Oracle Financial Services Retail Customer Analytics

User Guide

Release 8.1.2.0.0

December 2021

F41959-01

ORACLE
Financial Services

ORACLE

Document Control

Version Number	Revision Date	Change Log
1.0	Dec-2021	Created the user guide for the OFS Retail Customer Analytics application, Release 8.1.2.0.0.

Table of Contents

1 Preface

This section provides a brief description of the scope, the audience, the references, the organization of the user guide, and conventions incorporated into the user guide.

Topics:

- Foreword
- Intended Audience
- Documentation Accessibility
- Access to Oracle Support
- Structure
- Related Documents
- Important Information
- Conventions
- Abbreviations

1.1 Foreword

This user guide documents OFS Retail Customer Analytics for all versions of release 8.1.1.0.0

This section documents the levels at which various functional enhancements to the Retail Customer Analytics application were first introduced.

1.2 Intended Audience

Welcome to Release 8.1.1 of the Oracle Financial Services Retail Customer Analytics User Guide.

- Technical end users
- Functional end users
- Data Administrators
- Consultants
- Systems Analysts
- System Administrators
- Other MIS professionals

1.3 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/us/corporate/accessibility/index.html.See Related Information Sources for more Oracle Applications product information.

1.4 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For more 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.

1.5 Related Documents

We strive to keep this document and all other related documents updated regularly; visit the OHC Documentation Library to download the latest version available. The list of related documents for the Oracle Financial Services Retail Customer Analytics Application Pack (OFS RCA) is provided here:

- Oracle Financial Services Retail Customer Analytics Installation and Configuration Guide, Release 8.1.2.0.0
- Oracle Financial Services Retail Customer Analytics Release Notes, Release 8.1.2.0.0
- Oracle Financial Services Retail Customer Analytics Reports User Guide, Release 8.1.2.0.0
- Oracle Financial Services Retail Customer Analytics Security Guide, Release 8.1.1.x
- Oracle Financial Services Retail Customer Analytics Cloning Reference Guide, Release 8.1.1.x

1.6 Additional Documents to Read

Oracle Financial Services Retail Customer Analytics 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 Retail Customer Analytics Pack:

- OFS Analytical Applications Infrastructure (OFS AAAI) Application Pack Installation and Configuration Guide Release 8.1.2.0.0
- OFS Analytical Applications Infrastructure Administration and Configuration Guide Release 8.1.x
- OFS Analytical Applications Infrastructure User Guide Release 8.1.2.0.0
- OFS Analytical Applications Infrastructure Cloning Reference Guide Release 8.1.x
- OFS Analytical Applications Infrastructure Security Guide Release 8.1.x

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

- OFS Analytical Applications 8.1.2.0.0 Technology Matrix

1.7 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 and internal document links.

1.8　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
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

Abbreviation	Meaning
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
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
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

2   Overview of OFS Retail Customer Analytics

This guide explains the concepts of Oracle Financial Services Retail Customer Analytics (OFS RCA) and provides step-by-step instructions for navigating the OFS RCA user interface. OFS RCA 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 RCA 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 or 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 RCA helps you to monitor customer distribution across credit and delinquency bands and related exposures.

This OFS RCA is supported for Oracle 12c.

OFS RCA solution is built using:

- OBIEE 12.2.1.4 for Dashboard and Reports activities

- Essbase 11.1.2.3+ for 12c database

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

3   Overview of OFS RCA Process Flow

Oracle Financial Services Retail Customer Analytics (OFSRCA) 8.0 utilizes OBIEE technology to present:

- Performance tracking of current campaigns across key measures like Sales, Asset, and Liability balances, Fee-based product subscriptions and sustained performance over time, Credit score distribution of new accounts sourced, and early alerts on any negative skews.

- Predictive analysis to determine to cross-sell/ up-sell scores, product, and channel propensities leveraging transactional/ behavioral data.

- Return On Investment (ROI) of campaigns over time (transaction performance needs to be measured for at least over 12 months for accurate Lifetime Value (LTV) predictions).

- Prospect/ list scoring leveraging any internal/ bureau information, cluster analysis, and projected Net Present Value (NPV).

- Customer Segmentation.

- Wallet Share (spend diversity, activation, and so on) and Attrition analysis.

The following diagram explains the product objectives of OFS RCA:

Figure 1: Product Objectives
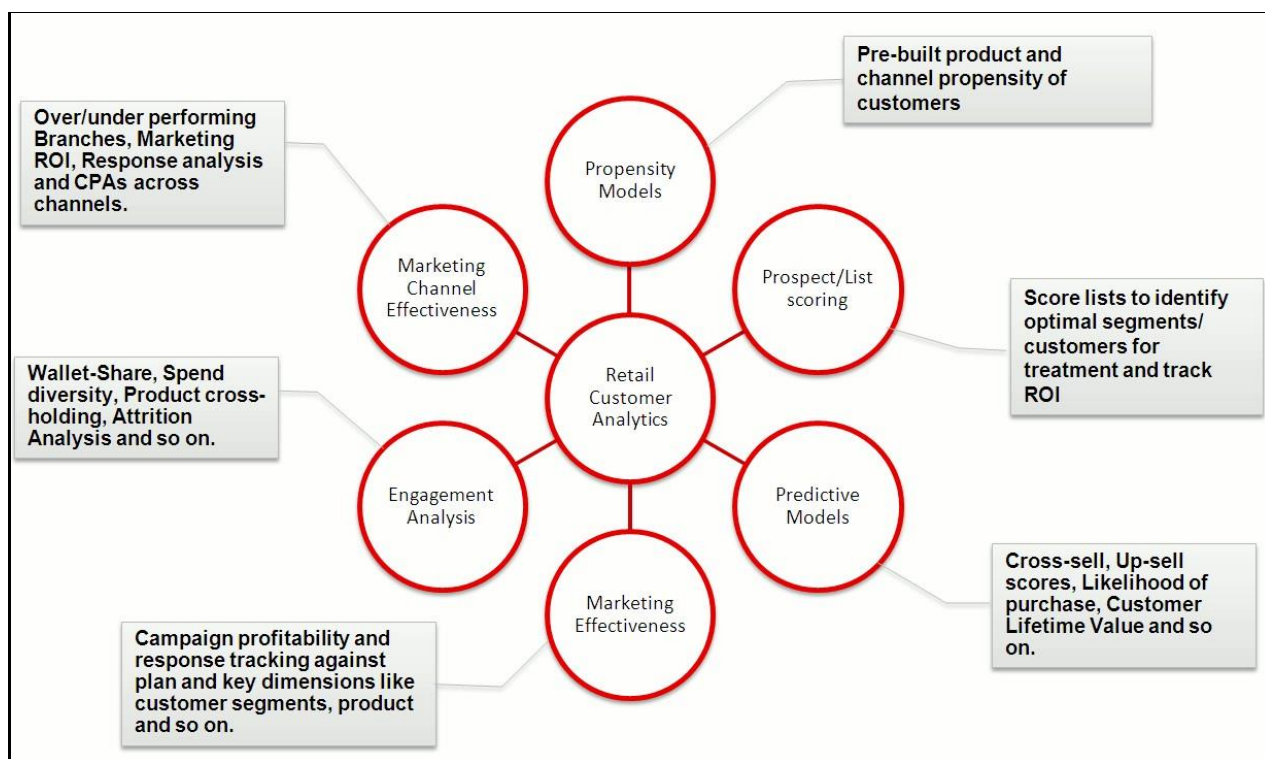
For details on OFS RCA reports and how OBIEE is being utilized, see Overview of OFS RCA Reports.

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

OFS RCA 8.1.1.0.0 can be independently licensed and installed to work on top of the OFSAA 8.1.1.0.0 Infrastructure.

3.1 Data Flow

Retail Customer 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, etc. The user has to populate data into these staging tables.

Figure 2: Staging Tables
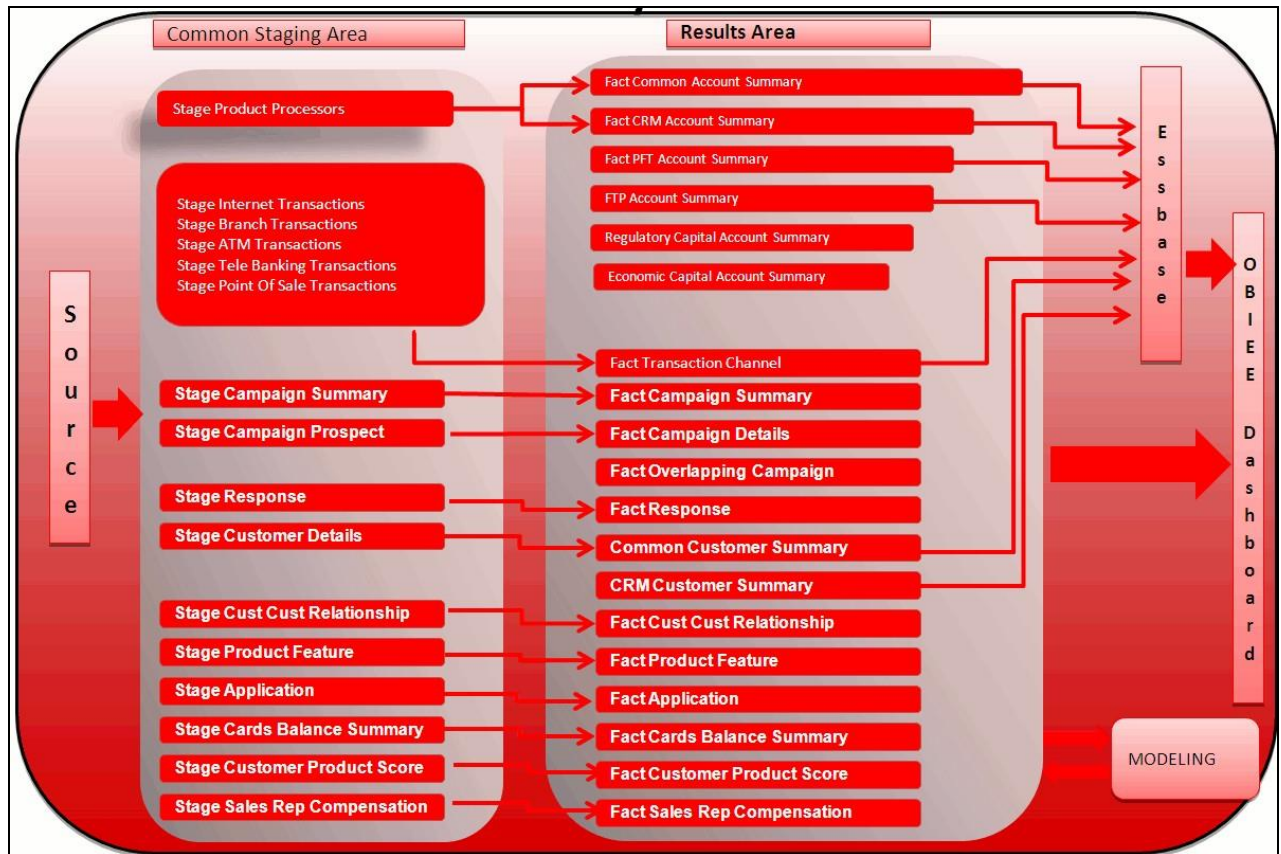
3.2 Dimension Data Flow

Dimension Data in the OFS RCA 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 are the list of dimensions used in OFS RCA:

Table 3: OFSRCA Dimensions

Dimension Entity Name	Staging Entity Name(s)	Loading or Maintenance method
Account Status Dimension	Stage Account Status Dimension	SCD
Campaign Source Type Dimension	Stage Campaign Source Type Dimension	SCD
Campaign Status Dimension	Stage Campaign Status Dimension	SCD
Campaign Type	Stage Campaign Type	SCD
Card Type Dimension	Stage Card Type Dimension	SCD
Channel Transaction Dimension	Stage Channel Transaction Dimension	SCD
Contact Dimension	Stage Contact Dimension	SCD
Country Dimension	Stage Country Dimension	SCD
Credit Center Dimension	Stage Credit Center Dimension	SCD
Credit Officer Dimension	Stage Credit Officer Dimension	SCD
Application Reject Reasons Dimension	Stage Application Reject Reasons Dimension	SCD
Pool Identification Dimension	Stage Pool Identification Dimension	SCD
Prepayment Reason Dimension	Stage Prepayment Reason Dimension	SCD
Product Dimension	Stage Product Dimension	SCD
Channel Dimension	Stage Channel Dimension	SCD
Cards Dimension	Stage Cards Dimension	SCD
Social Media Dimension	Stage Social Media Dimension	SCD
Social Media Post Dimension	Stage Social Media Post Dimension	SCD
Location Dimension	Stage Location Dimension	SCD
Request Type Dimension	Stage Request Type Dimension	SCD
Survey Dimension	Stage Survey Dimension	SCD
Service Rep Dimension	Stage Service Rep Dimension	SCD
Loan Product Category Dimension	Stage Loan Product Category Dimension	SCD
Product Feature Dimension	Stage Product Feature Dimension	SCD
Product Type Dimension	Stage Product Type Dimension	SCD
Prospect Dimension	Stage Prospect Dimension	SCD
Purchase Category Dimension	Stage Purchase Category Dimension	SCD
Rejection Reason Dimension	Stage Rejection Reason Dimension	SCD
Application Status Dimension	Stage Application Status Dimension	SCD
Retention Offer Type Dimension	Stage Retention Offer Type Dimension	SCD

Dimension Entity Name	Staging Entity Name(s)	Loading or Maintenance method
Terminal Dimension	Stage Terminal Dimension	SCD
Terminal Type Dimension	Stage Terminal Type Dimension	SCD
Transaction Dimension	Stage Transaction Dimension	SCD
Treatment Dimension	Stage Treatment Dimension	SCD
Transaction Channel Dimension	Stage Transaction Channel Dimension	SCD
Txn Failure Reason Dimension	Stage Txn Failure Reason Dimension	SCD
Transaction Status Dimension	Stage Transaction Status Dimension	SCD
Vendor Dimension	Stage Vendor Dimension	SCD
Application Type Dimension	Stage Application Type Dimension	SCD
Vintage Dimension	Stage Vintage Dimension	SCD
Wave Dimension	Stage Wave Dimension	SCD
Customer Type Dimension	Stage Customer Type Dimension	SCD
Decision Status Dimension	Stage Decision Status Dimension	SCD
Deviation Reasons Dimension	Stage Deviation Reasons Dimension	SCD
Education Dimension	Stage Education Dimension	SCD
Home Ownership Dimension	Stage Home Ownership Dimension	SCD
Household Dimension	Stage Household Dimension	SCD
Industry Dimension	Stage Industry Dimension	SCD
Legal Reporting	Stage Legal Reporting	SCD
Attrition Dimension	Stage Attrition Dimension	SCD
LoB Dimension	Stage LoB Dimension	SCD
Management Dimension	Stage Management Dimension	SCD
Market Cell	Stage Market Cell	SCD
Merchant Dimension	Stage Merchant Dimension	SCD
Merchant Category Dimension	Stage Merchant Category Dimension	SCD
Migration Reasons Dimension	Stage Migration Reasons Dimension	SCD
Marketing Program Dimension	Stage Marketing Program Dimension	SCD
Offer Dimension	Stage Offer Dimension	SCD
Organization Structure Dimension	Stage Organization Structure Dimension	SCD
Authorization Decision Reasons Dimension	Stage Authorization Decision Reasons Dimension	SCD
Geography Dimension	Stage Geography Dimension	SCD

Dimension Entity Name	Staging Entity Name(s)	Loading or Maintenance method
Response Type Dimension	Stage Response Type Dimension	SCD
Balance Category Dimension	Stage Balance Category Dimension	SCD
Campaign Dimension	Stage Campaign Dimension	SCD
Campaign Channel Dimension	Stage Campaign Channel Dimension	SCD
Account Dimension	Stage LC Contracts	SCD
Account Dimension	Stage Commitment Contracts	SCD
Party Dimension	Stage Party	SCD
Account Dimension	Stage Stage OD accounts	SCD
Account Dimension	Stage Stage TD contracts	SCD
Account Dimension	Stage Stage Trusts	SCD
Account Dimension	Stage Stage Loan Contracts	SCD
Account Dimension	Stage Stage Mutual Funds	SCD
Account Dimension	Stage Bills Contracts	SCD
Account Dimension	Stage CASA Accounts	SCD
Account Dimension	Stage Guarantees	SCD
Account Dimension	Stage Stage leases contracts	SCD
Account Dimension	Stage Stage mm contracts	SCD
Account Dimension	Stage Annuity Contracts	SCD
Account Dimension	Stage Borrowings	SCD
Account Dimension	Stage Card Accounts	SCD
Account Dimension	Stage Investments	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.

