customer Non-Interest Income	Yes
94000	Non-Customer Non-Interest Revenue	Yes
107420	Indirect Non-interest Income	Yes
107501	Other Revenue	Yes
107502	Net Credit Losses	Yes
107503	Operating Expenses	Yes
107504	Non-Interest Income	Yes
107505	Outstanding Balance	Yes
87000	Utilization Rate	Yes
87500	Revolve Rate	Yes
90000	Purchases	Yes
86000	Payments	Yes

Observation:

1. From the above, Interest Income (82100) and Transfer Pricing Credit (82200) are two leaf-level Replines which are the members of the Node Total Interest Income (82000) are being forecasted while the other leaf lever members coming under Node Total Interest Income are not considered for forecasting

    All Leaf Level members of the Node Total Interest Income is given below:

Table 51: Leaf-Level Members of Node Total Interest Income

Leaf members of Node Total Interest Income		
82000	Total Interest Income	Node

Leaf members of Node Total Interest Income		
84000	Amortization of Discount for Liability	Leaf
84100	Amortization of Premium for Asset	Leaf
82900	Central Bank Int. Income	Leaf
83000	Credit for Float	Leaf
82800	Credit for Liquidity	Leaf
82700	Credit for Other Allocated Liabilities	Leaf
82300	Customer Break Funding Fees	Leaf
82100	Interest Income	Leaf, Forecasted
82200	Transfer Pricing Credit	Leaf, Forecasted

2. From the above, Interest Expense (83300) and Transfer Pricing Charges (90000) are two leaf-level Replines which are the members of the Node Total Interest Expenses (82001) are being forecasted while the other leaf lever members coming under Node Total Interest Expenses are not considered for forecasting.

Table 52: Leaf-Level Members of Node Total Interest Expenses

Leaf members of Node Total Interest Expenses		
82001	Total Interest Expense	This is a node
82500	Amortization of Discount for Asset	Leaf
82600	Amortization of Premium for Liability	Leaf
83500	Charge for Basis Risk	Leaf
83900	Charge for Central Bank Reserves	Leaf
83400	Charge for Liquidity	Leaf
83600	Charge for Optionality	Leaf
83700	Charge for Other Allocated Assets	Leaf
84500	Credit for Equity	Leaf
84200	Economic Provision	Leaf
82400	Pricing Incentive	Leaf
83300	Transfer Pricing Charge	Leaf, Forecasted
83200	Interest Expenses	Leaf, Forecasted

Reason for Considering Specific leaf-level Replines for forecasting in-place of Node level forecasting

Reason for forecasting interest income, Transfer Pricing Credit, interest expense, and Transfer Pricing Charges at leaf members and not their respective parent nodes, to apply variations on

interest income/expense and transfer pricing charge/credit separately in what-if analysis and monitor sensitivity at each of these leaf levels.

Doing projections, at the parent node, the level would not allow monitoring and reflecting variations at such micro levels; as both interest income & Transfer Pricing Credit belong to Parent Node Total Interest Income, as well as both Interest Expense and Transfer Pricing Charges belong to Parent Node Total Interest Expense.

Issues arising out of such an approach

If only the above leaf-level replines are projected, and other leaf-level replines under their respective node replines are not projected, leading to an incorrect aggregation of NIBT in forecasted values.

Suggested Approach

Repline Forecasting: IPA

- The leaf level members under Total Interest Income & Total Interest Expense, which are missed out in forecasts, should be aggregated to one head each respectively, and then forecasted.

- The head can be an additional node in the income statement hierarchy

Details: Other than the 4 leaf-level replines viz. Interest Income, Interest Expense, TP Charge, and TP Credit, aggregate the remaining replines under the nodes 'Total Interest Income' and 'Total Interest Expense' to form two notional replines under their heads respectively. Then forecast for these two notional replines and use for aggregation. These notional replines would be named "Other Interest Income" and "Other Interest Expenses"

The node replines of 'Total Interest Income' and 'Total Interest Expense' are not projected; instead only 4 child leaf replines from the hierarchy viz interest income/expense, TP charge/credit are projected.

Introduce two new replines, which would be aggregations of several replines as:

- Additional repline 1 (Other Interest Income) = sum(All leaf replines which are a child of Total Interest Income)-Interest Income - TP Credit

  When calculating total interest income; it will be, total interest income = other interest income+ interest income+ TP Credit

- Additional repline 2 (Other Interest Expenses) = sum(All leaf replines which are a child of Total Interest Expense)-Interest Expense - TP Charge

  When calculating total interest expense; it will be, total interest expense = other interest expense+ interest expense+ TP charges

These 2 new replines would be a part of the income statement hierarchy.

When aggregating for NIBT in the historical data, these two new replines would not get considered.

When aggregating for NIBT in projected data, the new replines 1&2 should roll up into Total Interest Income and Total Interest Expense respectively.

New Structure of Repline Hierarchy:

Total Interest Income:

Table 53: Leaf Members of Node Total Interest Income

Leaf members of Node Total Interest Income		
82000	Total Interest Income	Node
	+ Other Interest Income - New Repline	Node
	84000 Amortization of Discount for Liability	
	84100 Amortization of Premium for Asset	
	82900 Central Bank Int. Income	
	83000 Credit for Float	
	82800 Credit for Liquidity	
	82700 Credit for Other Allocated Liabilities	
	82300 Customer Break Funding Fees	
82100	Interest Income	Leaf, Forecasted
82200	Transfer Pricing Credit	Leaf, Forecasted

Total Interest Expense:

Table 54: Leaf Members of Node Total Interest Expenses

Leaf members of Node Total Interest Expenses		
82001	Total Interest Expense	This is a node
	+ Other Interest Expenses - New Repline	Node
	82500 Amortization of Discount for Asset	
	82600 Amortization of Premium for Liability	
	83500 Charge for Basis Risk	
	83900 Charge for Central Bank Reserves	
	83400 Charge for Liquidity	
	83600 Charge for Optionality	
	83700 Charge for Other Allocated Assets	
	84500 Credit for Equity	
	84200 Economic Provision	
	82400 Pricing Incentive	
83300	Transfer Pricing Charge	Leaf, Forecasted
83200	Interest Expenses	Leaf, Forecasted

Key Point:

- The New nodes other interest income and other interest expenses would be a part of the income statement hierarchy

- The forecasts required for PCD, Profitability, LTV, and What-if. The inputs would be required at this granularity.

Addendum and Update

- Repline Codes: 81000 (Other Interest Income) and 81001 (Other Interest Expenses) are finalized and accepted.

- Operational Signage: Operational Signage for Both Replines would be +. These replines are having other leaf replines under each, which are in turn aggregated to Total Interest Income (82000) and Total Interest Expenses (82001) respectively.

- The New nodes other interest income and other interest expenses would be a part of the income statement hierarchy

- The forecasts required for PCD, Profitability, LTV, and What-if. The inputs would be required at this granularity.

- These new Notional Replines need to be introduced under What-if Logic and for calculation under Advanced What-If Logic.

- What-if logic would be following the changes to be made in the Repline Hierarchy (on Aggregation and Projection Logic for forecasting of Replines under PA). In case of addition to any new (or notional repline), the same would be required to be introduced under what-if logic.

- For what if Advanced Logic; the newly introduced replines to be added to the calculation while establishing the impacts (happening through Linear Regression Approach).

OFSAA Support

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to OFSAA applications.

Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?

- Is the information clearly presented?

- Do you need more information? If so, where?

- Are the examples correct? Do you need more examples?

- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site that has all the revised/recently released documents.