3.3 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 PFT Account Summary	Instrument	Annuity Contracts Borrowings Checking and Savings Account Credit Cards Credit Lines Guarantees Investments Leases Loan Contracts Mortgages Term Deposits Trusts	T2T
Fact FTP Account Summary	Instrument	Annuity Contracts Borrowings Checking and Savings Account Credit Cards Credit Lines Guarantees Investments Leases Loan Contracts Money Market Contracts Mortgages Term Deposits Trusts	T2T
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
Fact Account Feature Map	Stage	Stage Account Feature Map	T2T

Fact Entity Name	Source	Source Entities	Method of populating measures
Fact Customer to Customer Relationship	Stage	Stage Customer to Customer Relationships	T2T
Fact Campaign Details	Stage	Stage Campaign Prospect	T2T
Fact Campaign Execution Summary	Stage	Fact Campaign Details	T2T
Campaign Summary Facts	Stage	Stage Campaign Summary	T2T
Fact Overlapping Campaign	Stage	Fact Campaign Details	T2T
Response Facts	Stage	Stage Responses	T2T
Fact Cross Sell Score	Fact	Fact Common Account Summary	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

3.4 BI Data Model

The BI data model is a star schema for the fact tables, FCT_COMMON_CUSTOMER_SUMMARY, FCT_CRM_CUSTOMER_SUMMARY, and FCT_<Application>_ACCOUNT_SUMMARY. Following are the subject areas in the erwin data model:

Figure 3: Application
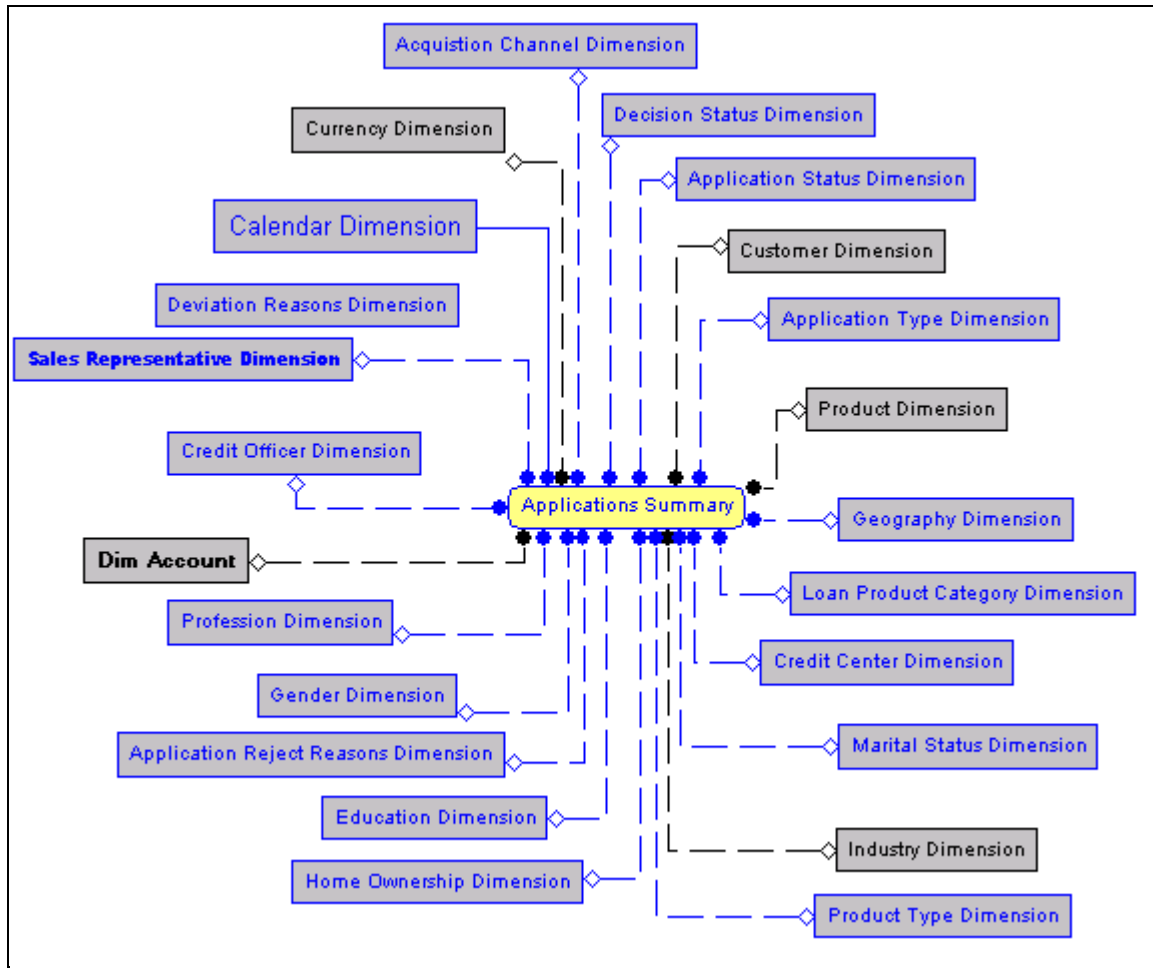
Figure 4: Campaign Details
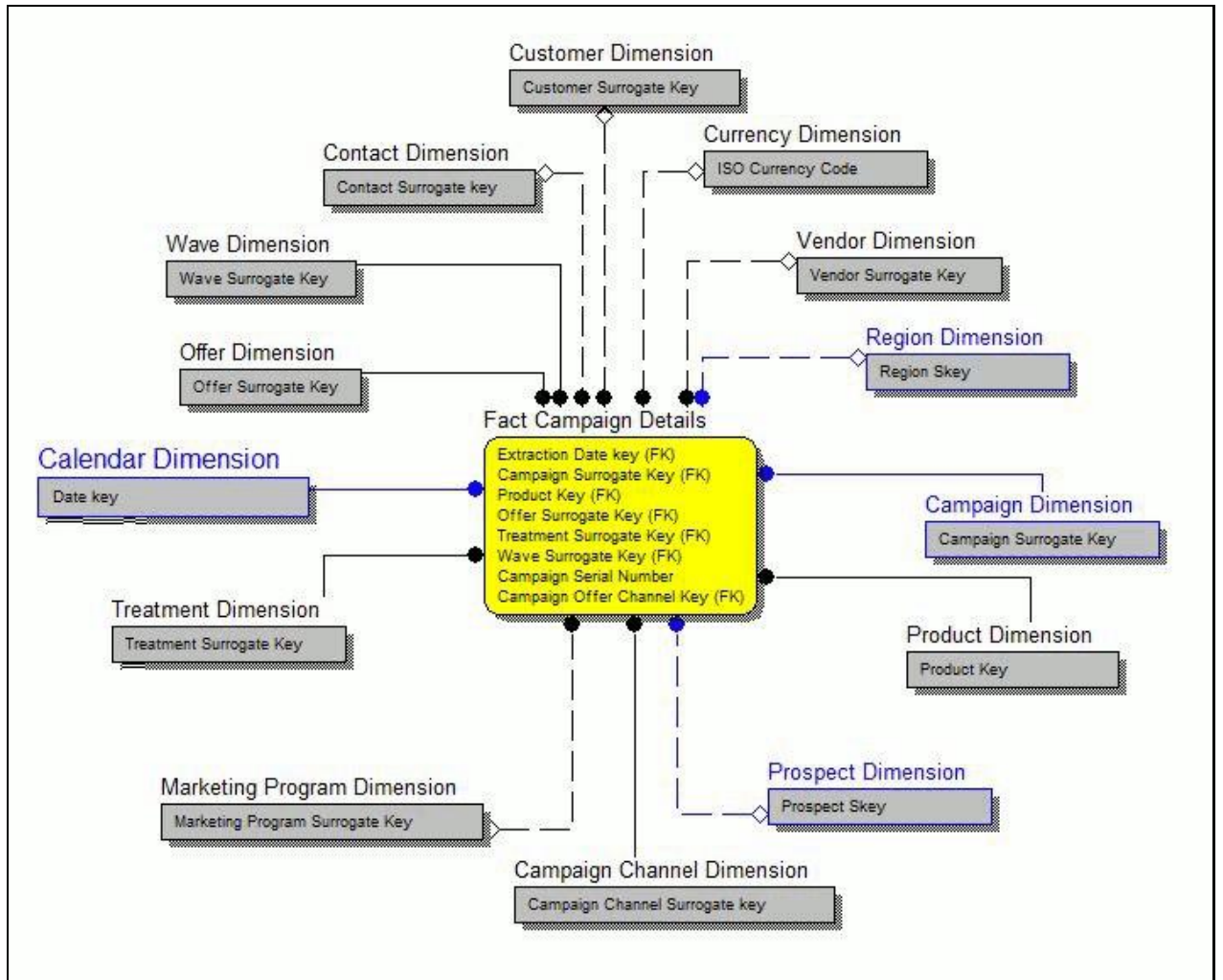
Figure 5: Campaign Execution Summary
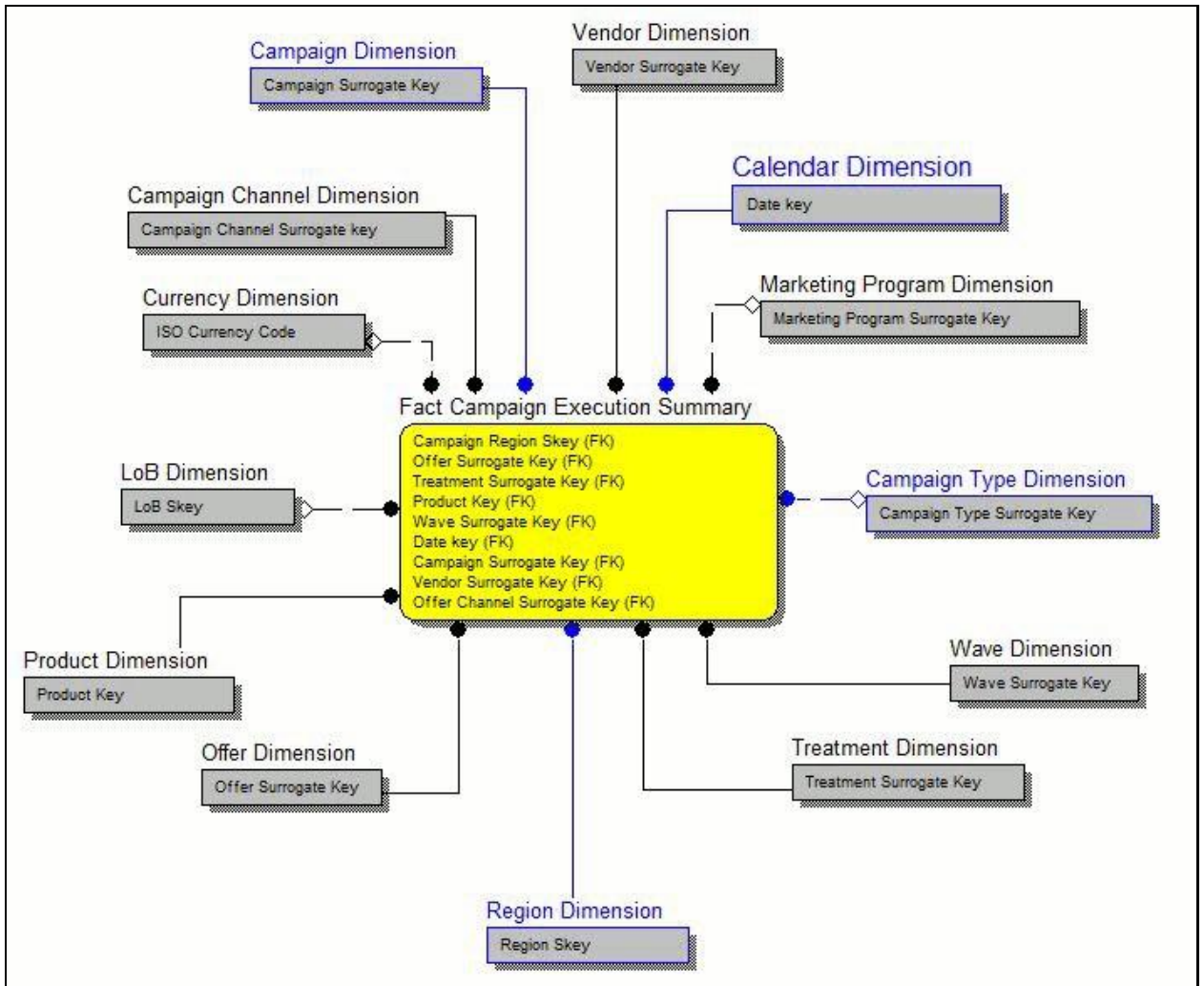
Figure 6: Campaign Response
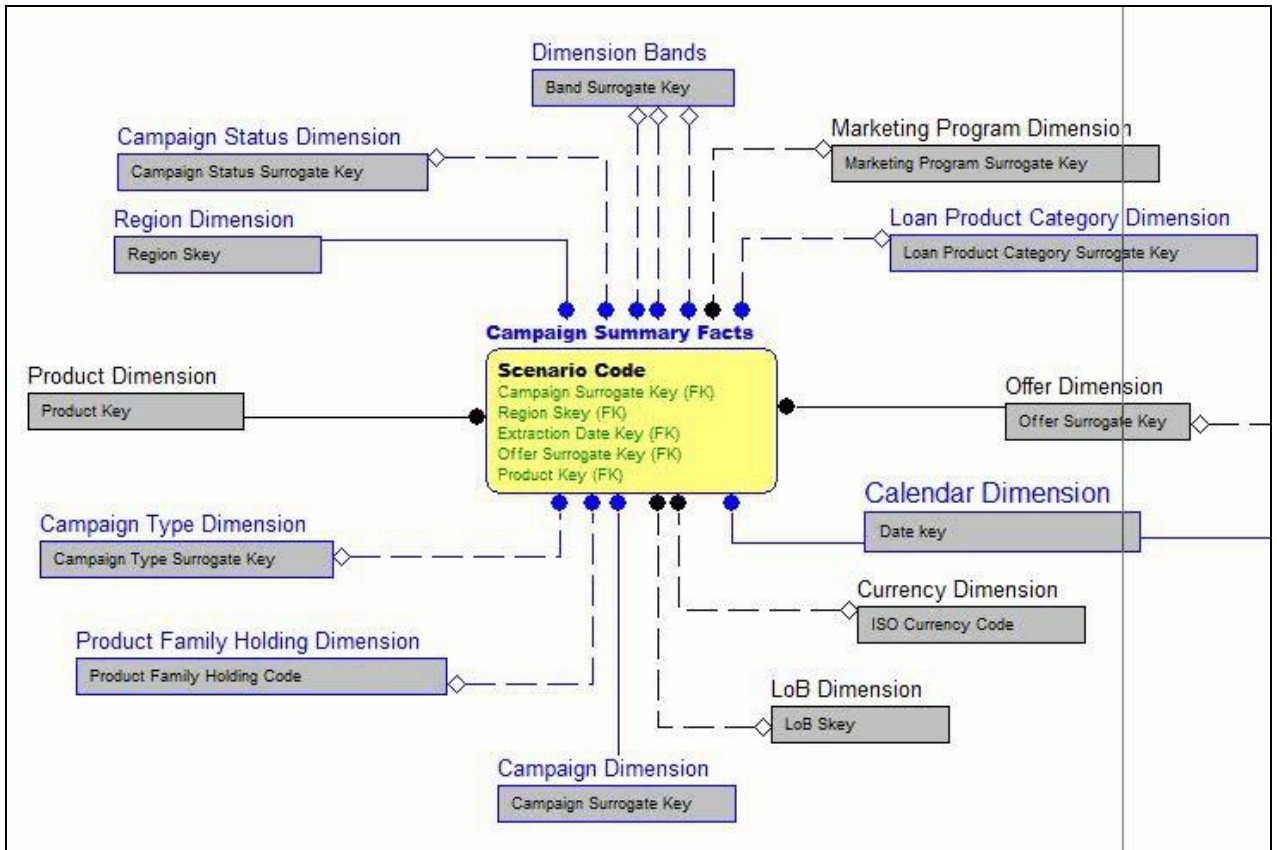
Figure 7: Summary

Figure 8: Cross Sell Score
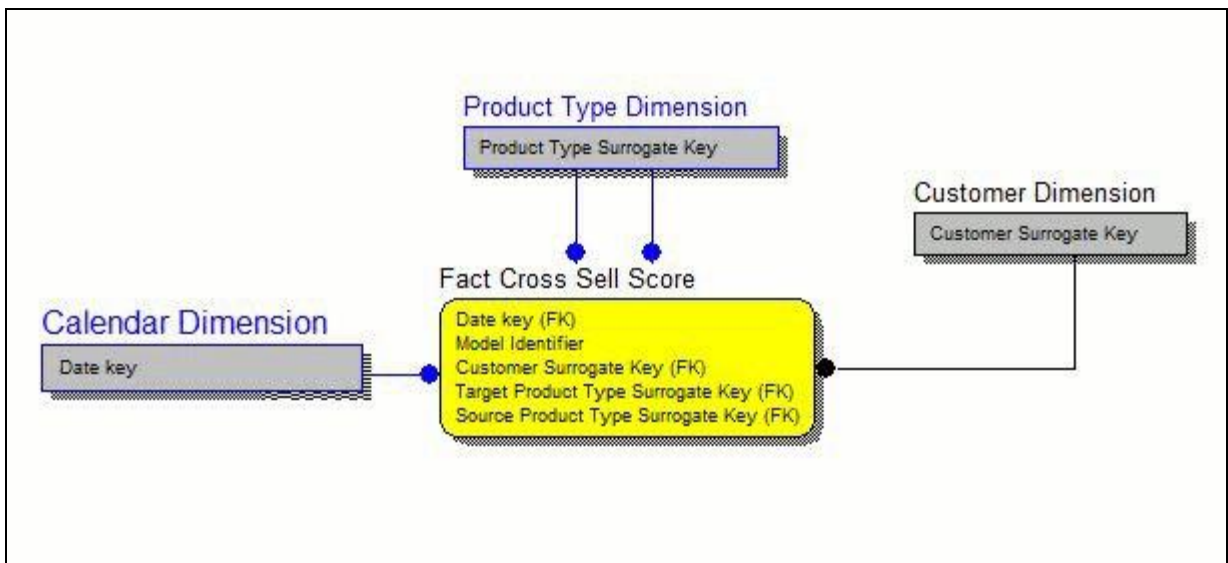
Figure 9: Customer to Customer Relationship
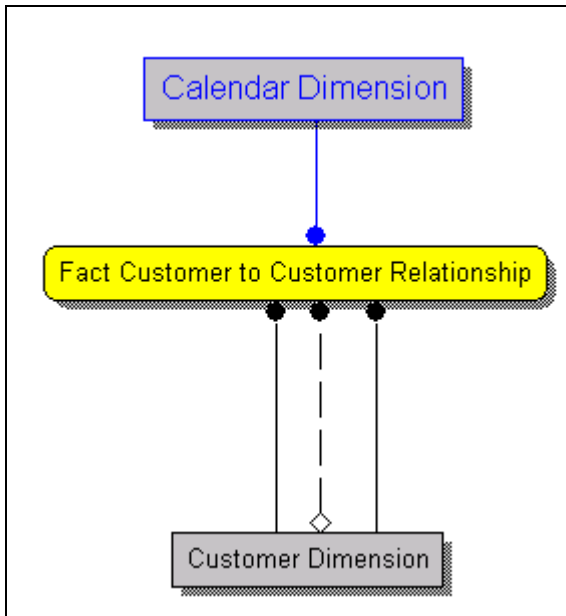
Figure 10: FTP Account Summary
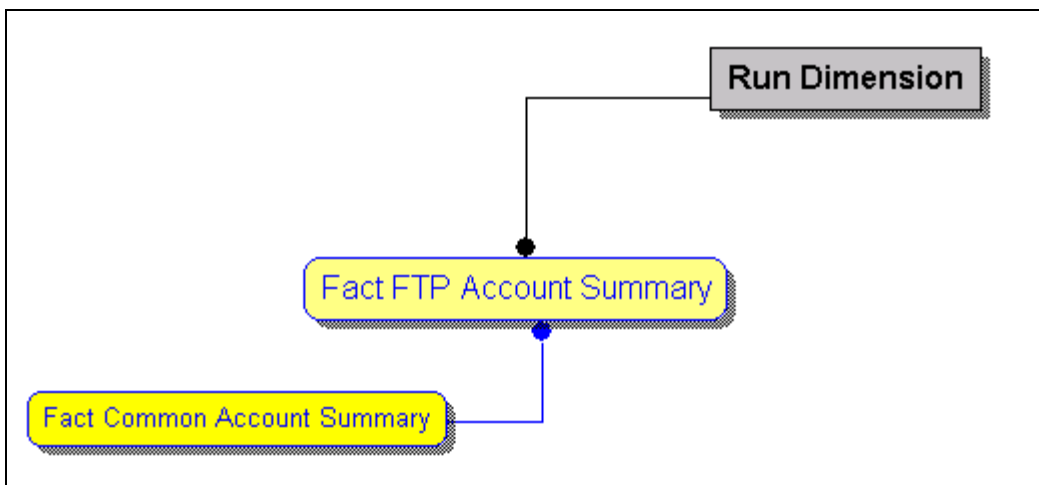
Figure 11: Overlapping Campaign
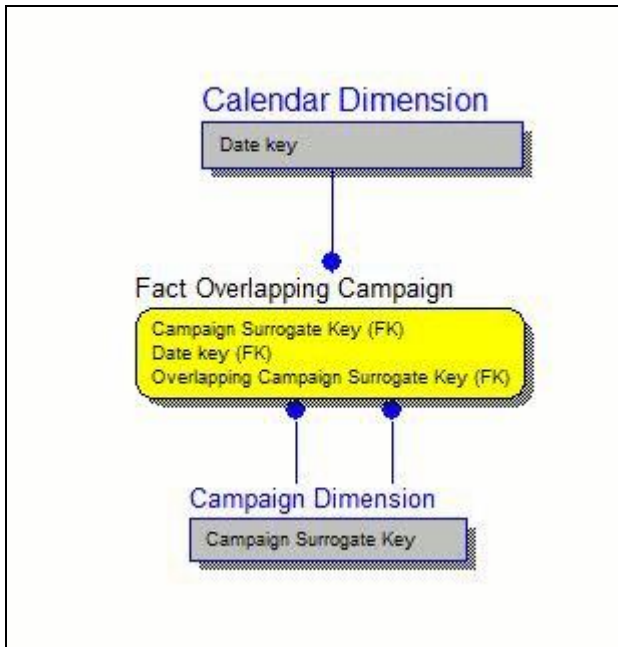
Figure 12: PFT Account Summary
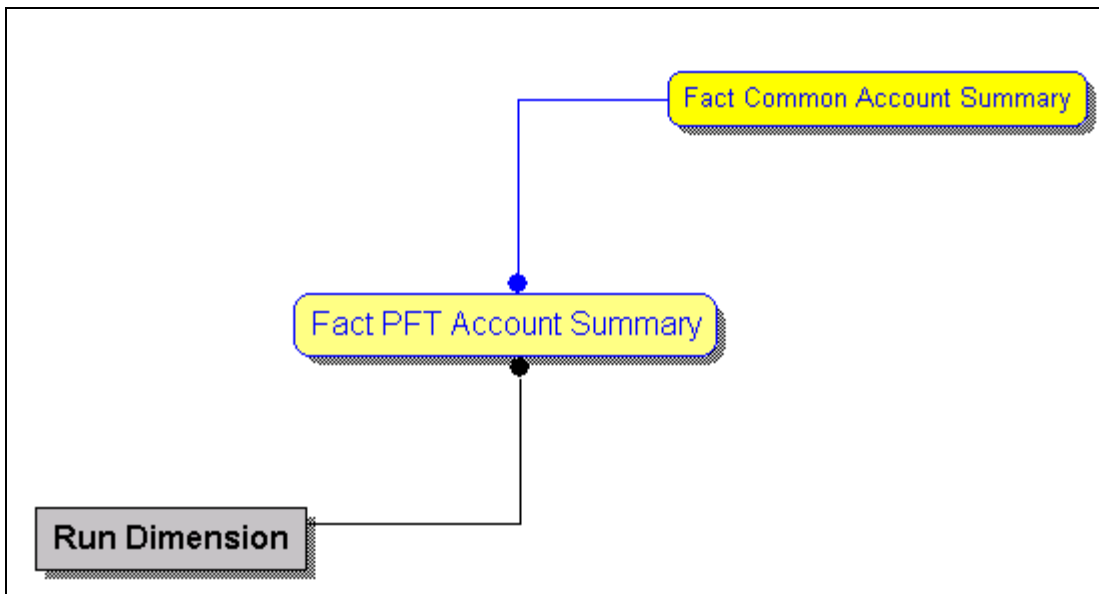
Figure 13: Product Feature
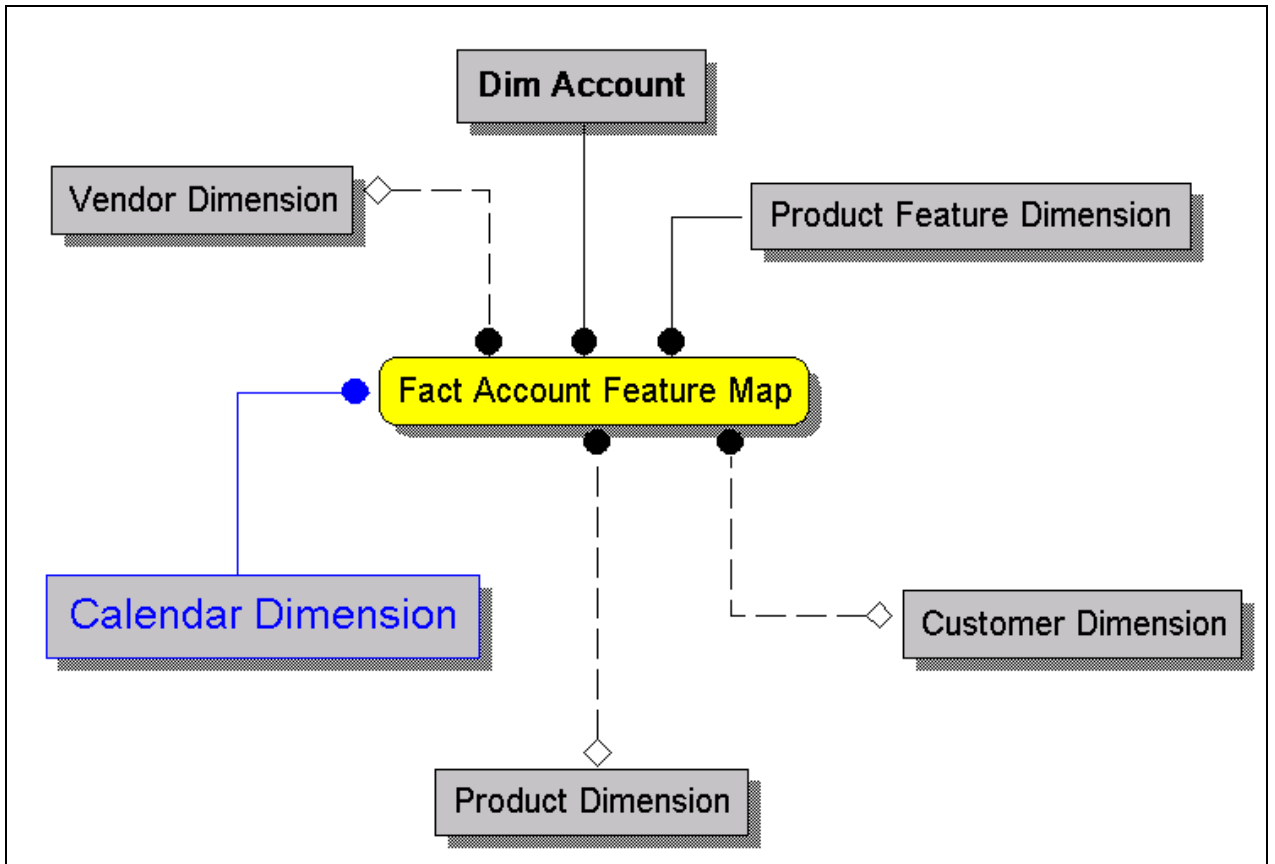
Figure 14: Transaction Channel
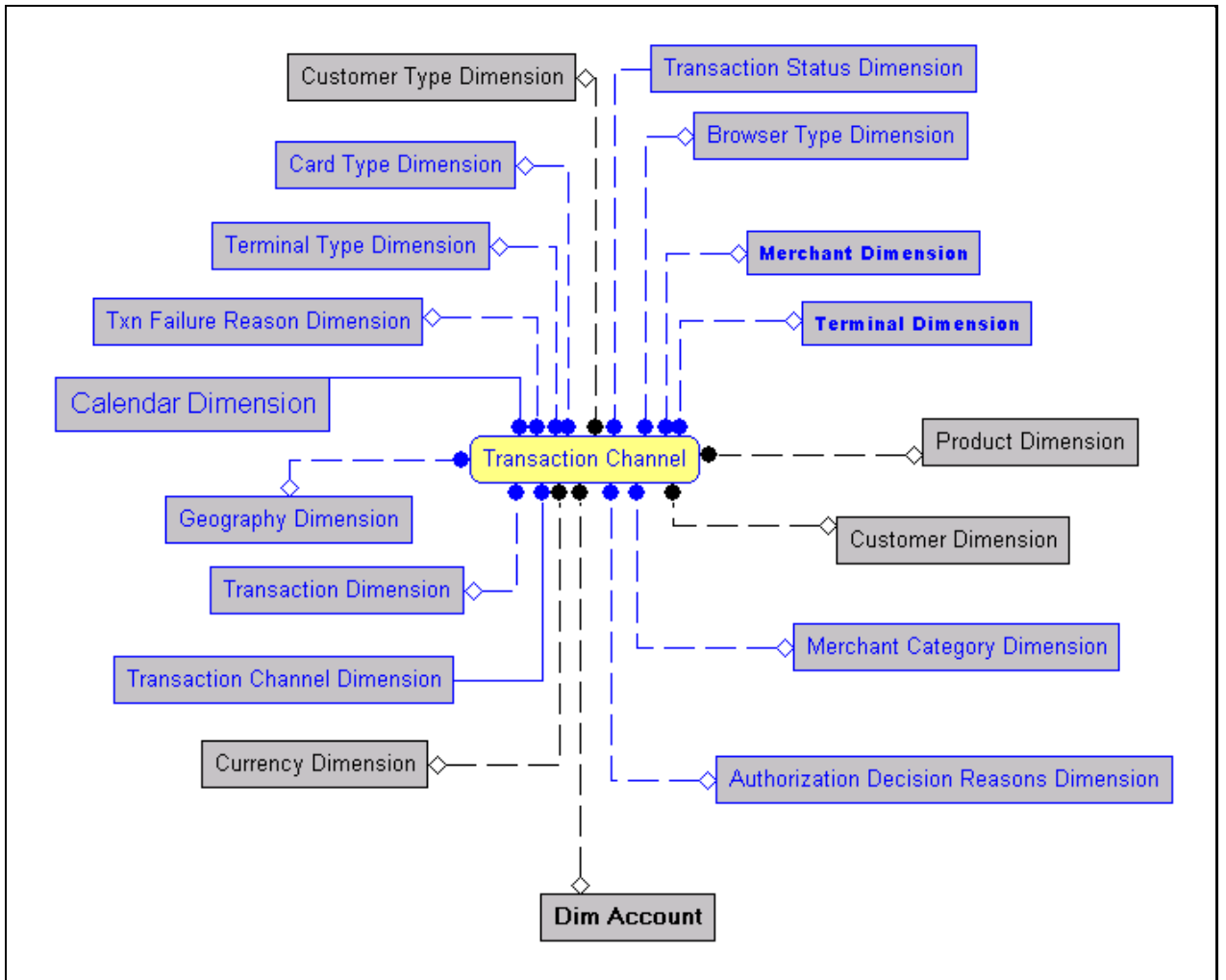
Figure 15: Fact Account Profitability
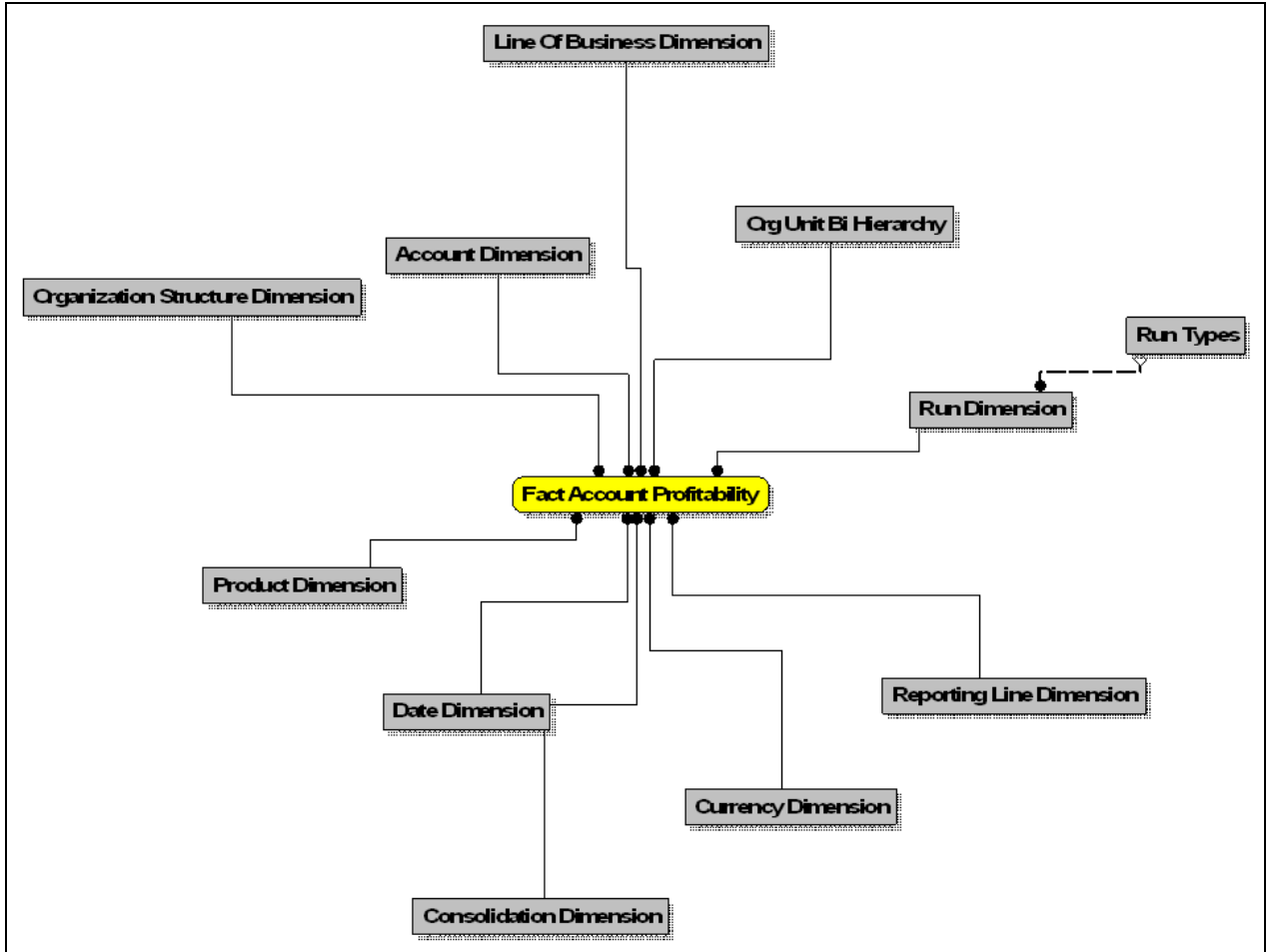
Figure 16: Fact Common Account Summary
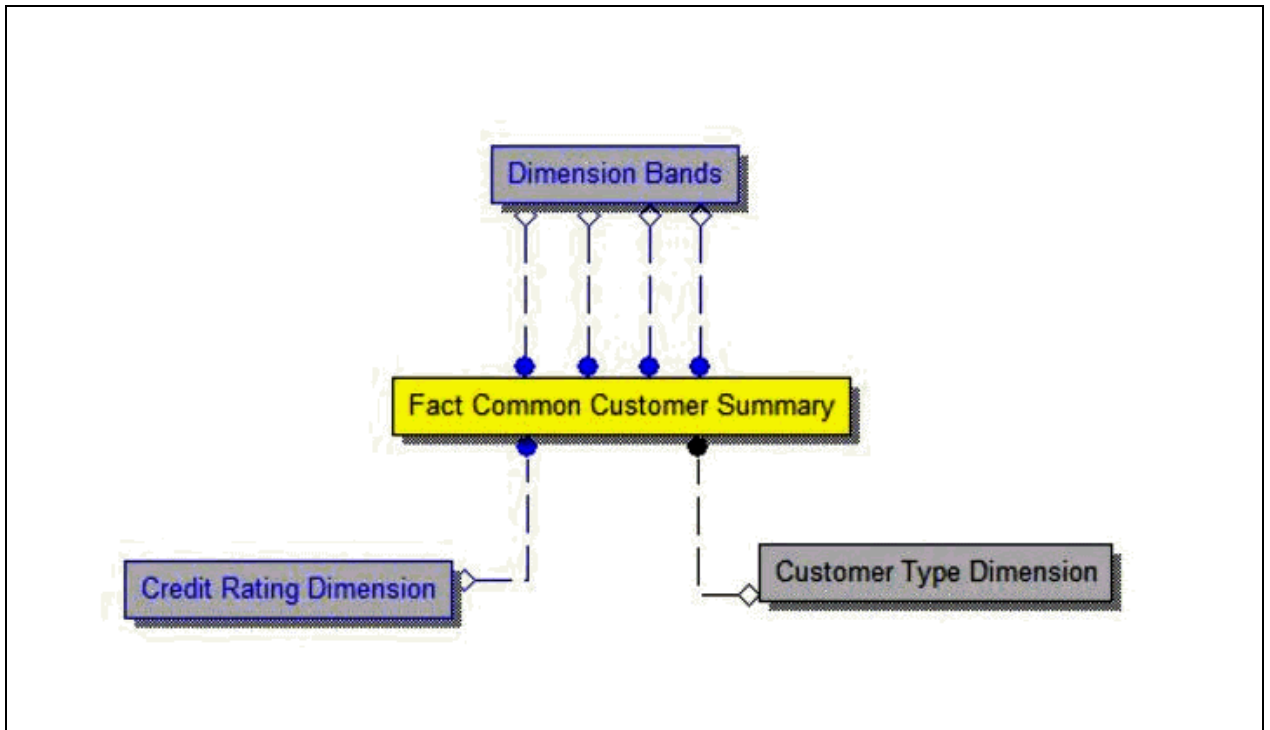
Figure 17: Fact Common Customer Summary
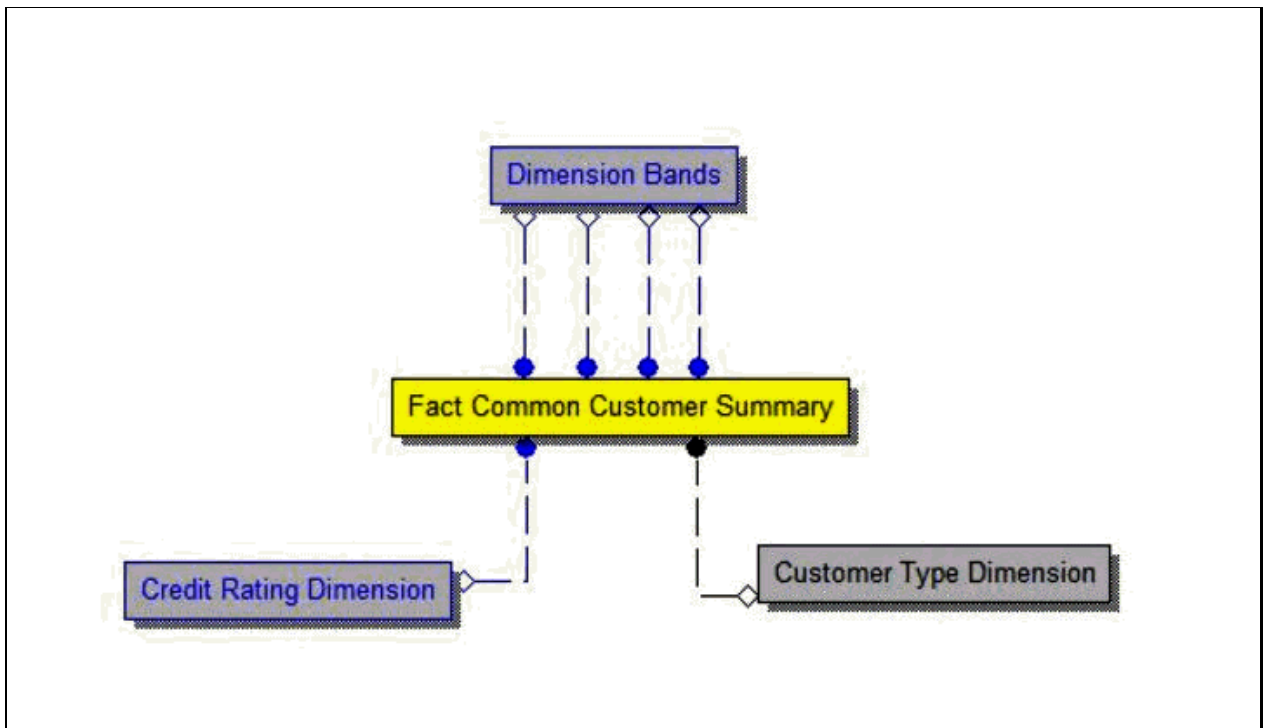
Figure 18: Fact CRM Account Summary
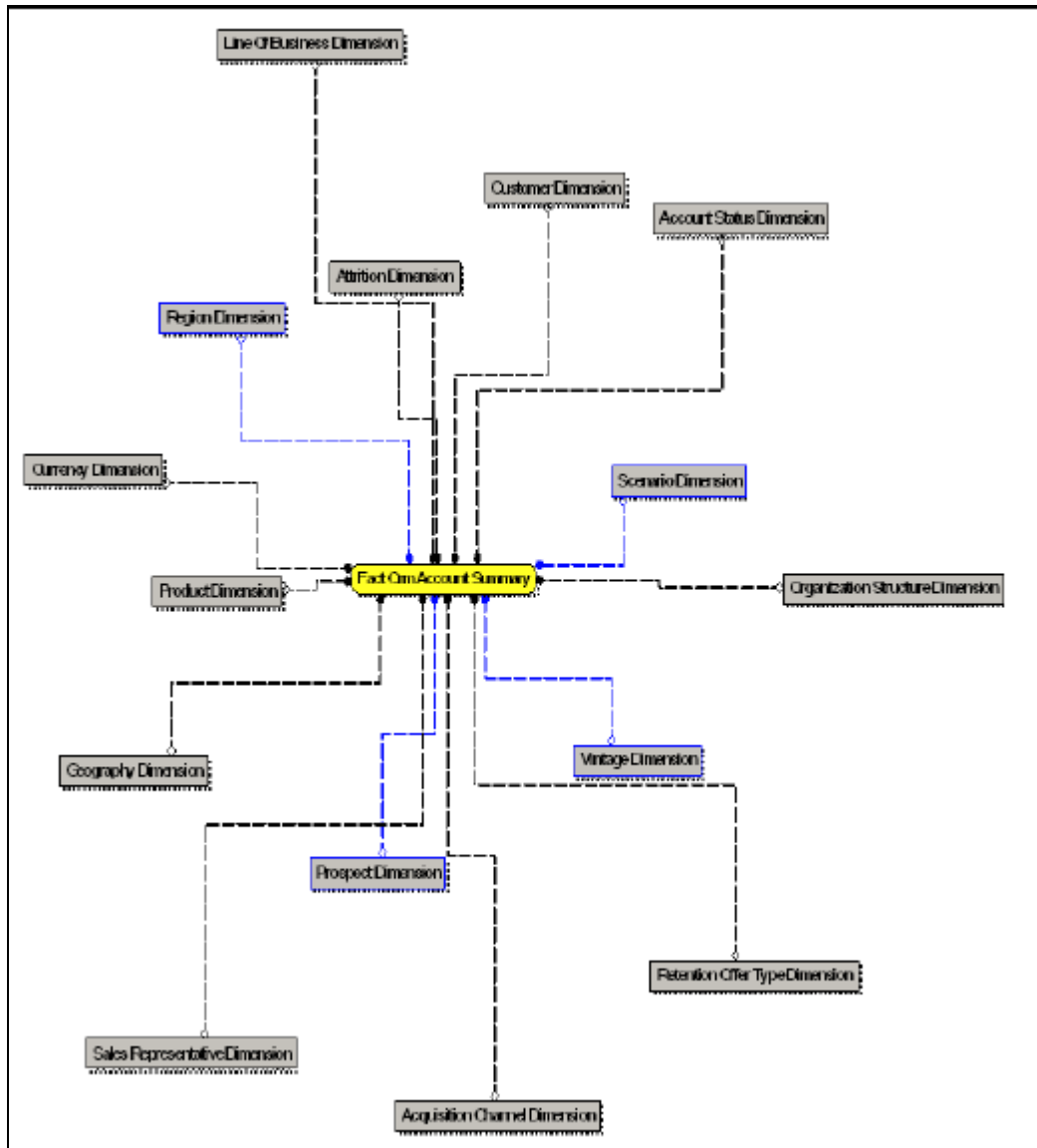
4   Dimension Loading Process

This chapter discusses the following topics:

- Dimension Tables Population
- Overview of SCD Process
- Executing the SCD Component

4.1   Dimension Tables Population

OFS RCA solution 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:
  http://www.oracle.com/technetwork/middleware/data-integrator/overview/odi-bestpractices-datawarehouse-whi-129686.pdf
- Oracle® Warehouse Builder Data Modeling, ETL, and Data Quality Guide at:
  http://docs.oracle.com/cd/E14072_01/owb.112/e10935.pdf

Additional online sources include:

- http://en.wikipedia.org/wiki/Slowly_changing_dimension
- http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/10g/r2/owb/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=2
  04800027&pgno=1
- http://www.informationweek.com/news/software/bi/showArticle.jhtml?articleID=5 9301280

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

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 SCD executable should be present under <installation home>ficdb/bin. The file name is scd.
- The user executing the SCD component should have execute rights on the file mentioned as a prerequisite in point 2.
- 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 complex.
- 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 database 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 5: SYS_TBL_MASTER Dimensions

Column Name	Data Type	Column Description
MAP_REF_NUM	NUMBER(3) NOT NULL	The Mapping Reference Number for this unique mapping of a Source to a Dimension Table.
TBL_NM	VARCHAR2(30) NOT NULL	Dimension Table Name
STG_TBL_NM	VARCHAR2(30) NOT NULL	Staging Table Name
SRC_PRTY	NUMBER(2) NULL	Priority of the Source when multiple sources are mapped to the same target.
SRC_PROC_SEQ	NUMBER(2) NOT NULL	The sequence in which the various sources for the DIMENSION will be taken up for processing.
SRC_TYP	VARCHAR2(30) NULL	The type of the Source for a Dimension, that is, Transaction Or Master Source.
DT_OFFSET	NUMBER(2) NULL	The offset for calculating the Start Date based on the Functional Requirements Document (FRD).
SRC_KEY	NUMBER(3) NULL	
Sample Data: This is the row put in by the solution installer for the Line of Business dimension.		
MAP_REF_NUM	6	
TBL_NM	DIM_LOB	
STG_TBL_NM	STG_LOB_MASTER	
SRC_PRTY		
SRC_PROC_SEQ	23	
SRC_TYP	MASTER	
DT_OFFSET	0	

Column Name	Data Type	Column Description
SRC_KEY		

> **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 6: SYS_STG_JOIN_MASTER Dimensions

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

Column Name	Data Type	Column Description
COL_FORMAT	VARCHAR2(15) NULL	The possible values for column type (the COL_TYPE column) in SYS_STG_JOIN_MASTER are: • PK: Primary Dimension Value (maybe multiple for a given "Mapping Reference Number") • SK: Surrogate Key • DA: Dimensional Attribute (maybe multiple for a given "Mapping Reference Number") • SD: Start Date • ED: End Date • LRI: Latest Record Indicator (Current Flag) • CSK: Current Surrogate Key • PSK: Previous Surrogate Key • SS: Source Key • LUD: Last Updated Date / Time • LUB: Last Updated By
Sample Data: This is the row put in by the Solution installer for the Line of Business Dimension.		
MAP_REF_NUM	6	
COL_NM	V_LOB_CODE	
COL_TYP	PK	
STG_COL_NM	V_LOB_CODE	
SCD_TYP_ID		
PRTY_LOOKUP_REQD_FLG	N	
COL_DATATYPE	VARCHAR	
COL_FORMAT	61	

NOTE	For any new dimension added, the column details will have to be inserted 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 99999999999999999999999999
increment by 1
```

4.3 Executing the SCD Component

To execute the SCD component from the OFSAAI ICC framework create a batch as per the following steps:

NOTE	For more comprehensive coverage of configuration and execution of a batch, see OFS Analytical Applications Infrastructure User Guide.

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

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

3. Click **Save**.

4. Select the Batch you created in the earlier step by clicking the check box in the Batch Name container.

5. Click **New Task** ('+' symbol in Task Details container).

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

7. Select **Run Executable** from the Component ID list.

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

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

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

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

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

     Example: scd, 61

   - **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:

- -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.

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

> **NOTE**   Seeded batch <Infodom>_FTP_PFT_Reqd_Dim is provided FTP or PFT application is installed which can be executed for populating FTP/PFT required dimensions.

4.3.1   Checking the Execution Status

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

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

> **NOTE**   For more comprehensive coverage of configuration and execution of a batch, see OFS Analytical Applications Infrastructure User Guide.

The status messages in Batch Monitor are:

- N: Not Started

- O: On Going

- F: Failure

- S: Success

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

The file name will have the batch execution ID.

Sample:

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

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

The file name will have the batch execution ID.

Sample:

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

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

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 the Quarter 1, 2, 3, and 4 data get rolled up to, say 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 non-additive rollups wherein a node formula are used to specify how to rollup the child member balances (for example, 3 months rolling average).

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:

- Database function - FN_DIM_DATES

- Database procedure - PROC_DIM_DATES_POPULATION, which is called by the function FN_DIM_DATES.

5.1.1   Prerequisites

All the post-installation steps mentioned in the OFS AAI Installation and Configuration Guide and the solution installation manual of Oracle Financial Services Retail Customer Analytics have to be completed successfully.

Application User must be mapped to a role that has 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 OFS Analytical Applications Infrastructure User Guide).

- Iccserver

- Router

- AM Server

- Messageserver

You should create batches to execute the function. For more details see the Executing the Time Dimension Population Transformation section.

5.1.2   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 earlier tables, see *Oracle Financial Services Analytical Applications Data Model Data Dictionary* or the *Erwin Data Model*.

5.2 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:

> **NOTE**   For more comprehensive coverage of configuration and execution of a batch, see <u>OFS Analytical Applications Infrastructure User Guide</u>.

1. From the Home menu, select Operations, then select Batch Maintenance.

2. Click New Batch ('+' symbol in Batch Name container) and enter the Batch Name and description.

3. Click **Save**.

4. Select the Batch you have created in the earlier step by clicking on the checkbox in the Batch Name container.

5. Click New Task ('+' symbol in Task Details container).

6. Enter the Task ID and Description.

7. Select Transform Data, from the components list.

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

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

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

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

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

   - **Parameter List**: Start Date, End Date

     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'.

     Sample parameter for this task is '20081131','20091231'.

9. You can execute the batch in two ways:

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

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

- The function can also be executed directly on the database through SQLPLUS. Details are:
  - Function Name : FN_DIM_DATES
  - Parameters : p_batch_run_id, p_as_of_date, P_ST_DT, P_ED_DT Sample parameter values : 'Batch1','20091231', '20081131','20091231'

5.2.1   Checking the Execution Status

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

> **NOTE** For more comprehensive coverage of configuration and execution of a batch, see OFS 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 following directory $FIC_DB_HOME/log/date. The file name will have the batch execution ID.

The database-level operations log can be accessed by querying the FSI_MESSAGE_LOG table. The batch run ID column can be filtered for identifying the relevant log.

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

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.

6.1 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.

6.1.1 FSI_MERGE_SETUP_DETAILS

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

Table 7: Columns in FSI_MERGE_SETUP_DETAILS

Column Name	Data Type	Column Description
MERGE_CODE	VARCHAR2 (50 CHAR)	This is the role of the party, that is, customer, issuer, and so on.
SOURCE_TABLE	VARCHAR2 (30 CHAR)	This is the source table for 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 on the source column for the Customer dimension population.
AGGREGATE_FUNCTION	VARCHAR2 (30 CHAR)	This is used for aggregating data for some source columns.
Sample Data:		
MERGE_CODE	MI	
TABLE SOURCE	DIM_PARTY	

Column Name	Data Type	Column Description
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		

6.1.2 FSI_MERGE_SETUP_MASTER

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

Table 8: Columns in FSI_MERGE_SETUP_MASTER

Column Name	Data Type	Column Description
MERGE_CODE	VARCHAR2 (50 CHAR)	This is the role of the party that is, customer, issuer, and so on.
SOURCE_TABLES	VARCHAR2 (4000 CHAR)	This is the list of source tables for 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.
Sampe Data:		
MERGE_CODE	MI	
SOURCE_TABLES	DIM_PARTY	
TARGET_TABLE	DIM_CUSTOMER	
ANSI_JOIN		
FILTER_CONDITION		

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

| **NOTE** | A seeded batch<INFODOM> aCRM_CommonTasks - Task3 is provided so that the user can just modify the parameters and execute the batch. |

6.2.1 Checking the Execution Status

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

The status messages in Batch Monitor are:

- N: Not Started

- O: On Going

- F: Failure

- S: Success

The Event Log window in Batch Monitor 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

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

7.1   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.

7.1.1   Type 1 SCDs - Overwriting

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

Table 9: Type 1 SCDs - Overwriting

Name	Description	Value
N_PRODUCT_SKEY	The surrogate key column is a unique key for each record in the dimension table.	1
V_PRODUCT_ NAME	Product Name	PL
D_START_DATE	Indicates the date from which this product record is valid.	5/31/2010
D_END_DATE	Indicates the date till which this product record is valid.	12/31/9999
F_LATEST_RECORD_INDICATOR	Value 'Y' indicates this is the latest record in the dimension table for this product and 'N' indicates it is not.	N

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

Table 10: Type 1 SCDs - Overwriting1

Name	Description	Value
N_PRODUCT_SKEY	The surrogate key column which is a unique key for each record in the dimension table.	1
V_PRODUCT_ NAME	Product Name	Personal Loan
D_START_DATE	Indicates the date from which this product record is valid.	6/30/2010
D_END_DATE	Indicates the date till which this product record is valid.	12/31/9999

Name	Description	Value
F_LATEST_RECORD_INDICATOR	Value 'Y' indicates this is the latest record in the dimension table for this product and 'N' indicates it is not.	Y

7.1.2   Type 2 SCDs - Creating another Dimension Record

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

Table 11: Type 2 SCDs - Creating another dimension record

Name	Description	Value
N_PRODUCT_SKEY	The surrogate key column which is a unique key for each record in the dimension table.	1
N_PRODUCT_SKEY	The surrogate key column which is a unique key for each record in the dimension table.	1
V_PRODUCT_ NAME	Product Name	PL
V_PRODUCT_ NAME	Product Name	Personal Loan
D_START_DATE	Indicates the date from which this product record is valid.	5/31/2010
D_START_DATE	Indicates the date from which this product record is valid.	6/30/2010
D_END_DATE	Indicates the date till which this product record is valid.	12/31/9999
D_END_DATE	Indicates the date till which this product record is valid.	12/31/9999
F_LATEST_RECORD_INDICATOR	Value 'Y' indicates this is the latest record in the dimension table for this product and 'N' indicates it is not.	N
F_LATEST_RECORD_INDICATOR	Value 'Y' indicates this is the latest record in the dimension table for this product and 'N' indicates it is not.	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'.

7.1.3   Type 3 SCDs - Creating a Current Value Field

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

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

For more information on SCDs, see:

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

- OracleR Warehouse Builder Data Modeling, ETL, and Data Quality Guide at Warehouse Builder Data Modeling, ETL, and Data Quality Guide

Additional online sources include:

- https://en.wikipedia.org/wiki/Slowly_changing_dimension

- https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/10g/r2/owb/owb10gr2_gs/owb/lesson3/slowlychangingdimensions.htm

- 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.

7.2   Prerequisites

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

- The user executing the SCD component should have execute rights on the file mentioned as a prerequisite in point 2.

- 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 complex.

  - 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 database sequence to populate into surrogate key columns of dimension tables.

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.

7.2.1   Tables Used by the SCD Component

The database tables used by the SCD component are:

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

Table 12: SYS_TBL_MASTER Dimensions

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.

Column Name	Data Type	Column Description
TBL_NM	VARCHAR2(30) NOT NULL	Dimension Table Name
STG_TBL_NM	VARCHAR2(30) NOT NULL	Staging Table Name
SRC_PRTY	NUMBER(2) NULL	Priority of the Source when multiple sources are mapped to the same target.
SRC_PROC_SEQ	NUMBER(2) NOT NULL	The sequence in which the various sources for the DIMENSION will be taken up for processing.
SRC_TYP	VARCHAR2(30) NULL	The type of the Source for a Dimension, that is, Transaction Or Master Source.
DT_OFFSET	NUMBER(2) NULL	The offset for calculating the Start Date based on the Functional Requirements Document (FRD).
SRC_KEY	NUMBER(3) NULL	
Sample Data: This is the row put in by the solution installer for the Line of Business dimension.		
MAP_REF_NUM	6	
TBL_NM DIM_LOB		
STG_TBL_NM	STG_LOB_MASTER	
SRC_PRTY		
SRC_PROC_SEQ	23	
SRC_TYP MASTER		
DT_OFFSET		
SRC_KEY		

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 13: 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.

Column Name	Data Type	Column Description
MAP_REF_NUM	NUMBER(3) NOT NULL	The Mapping Reference Number for this unique mapping of a Source to a Dimension Table.
COL_NM	VARCHAR2(30) NOT NULL	Name of the column in the Dimension Table.
COL_TYP	VARCHAR2(30) NOT NULL	Type of column. The possible values are given in the following section.
STG_COL_NM	VARCHAR2(60) NULL	Name of the column in the Staging Table.
SCD_TYP_ID	NUMBER(3) NULL	SCD type for the column.
PRTY_LOOKUP_REQD_FLG	CHAR(1) NULL	Column to determine whether Lookup is required for Priority of Source against the Source Key Column or not.
COL_DATATYPE	VARCHAR2(15) NULL	The list of possible values is VARCHAR, DATE, NUMBER based on the underlying column data type.
COL_FORMAT	VARCHAR2(15) NULL	The possible values for column type (the COL_TYPE column) in SYS_STG_JOIN_MASTER are: • PK: Primary Dimension Value (maybe multiple for a given "Mapping Reference Number") • SK: Surrogate Key • DA: Dimensional Attribute (maybe multiple for a given "Mapping Reference Number") • SD: Start Date • ED: End Date • LRI: Latest Record Indicator (Current Flag) • CSK: Current Surrogate Key • PSK: Previous Surrogate Key • SS: Source Key • LUD: Last Updated Date / Time • LUB: Last Updated By
Sample Data: This is the row put in by the solution installer for the Line of Business dimension.		
MAP_REF_NUM	6	
COL_NM	V_LOB_CODE	

Column Name	Data Type	Column Description
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
```

7.3 Executing the SCD Component

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

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

> **NOTE** A seeded batch <Infodom>_SCD_Retail_Perf_Analy_Dim is provided which has all the required dimensions as different tasks that are part of SCD.

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

> **NOTE** For more comprehensive coverage of configuration and execution of a batch, see OFS Analytical Applications Infrastructure User Guide.

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

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

     Example

     scd, 61 (see the following sections for details)

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

7. Click **Save**. The Task definition is saved for the selected Batch.

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

7.3.1   Checking the Execution Status

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

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

> **NOTE** For more comprehensive coverage, see OFS Analytical Applications Infrastructure User Guide.

The status messages in Batch Monitor are:

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

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

The file name will have the batch execution ID.

Sample:

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

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

The file name will have the batch execution ID.

Sample:

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

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

7.4   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.

7.4.1   DIM_ACCOUNT SCD

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

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

7.4.2   LOAD DIM TABLES THROUGH SCD

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

7.4.3   Improve SCD Performance

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

- merge_hint
- select_hint
- session_enable_statement
- session_disable_statement

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

```
ALTER TABLE SYS_TBL_MASTER ADD MERGE_HINT VARCHAR2(255)

/

ALTER TABLE SYS_TBL_MASTER ADD SELECT_HINT VARCHAR2(255)

/

ALTER TABLE SYS_TBL_MASTER ADD SESSION_ENABLE_STATEMENT VARCHAR2(255)

/

ALTER TABLE SYS_TBL_MASTER ADD SESSION_DISABLE_STATEMENT VARCHAR2(255)

/
```

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

Table 14: MERGE_HINT and SESSION_ENABLE_STATEMENT in SYS_TBL_MASTER

Table Name	Stage Table Name	Merge Hint	Session Enable Statement
DIM_ACCOUNT	STG_LOAN_CONTRACTS_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 15: 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.

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

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

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

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

8   Exchange Rate History Population

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 by means of the Table to Table Transformation process. Following is the seeded Table-to-Table definition that loads data into Exchange Rate History:

Table 16: Table to Table defintion

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

8.1   Exchange Rate History Population

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen), a seeded batch, <INFODOM>_aCRM_CommonTasks - Task4 has to be executed for the required date.

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 OFS 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: The 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.

8.2 Execution of Currency Exchange Rates Population T2T

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

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

- Metadata Browser

- Common Account Summary

8.2.1 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 19: 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**
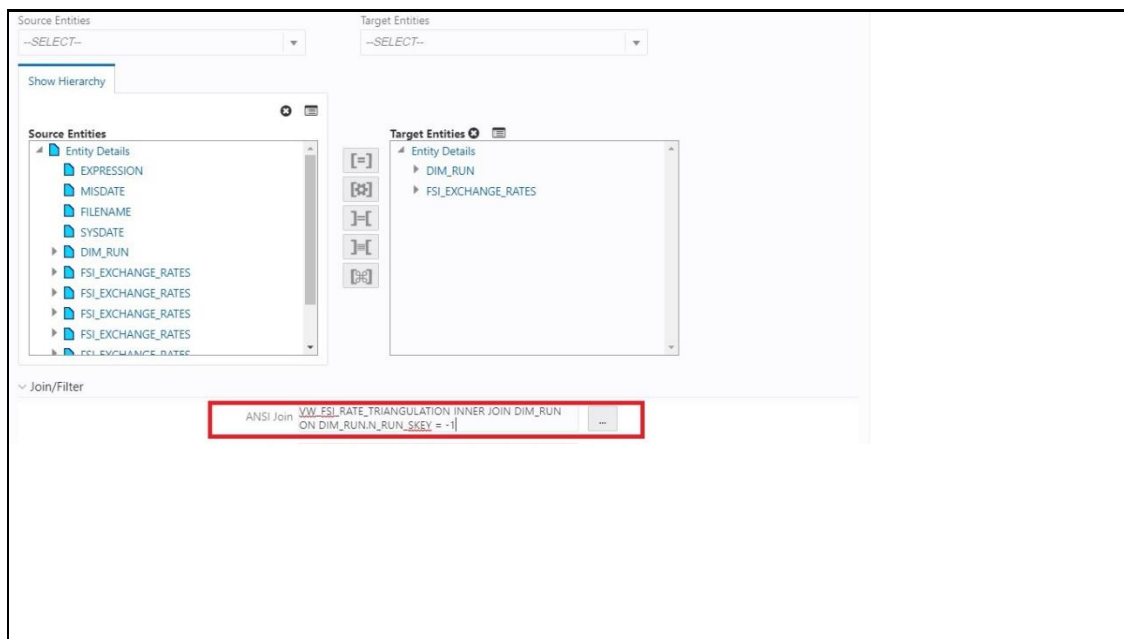
3. If the RHS part of the equation is **$RUNSK** then RHS should be changed to **-1**.

Figure 20: Mapping Details screen
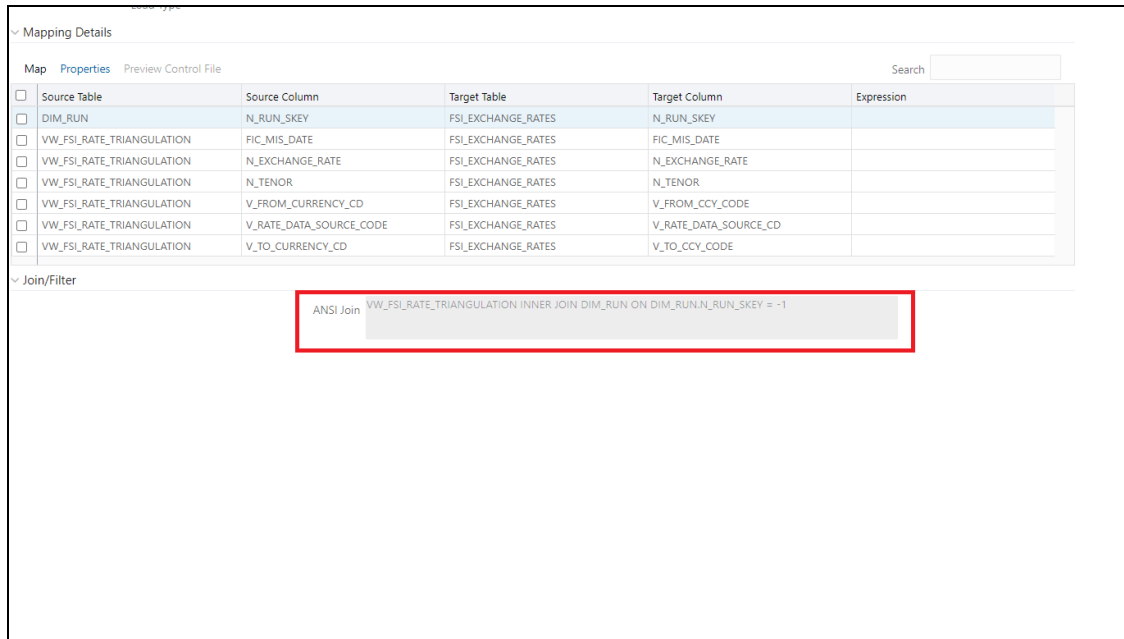
4. Click Map to modify the ANSIJOIN parameter.

Figure 21: 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**.

8.2.2   Currency Execution Rates - Batch Execution

A seeded batch, <Infodom>_POP_EXCHANGE_RATES 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 any need for currency conversion in T2T transactions, a Default value has to be provided.

9. Execute the batch created in the preceding steps.

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

<INFODOM>_FN_RATEVALIDATION is invoked for the exchange rate history. Once data is loaded into fsi_exchange_rate_hist table, run the batch <INFODOM>_FN_RATEVALIDATION.

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.

9.1   Overview of Account Summary Tables

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

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

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

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

- FCT_CRM_ACCOUNT_SUMMARY: This table has the measures used by all the Customer Insight applications. Yet, few other Account Summary tables have been designed to store Enterprise Risk Management (ERM) 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.

The Account Summary tables are part of the data model, but there are no seeded T2T definitions available to populate these tables. T2T processes must be custom configured to populate these tables to use measures defined on these tables for reporting.

Data Flow

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

Figure 22: Account Summary Data Flow
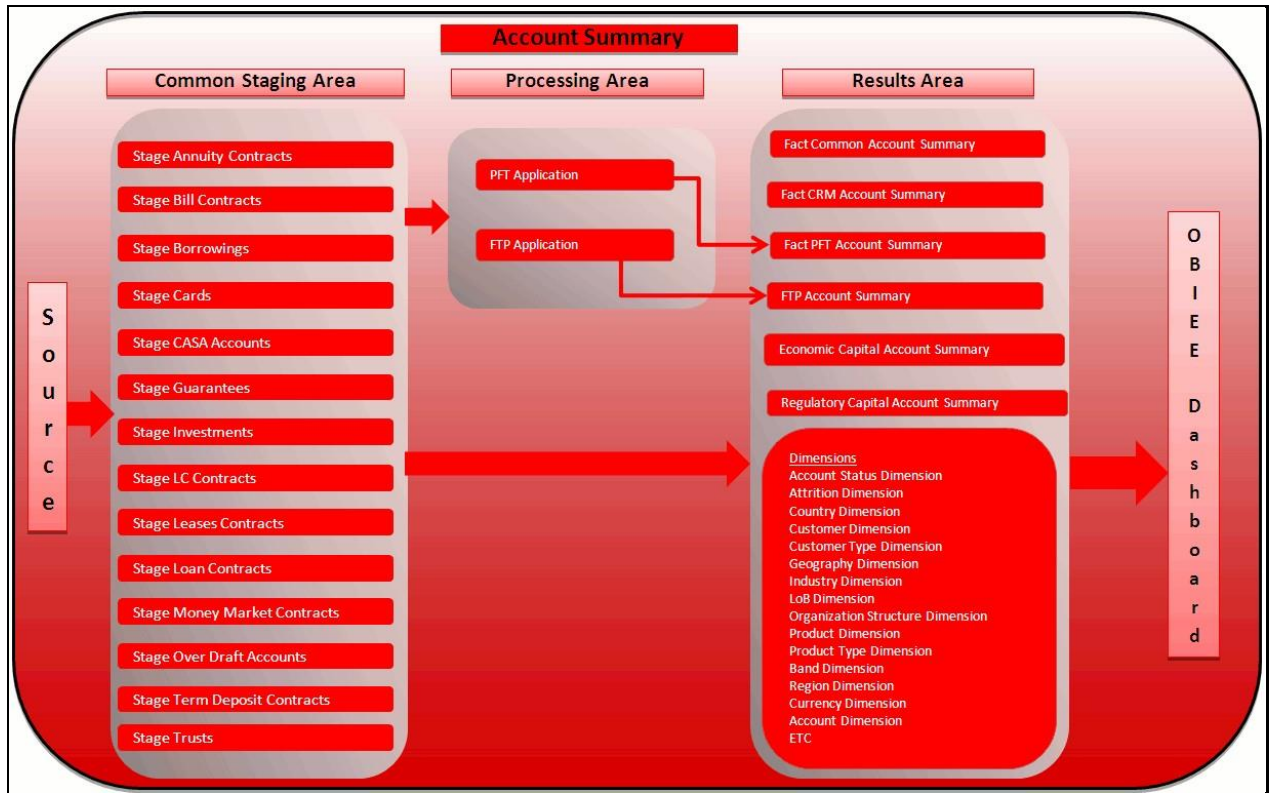
9.2 Overview of Account Summary Population

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

Following are the lists for the same:

Common Account Summary:

Table 17: Common Account Summary Definitions

S L	Source Table	T2T Definition Name	Destination Table No
1	STG_ANNUITY_CONTRACTS	T2T_STG_ANNUITY_CONTRACTS_CAS	FCT_COMMON_ACCOUNT_SUMMARY
2	STG_BILLS_CONTRACTS	T2T_STG_BILLS_CAS	FCT_COMMON_ACCOUNT_SUMMARY
3	STG_BORROWINGS	T2T_STG_BORROWINGS_CAS	FCT_COMMON_ACCOUNT_SUMMARY
4	STG_CARDS	T2T_STG_CARDS_CAS	FCT_COMMON_ACCOUNT_SUMMARY
5	STG_CASA	T2T_STG_CASA_CAS	FCT_COMMON_ACCOUNT_SUMMARY

S L	Source Table	T2T Definition Name	Destination Table No
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

CRM Account Summary:

Table 18: CRM Account Summary Definitions

S L	Source Table	T2T Definition Name	Destination Table No
1	STG_ANNUITY_CONTRACTS	T2T_STG_CRMAS_ANNUITY_CONTRACTS	FCT_CRM_ACCOUNT_SUMMARY
2	STG_BILLS_CONTR ACTS	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
5	STG_CASA	T2T_STG_CRMAS_CASA	FCT_CRM_ACCOUNT_SUMMARY

S L	Source Table	T2T Definition Name	Destination Table No
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	CT_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_CRM_ACCOUNT_SUMMARY

Tables FTP Account Summary and PFT Account Summary must be loaded directly if PFT and FTP applications do not already co-exist with Retail Customer Analytics.

> **NOTE** The Currency Exchange Rate History table has to be populated before loading the Account Summary tables.

9.3 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 OFS 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. (See Dimension Loading Process and Time Dimension Population chapters).

9.4   Fact Common Account Summary

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

The following 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.

9.5   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_CAMPAIGN

- 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 more information, see the Dimension Tables Population section for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information on populating account dimensions, see the Account Dimension Population chapter.

For more information, see the Time Dimension Population chapter for details on populating the DIM_DATES dimension table. See Download Specification for identifying fields required in Stage Customer Master and Stage Customer Details for Customer Insight Application(s).

For more information on the dimensions, see the *ERwin Datamodel*.

9.6 Executing the Account Summary Population T2T

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

9.6.1 Fact Common Account Summary

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen)a seeded batch, <Infodom>_aCRM_Comm_Acc_Summ 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 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. 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.

9.7   Fact CRM Account Summary

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen)a seeded batch, <Infodom>_aCRM_CRM_Acc_Summ 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 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. 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.

9.7.1   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 OFS 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$

9.8   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   Customer Summary Population

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

Fact Common Customer Summary table stores 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 23: Fact Common Customer Summary Data Flow
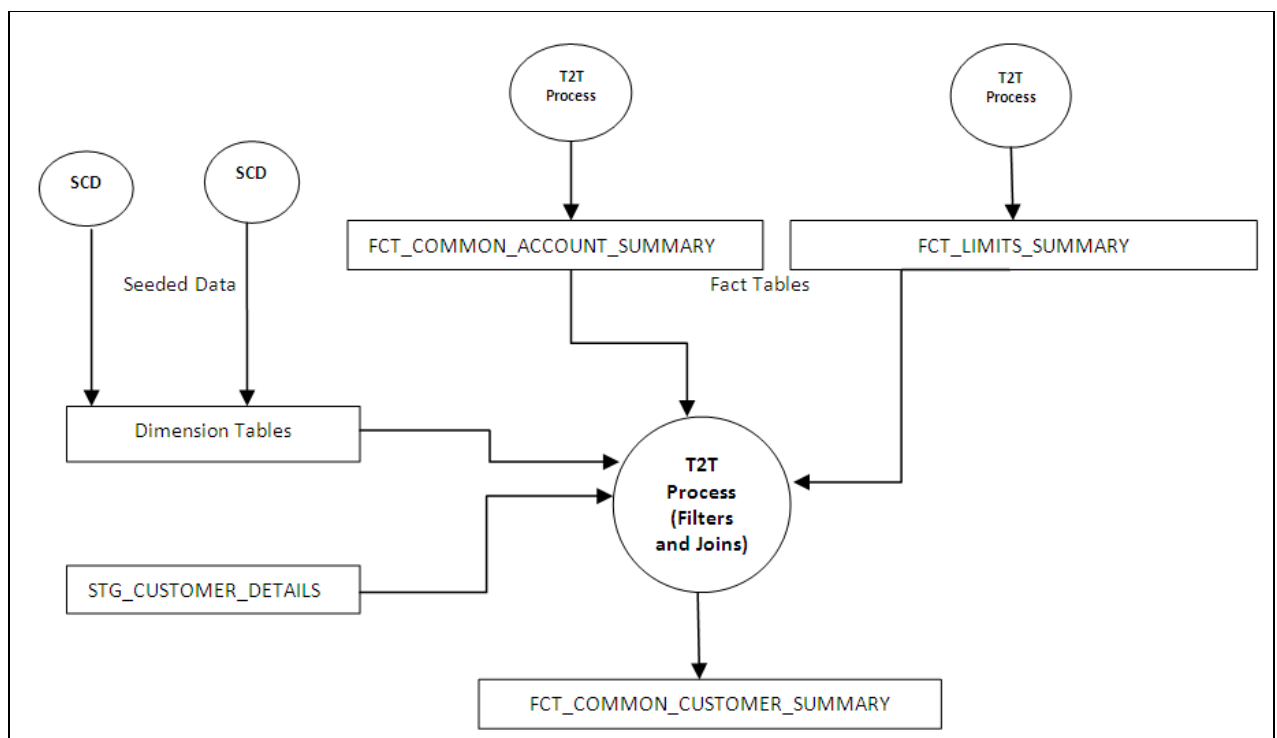
10.1   Prerequisites

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_SUMMAY and FCT_LIMITS_SUMMARY are loaded from their respective T2T processes.

For more information on SCDs, see the [Dimension Loading Process](#) chapter.

10.2 Executing the Customer Summary Population T2T

Fact Common Customer Summary T2T can be executed by executing task present in the seeded batch <INFODOM>_aCRM_CommCust_Appln. The following steps will help you to execute the batch:

1. Navigate to the Batch Execution screen.

2. Select the seeded batch <INFODOM>_aCRM_CommCust_Appln where INFODOM is the information domain where 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.

10.2.1 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.

11 Fact Data Population

This chapter explains all the fact tables which within describe about 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.

11.1 Fact CRM Customer Summary

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

Load Data into Fact CRM Customer Summary

Customer balances in the Fact CRM Customer Summary entity are derived from the account summary. 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 19: Fact CRM Customer Summary Definitions

T2T Definition Name	Source Table(s)	Destination Table
T2T_FCT_CRM_CUSTOMER_SUMMARY	STG_CUSTOMER_MASTER STG_CUSTOMER_DETAILS FCT_COMMON_ACCOUNT_SUMMARY FCT_CRM_ACCOUNT_SUMMARY	FCT_CRM_CUSTOMER_SUMMARY

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

11.1.1 Prerequisites

Fact Common Customer Summary entity needs to be populated before executing the Fact CRM Customer Summary T2T. See Fact Common Account Summary chapter for details related to Fact Common Customer Summary T2T.

The following tables that are used in the population of Fact CRM Customer Summary should 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 more information, see the Dimension Tables Population section in the Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information, see the Time Dimension Population chapter for details on populating the DIM_DATES dimension table. See Download Specification for identifying fields required in Stage Customer Master and Stage Customer Details for Customer Insight Application(s).

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

11.1.2 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, <Infodom>_aCRM_CRM_Cust_Summ 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 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. 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.

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. The following tables can be queried for errors:

FCT_CRM_CUSTOMER_SUMMARY

11.2 Update Fact CRM Customer Summary with Transaction Attributes

A seeded Data Transformation is provided with the installer which updates the entity Fact CRM Customer Summary with transaction attributes of the customer such as ATM usage, Branch usage, net usage, Point of Sale (POS) usage, Number of ATM transactions, transacted amount, and so on.

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

Table 20: Post Load Transformation Definition

DT Definition Name	Source Tables	Destination Table
FN_UPD_CRM_CUST_CHNL	FCT_TXN_CHANNEL	FCT_CRM_CUSTOMER_SUMMARY

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

Alternatively, you can create a new batch by following the steps:

1. Select the check box adjacent to the newly created 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 to the 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 FN_UPD_CRM_CUST_CHNL 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.

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

11.2.1 Checking the Execution Status

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

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.

11.3 Fact Account Feature Map

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

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

Table 21: Fact Account Feature Map T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_ACCOUNT_FEATURE_MAP	STG_ACCT_FEATURE_MAP	FCT_ACCOUNT_FEATURE_MAP

To view the detailed structure of the tables, see the Customer Insight Erwin Data Model.

11.3.1 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_CAMPAIGN

- DIM_CHANNEL

- STG_ACCT_FEATURE_MAP

For more information, see the Dimension Tables Population section under Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information on populating account dimensions, see the Account Dimension Population chapter.

For more information, see the Time Dimension Population chapter for details on populating the DIM_DATES dimension table. See Download Specification for identifying fields required in Stage Customer Master and Stage Customer Details for Customer Insight Application(s).

11.3.2   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, <Infodom>_aCRM_Account_Feature_Map, 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 and click **Save**.

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

    - Datastore Name: Select the appropriate name from the list.

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

    - Load Mode: Select Table to Table from the list.

    - Source Name: Select <T2T Source Name> from the list.

    - File Name: Select the T2T name "T2T_FCT_ACCOUNT_FEATURE_MAP" you want to process.

8. The data file name will be blank for any Table to Table Load mode. Default value refers to any parameter that has to be passed to T2T. This should be blank.

9. Execute the batch created in the preceding steps.

11.3.3 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_ACCOUNT_FEATURE_MAP$

11.4 Fact Customer to Customer Relationship

Fact Customer to Customer Relationship entity stores the relationship between the customers. Examples of relationships amongst customers could be Employer, Employee, Children, Parent, Spouse, and so on.

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

Table 22: Fact Customer to Customer Relationship T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_CUST_CUST_RELATION	STG_CUST_CUST_RELATIONSHIP	FCT_CUST_CUST_RELATIONSHIP

To view the detailed structure of the tables, see Customer Insight Erwin Data Model.

11.4.1 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 more information, see the Dimension Tables Population section under Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information, the Time Dimension Population chapter for details on populating the DIM_DATES dimension table. See the Download Specifications for identifying fields required in Stage Customer Master and Stage Customer Details for Customer Insight Application(s).

11.4.2   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, <Infodom>_aCRM_Customer_Customer_Reln - Task1, 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 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. 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.

11.4.3   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_CUST_CUST_RELATIONSHIP$

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

11.5   Fact Transaction Channel

Fact Transaction Channel entity stores the details of all transactions (successful and failed) done through any of the transaction channels offered by the Financial Institutions. This fact entity is loaded from multiple source staging tables.

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

Table 23: Fact Transaction Channel T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_TEL_FCT_TXN_CHANNEL	STG_SRC_TB_TXNS	FCT_TXN_CHANNEL
T2T_POS_FCT_TXN_CHANNEL	STG_SRC_POS_TXNS	
T2T_NET_FCT_TXN_CHANNEL	STG_SRC_NET_TXNS	
T2T_BRA_FCT_TXN_CHANNEL	STG_SRC_BRANCH_TXNS	
T2T_ATM_FCT_TXN_CHANNEL	STG_SRC_ATM_TXNS	

To view the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

11.5.1   Prerequisites

Following are the lists of tables used in the population of Fact Transaction Channel and these tables are required to be loaded before running the T2T:

- DIM_DATES
- DIM_TXN_CHANNEL
- DIM_ACCOUNT
- DIM_AUTH_DECISION_REASONS
- DIM_BANDS
- DIM_BROWSER_TYPE
- DIM_CARD_TYPE
- DIM_CURRENCY
- DIM_CUSTOMER
- DIM_CUSTOMER_TYPE
- DIM_GEOGRAPHY
- DIM_MERCHANT
- DIM_MERCHANT_CATEGORY
- DIM_PRODUCT
- DIM_TERMINAL
- DIM_TERMINAL_TYPE

- DIM_TRANSACTION
- DIM_TXN_FAILURE_REASON
- DIM_TXN_STATUS
- STG_SRC_ATM_TXNS
- STG_SRC_BRANCH_TXNS
- STG_SRC_NET_TXNS
- STG_SRC_POS_TXNS
- STG_SRC_TB_TXNS

For more information, see the Dimension Tables Population section under Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information on populating account dimensions, see the Account Dimension Population chapter.

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

11.5.2   Executing the Fact Transaction Channel Population T2Ts

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen), a seeded batch, <Infodom>_aCRM_Txn_Channel Task1 to Task5, 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 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. Default value refers to currency calculation. If there is any need for currency conversion in T2T transactions, a Default value has to be provided.

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

10. Steps 4 to 8 must be repeated for adding the remaining 4 T2Ts within the same batch definition.

11. Execute the batch created in the preceding steps.

11.5.3 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_TXN_CHANNEL$

11.6 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 24: Fact Application T2T Definition

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.

11.6.1 Prerequisites

Following are the lists of tables used in the population of 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_PROFESSION
- DIM_HOME_OWNERSHIP
- DIM_EDUCATION
- DIM_MARITAL_STATUS
- DIM_APPLICATION_REJECT_REASONS
- DIM_DEVIATION_REASONS
- DIM_SALES_REPRESENTATIVE
- DIM_CAMPAIGN
- DIM_ACCOUNT
- DIM_PROSPECT
- DIM_BANDS
- STG_APPLICATION

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, see the section Dimension Tables Population in Dimension Loading Process chapter.

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

11.6.2 Executing the Fact Application Population T2T

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

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

11.6.3 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 executionID. The following tables can be queried for errors:

FCT_APPLICATION$

11.7 Fact Campaign Details

Fact Campaign Details entity stores the information about the details of the campaign like expected 5-year NPV, targeted prospect contact status, days to contact, no of times contacted, and so on.

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

Table 25: Fact Campaign Details T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_CAMPAIGN_DETAILS	STG_CAMPAIGN_DETAILS	FCT_CAMPAIGN_DETAILS

To view the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

11.7.1   Prerequisites

Following are the lists of tables used in the population of Fact Campaign Details and these tables are required to be loaded before running the T2T:

- DIM_CAMPAIGN_CHANNEL
- DIM_CAMPAIGN
- DIM_CUSTOMER
- DIM_PROSPECT
- DIM_DATES
- DIM_PRODUCT
- DIM_OFFER
- DIM_TREATMENT
- DIM_WAVE
- DIM_VENDOR
- DIM_CONTACT
- DIM_REGION
- DIM_MKTG_PROGRAM
- STG_CAMPAIGN_DETAILS

For more information, see the Dimension Tables Population section under Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

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

11.7.2   Executing the Fact Application 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_CampaignAnalysis - Task1, 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 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_CAMPAIGN_DETAILS', you want to process.

8. The data file name will be blank for any Table to Table Load mode. Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, 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. Execute the batch created in the preceding steps.

11.7.3　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 executionID. The following tables can be queried for errors:

FCT_CAMPAIGN_DETAILS$

11.8　Fact Campaign Execution Summary

Fact Campaign Execution Summary entity is a summary table which stores fact information like mail base, no of campaign prospects contacted, cost incurred, a number of outputs from the campaign, expected 5-year NPV, and so on across dimensions like Campaign Region, Offer, Treatment, Product, Wave, Campaign, Vendor, Offer Channel, and so on.

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

Table 26: Fact Campaign Execution Summary T2T Definition

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_CAMP_EXEC_SUM MARY	FCT_CAMPAIGN_DETAILS	FCT_CAMPAIGN_EXEC_SUMMARY

To view the detailed structure of the earlier tables, see Erwin Data Model.

11.8.1 Prerequisites

Fact Campaign Details T2T needs to be executed before populating the Fact Campaign Execution Summary fact table.

Following are the lists of tables used in the population of Fact Campaign Execution Summary and these tables are required to be loaded before running the T2T:

- FCT_CAMPAIGN_DETAILS
- DIM_DATES

For more information, see the Fact Campaign Details section on populating campaign details.

For more information, see the Time Dimension Population chapter for details on populating the DIM_DATES dimension table.

11.9 Executing the Fact Application 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_CampaignAnalysis - Task2 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 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_CAMP_EXEC_SUMMARY', you want to process.
8. The data file name will be blank for any Table to Table Load mode. Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, 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. Execute the batch created in the preceding steps.

11.9.1 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 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_CAMPAIGN_EXEC_SUMMARY$

11.10 Fact Response

Fact Response entity stores all the responses for the campaign that was executed. The fact entity stores information such as response type, status, channel, product, offer channel, wave, offer, treatment, and so on.

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

Table 27: Fact Response T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_FCT_RESPONSE	STG_RESPONSE	FCT_RESPONSE

To view the detailed structure of the earlier tables, see Erwin Data Model.

11.10.1 Prerequisites

T2Ts related to the following should be executed before loading the Fact Response table:

- Fact Campaign Details
- Fact Campaign Summary
- Fact Common Customer Summary

Following are the lists of tables used in the population of Fact Response and these tables are required to be loaded before running the T2T:

- FCT_CAMPAIGN_DETAILS
- DIM_CAMPAIGN
- DIM_REGION
- DIM_PRODUCT

- DIM_CAMPAIGN_CHANNEL
- DIM_OFFER
- DIM_TREATMENT
- DIM_WAVE
- DIM_VENDOR
- DIM_DATES
- DIM_MKTG_PROGRAM
- DIM_CONTACT
- DIM_REJECTION_REASON
- DIM_RESPONSE_TYPE
- DIM_CHANNEL
- DIM_MARKET_CELL
- DIM_CUSTOMER
- FCT_COMMON_CUSTOMER_SUMMARY
- DIM_PROSPECT
- DIM_PROFESSION
- DIM_CALL_TYPE
- DIM_CAMPAIGN_SOURCE_TYPE

For more information, see the Fact Campaign Details section on populating campaign details and see the Campaign Summary section for details on populating Campaign Summary fact table. See the Customer Summary Population section for details on populating Common Customer Summary fact table.

For more information, see the Dimension Tables Population section under Dimension Loading Process chapter for details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on.

For more information, see the Time Dimension Population chapter for details on populating the DIM_DATES dimension table.

11.10.2 Executing the Fact Application 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_CampaignAnalysis - Task4 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 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_RESPONSE', you want to process.

8. The data file name will be blank for any Table to Table Load mode. Default value refers to any parameter that has to be passed to T2T. If there is any need for currency conversion in T2T transactions, 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. Execute the batch created in the preceding steps.

11.10.3   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_RESPONSE$

11.11   Fact Overlapping Campaign

Fact Overlapping Campaign entity stores the summary information related to prospects who were targeted by multiple campaigns at a point in time.

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

Table 28: Fact Overlapping Campaign T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_OVERLAPPING_CAM PAIGN	STG_OVERLAPPING_CAMPAIGN	FCT_OVERLAPPING_CAMPAIGN

To view the detailed structure of the earlier tables, see Erwin Data Model.

11.11.1 Prerequisites

Following are the lists of tables used in the population of the Fact Overlapping Campaign and these tables are required to be loaded before running the T2T:

- DIM_DATES
- STG_OVERLAPPING_CAMPAIGN

For more information, see the [Time Dimension Population](#) chapter for details on populating the DIM_DATES dimension table.

See the Download Specifications for identifying fields required in Stage Customer Master and Stage Customer Details for Customer Insight Application(s).

11.11.2 Executing the Fact Application 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_CampaignAnalysis - Task5 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 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_OVERLAPPING_CAMPAIGN', you want to process.
8. The data file name will be blank for any Table to Table Load mode. Default value refers to any parameter that has to be passed to T2T. It has to be blank.
9. Execute the batch created in the preceding steps.

11.11.3 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_OVERLAPPING_CAMPAIGN$

11.12 Fact Cross Sell Score

Fact Cross Sell Score entity stores Cross Sell Scores of the customers between product types. This fact entity is loaded from the Fact Common Account Summary table. The T2T loads data required for predictive models. The predictive models make use of this data for deriving the cross sell score between the product types for a customer and the cross sell scores are updated back in this fact.

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

Table 29: Fact Cross Sell Score T2T Definitions

T2T Definition Name	Source Staging Table	Destination Table
T2T_XSELL_CARDS_TO_CASA	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE
T2T_XSELL_TD_TO_CARDS	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE
T2T_XSELL_CARDS_TO_MORT	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE
T2T_XSELL_CASA_TO_CARDS	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE
T2T_XSELL_CASA_TO_MORT	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE
T2T_XSELL_MORT_TO_CARDS	FCT_COMMON_ACCOUNT_SUMMARY	FCT_XSELL_SCORE

For the detailed structure of the earlier tables, see Customer Insight Erwin Data Model.

11.12.1 Prerequisites

Following are the lists of tables used in the population of Fact Cross Sell Score and these tables are required to be loaded before running the T2T:

- FCT_COMMON_ACCOUNT_SUMMARY

For details on populating dimension tables like DIM_CUSTOMER, DIM_BANDS, and so on, see the Dimension Tables Population section under the Dimension Loading Process chapter.

For more information on populating account dimensions, see the Account Dimension Population chapter.

For details on populating the DIM_DATES dimension table, see the Time Dimension Population chapter.

11.12.2 Executing the Fact Cross Sell Score Population T2Ts

To execute the T2T component from the OFSAA Infrastructure ICC framework (accessed through the application Batch Operations screen), a seeded batch, <Infodom>_ XSell_Score 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 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. The data file name will be blank for any Table to Table Load mode and the default value should be null.

8. Repeat steps 4 to 8 for adding the remaining 11 T2Ts within the same batch definition.

9. Execute the batch created in the preceding steps.

11.12.3 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_XSELL_SCORE

11.13   Fact Account Profitability

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

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

Table 30: Fact Account Profitability Definition

T2T Definition Name	Source Staging Table	Destination Table
FN_FCT_ACCOUNT_PFT	FCT_COMMON_ACCOUNT_SUMMARY FCT_PFT_ACCOUNT_SUMMARY FCT_FTP_ACCOUNT_SUMMARY FCT_REG_CAP_ACCOUNT_SUMMARY FCT_ECO_CAP_ACCOUNT_SUMMARY	FCT_ACCOUNT_PROFITABILITY

For more information and to view the detailed structure of the tables, see Oracle Financial Services Analytical Applications Data Model Data Dictionary and the Erwin Data Model.

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

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 item. To know more about AMHM, see the OFS Analytical Applications Infrastructure User Guide.

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

Hierarchy is the two hierarchies that are used for mapping. Reporting Line Hierarchy is a parent-child hierarchy that is based on Reporting Line Dimension entity.

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

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

For more information on defining/maintaining Mapper, see OFS Analytical Applications Infrastructure User Guide.

Steps to Define Mapping for Custom Reporting Line Items

To define the mapping for Custom Reporting Line items, follow these steps:

1. Add Custom Reporting Line or Modify existing Reporting Line.

2. Add Custom Reporting Line Hierarchy or modify existing seeded reporting line hierarchy.

3. Execute the seeded Batch <INFODOM>_ Repline_Dimension_Update specifying the Reporting line hierarchy as a parameter to Batch.

4. Modify the seeded Business Metadata.

5. Map Maintenance.

The following sections describe these steps.

11.13.1   Add Custom Reporting Line or Modify existing Reporting Line

Custom Reporting Lines can be added or modified from AMHM. Following are the seeded attributes of Reporting Line Dimension:

- Financial Element Code

- GL Account Code

- Rollup Signage

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.

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 24: Reporting Line Hierarchy

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

    a. Enter Numeric Code.

    b. Enter the Name of the custom reporting line.

    c. Enter the Description of the custom reporting line.

    d. Select **Yes**, if the custom reporting line has to be Enabled or not.

    e. Select **Yes**, if the custom reporting line Is Leaf or not.

    f. Select the Attributes for the reporting line member.

    g. **Save** the Member definition.

To modify a reporting line:

1. Click the **Edit** button from the Members screen. The Member Definition (Edit Mode)screen is displayed.

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

    a. Modify the Name of the custom reporting line.

    b. Modify the Description of the custom reporting line.

    c. Modify the selection of the radio button in the Enabled field.

    d. Modify the selection of the radio button in the Is Leaf field.

    e. Modify the Attributes for the reporting line member.

    f. Save the Member definition.

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

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

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

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

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

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

3. Execute the seeded batch <Infodom>_ Repline_Dimension_Update specifying the Reporting line hierarchy as a parameter to batch.

4. Execute the seeded batch <Infodom>_ Repline_Dimension_Update. It populates data into the DIM_REP_LINE table.

To modify an existing seeded Reporting Line Hierarchy, follow these steps:

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

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

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

For more information, see the OFS Analytical Applications Infrastructure User Guide.

11.13.2 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.

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 OFS Analytical Applications Infrastructure User Guide.

11.13.3 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 OFS Analytical Applications Infrastructure User Guide.

11.13.4 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.

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.

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

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

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

11.13.5 Prerequisites

Following are the lists of tables used in the population of Fact Account Profitability. 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 SCD, see the Dimension Loading Process chapter.

11.13.6 Executing the Fact Account Profitability Population DT

You can execute the T2T component from the Operations (formerly Information Command Center (ICC) framework) module of OFSAAI.

Define a new Batch and an underlying Task definition from the Batch Maintenance window of OFSAAI. For more information on defining a new Batch, see How to Define a Batch.

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

To define a new task for a Batch definition, follow these steps:

1. Click the check box in the Batch Name container to select the Batch.
2. Click **Add** (+) button from the Task Details grid. The Task Definition window is displayed.
3. Enter the **Task ID** and **Description**.
4. Select **TRANSFORM DATA** from the Components list.
5. 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_FCT_ACCOUNT_PFT** from the list.
6. 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.

7. Click **Save**. The Task definition is saved for the selected Batch.

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

For more comprehensive coverage of configuration and execution of a batch, see OFS Analytical Applications Infrastructure User Guide.

11.13.7   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 executionID. The following tables can be queried for errors.

11.13.8   Update Bands in Fact Tables

You have to update the band values based on the scores in certain cases. For instance, a predictive model's execution derives the score values, which are updated to the fact tables. Based on the new score values, it is necessary to have the new band values updated in the fact tables. A Data Transformation "Update_Bands" is seeded to update the bands in fact tables. Update of bands in fact tables makes use of a setup table FSI_BAND_SETUP_DETAILS.

Table 31: FSI_BAND_SETUP_DETAILS

Column Name	Data Type	Column Description
TABLE_NAME (PK)	VARCHAR2(30)	This stores the name of the table of the source and the target column.
SRC_COLUMN_NAME (PK)	VARCHAR2(30)	This stores the name of the source column based on which the bands would be updated in the target column.
TGT_COLUMN_NAME (PK)	VARCHAR2(30)	This stores the name of the target column where the bands are updated.
BAND_TYPE	VARCHAR2(30)	This stores the band type which has to be used from the DIM_BANDS table.

Seeded entries into the FSI_BAND_SETUP_DETAILS table are provided with the installer to update the attrition score band in the table FCT_CRM_ACCOUNT_SUMMARY and product propensity score band & product propensity segment band in the FCT_XSELL_SCORE table.

Execute the seeded batch <Infodom>_Update_Bands. The parameters passed to DT " Update_Bands" are:

- Batch Run ID: This is passed internally to the DT from the Batch in Operations modules of OFSAAI.

- FIC MIS Date/As of Date: This is passed internally to the DT from the Batch in Operations modules of OFSAAI.

- Band Type: You have to provide the values in the Parameter List of Batch Maintenance screen.

The following values can be entered:

Table 32: Band Type Parameters

Band Type to be updated	Parameter to be passed in DT
Account Attrition Score Band	ACCT_ATTRITION_SCORE
Product Propensity Score Band	PRODUCT_PROP_SCORE

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

For more information on defining a new Batch, see How to Define a Batch, section. To define a new task for a selected Batch definition, follow these steps:

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

2. Click **Add** (+) button from the Task Details grid. The Task Definition window is displayed.

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

4. Select 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 dropdown list.

8. Select the **Rule Name Update_Bands** from the drop-down list.

9. Enter the **Parameter List** details as mentioned below:

   ▪ Band Type: See Table 33 for the values which can be passed.

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

10. Click **Save**. The Task definition is saved for the selected Batch.

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

12 Predictive Modeling

OFS Retail Customer Analytics currently comprises of the following Predictive Models:

- Cross Sell Model
- Attrition Model
- Prepayment Analysis
- Channel Propensity Analysis
- Product Association Modeling
- Account Forecast Modeling

The following sections describe each of these models.

12.1 Cross Sell Model

Cross-Sell Model predicts the propensity of a Customer of a Source Product Type to purchase a product in the Target Product Type. The propensity is a probability value between 0 and 1.

12.1.1 Technique: Logistic Regression

Logistic regression is a statistical technique for predicting the outcome of a categorical dependent variable (a dependent variable that can take on a limited number of categories) based on one or more predictor variables (independent variables). The probabilities describing the possible outcome of a single trial are modeled, as a function of explanatory variables, using a logistic function. Logistic regression can be binomial or multinomial. In Cross Sell Model, binomial logistic regression is used.

12.1.2 Dependent Variable

The Dependent Variable for this model is defined as 1 if a customer who owned a product of the Source Product Type all through the historic period considered and just owned a product of the Target Product Type in the current period considered and 0 otherwise.

For example, a customer who owned a Platinum Credit Card (Product Type CARDS) all through the history (say 3 months) and opened a Savings Bank Account (Product Type CASA) 1 month ago (after the end of the historic period) falls in this category.

12.1.3 Data Considered

The historic Period to be considered is an input parameter to the model. The data on which prediction happens is the last available month for which data is available. The data on which the model fitting happens is all the data before the prediction period.

12.1.4 Independent Variables

Independent variables for these models are variables that describe Customer Demographics, Account/Customer Activity related information. The relevant variables for a specific Source - Target Combination are chosen appropriately.

The numerical independent variables are averaged over the entire historic calibration period. Categorical variables such as Gender, Product, Marital Status, and Profession are considered as segments, and calibration for the model is done group-wise for each relevant combination of these segments.

12.1.5 Source Product Type - Target Product Type combinations

Following Source Product Type - Target Product Type combinations are currently modeled in OFS Customer Analytics:

- Cross Sell CASA to Cards
- Cross Sell Cards to CASA
- Cross Sell Cards to Mortgage
- Cross Sell CASA to Mortgage
- Cross Sell Mortgage to Cards
- Cross Sell TD to Cards

For detailed information on the technique and variables used, see the Model Metadata Sheet.

12.2 Attrition Model

The attrition model predicts the probability of a Customer of a Product Type to churn i.e. close the account. The probability is a value between 0 and 1.

Both ORE-R and ORA are used for the attrition model.

12.2.1 Technique: Logistic Regression

In Attrition Model, binomial logistic regression is used.

12.2.2 Dependent Variable

The Dependent Variable for this model is defined as 1 if an account of a Product Type has been open all through the historic period considered and has just been closed in the current period considered and 0 otherwise.

For example, a Platinum Credit Card account (Product Type CARDS) has been open all through the history (say 3 months) and has been closed 1 month ago (after the end of the historic period) falls in this category.

12.2.3 Data Considered

The historic Period to be considered is an input parameter to the model. The data on which prediction happens is the last available month for which data is available. The data on which the model fitting happens is all the data before the prediction period.

12.2.4   Independent Variables

Independent variables for these models are variables that describe Customer Demographics, Account/Customer Activity related information. The relevant variables for a specific Product Type are chosen appropriately.

The numerical independent variables are averaged over the entire historic calibration period. Categorical variables such as Gender, Product, Marital Status, Profession are considered as segments, and calibration for the model is done group-wise for each relevant combination of these segments.

12.2.5   Weblog Variables

Weblog variables for these models are data elements that enhance the prediction of the models. The weblog variables are the recency and frequency of the event and the aggregate of the duration spent by the customer on the event for the entire day.

Following are the applicable list of events:

Table 33: List of Events

Event	Description
Fees	This category type lists all the Fees and Charges that are applicable to a product.
Terms and Conditions	This category type lists all the terms and conditions applicable to a product.
Application Start Page	This category type displays the first page of a multi-page activity, typically the opening of an account.
Application submission confirmation	This category type displays the last page of a multi-page activity, typically the confirmation of the opening of an account.
Calculator	This category type helps the customers to determine values such as EMI, Interest earned (in case of products like RD), and so on.
Product Details	This category type displays the page where product features are highlighted.
Customer Complaint	This category type displays the page where customers log their product or service complaints.
Financial Planning	This category type helps the customers to set and track financial goals.
Bill Payment	This category type displays the page after the customer logs in to pay bills.
Rewards/ Offers	This category type lists all the rewards and offers that are available from the bank.

12.2.5.1   Weblog Processing

Weblog processing involves processing the weblogs, which are data generated through user activity in the bank website. These weblogs are captured once a day using Apache Hadoop. The AAI platform then moves this data to the hive tables. Weblog variables for these models are data elements that enhance the prediction of the models. For more information, see the Weblog Processing for RCA section.

12.2.6   Product Types

Following Product Types are currently considered for Attrition in OFS Customer Analytics:

- Account Attrition - Cards
- Account Attrition - TD
- Account Attrition - CASA

Following is a sample list of product types and corresponding products:

Table 34: Sample list of Product Types and Corresponding Products

Product Type	Product Type Code	Product	Product Code
Auto Loans	AUTOLOAN	Car Loan	9039
Cards	CARDS	Cards	9007
Cards	CARDS	Gold Card	9026
Cards	CARDS	Platinum Card	9025
Cards	CARDS	Platinum Plus	9028
Cards	CARDS	Signature Card	9027
Contracts	CONTRACTS	Annuity Plus	9011
Contracts	CONTRACTS	Coporate Contracts	9003
Contracts	CONTRACTS	Leases	9013
Contracts	CONTRACTS	Other Contracts	9014
Current Savings	CASA	Family Savings Group Account	9032
Current Savings	CASA	Plus Current	9033
Current Savings	CASA	Regular Savings Account	9030
Current Savings	CASA	Savings(Corp)	9004
Current Savings	CASA	Savings(Ret)	9008
Current Savings	CASA	SavingsMaxAccount	9029
Current Savings	CASA	Senior Citizens Account	9031
Investments	INVEST	Corp Invest	9005
Investments	INVEST	Equi Plus (Ret)	9036
Investments	INVEST	Equity Plus	9020
Investments	INVEST	MF Long Gain	9021
Investments	INVEST	MF Regular	9037
Investments	INVEST	Retail Invest	9009
Loan	LOAN	Corp Loans	9006

Product Type	Product Type Code	Product	Product Code
Loan	LOAN	Gold Loan	9040
Loan	LOAN	Home Loan	9038
Loan	LOAN	Retail Loan	9010
Mortgages	MORTGAGE	Business Loans	9022
Mortgages	MORTGAGE	Government Loans	9024
Mortgages	MORTGAGE	Loans Against Assets	9023
Mortgages	MORTGAGE	Mortgage Plus	9012
Term Deposits	TD	Regular Fixed Deposit	9034
Term Deposits	TD	Super Saver Deposits	9035
Term Deposits	TD	Sweep In Deposits	9019

For detailed information on the technique and variables used, see the Model Metadata Sheet.

12.3 Prepayment Analysis

Pre Payment Model predicts the probability of a Customer to pre-pay on his/her loan. The probability is a value between 0 and 1.

12.3.1 Technique: Logistic Regression

In Pre Payment Model, binomial logistic regression is used.

12.3.2 Dependent Variable

The Dependent Variable for this model is defined as 1 if a loan account of a Product Type LOANS has been closed before completion of 95% of the maturity period in the historic period considered and 0 otherwise.

12.3.3 Data Considered

The historic Period to be considered is an input parameter to the model. The data on which prediction happens is the last available month for which data is available. The data on which the model fit.

12.3.4 Independent Variables

Independent variables for this model are variables that describe Customer Demographics, Account/Customer Activity related information. The numerical independent variables are averaged over the entire historic calibration period. Categorical variables such as Gender, Product, Marital Status, and Profession are considered as segments, and calibration for the model is done group-wise for each relevant combination of these segments.

For detailed information on the technique and variables used, see the Model Metadata Sheet.

12.4   Channel Propensity Analysis

Channel Propensity Model predicts the relative probability of a Customer to respond through a particular response channel. The probability is obtained for a Customer of a particular Product Type who was part of a campaign of a particular Campaign Type. The probability is a value between 0 and 1. Since the result obtained is the set of relative probabilities, the sum of the probabilities of all the channels together will be 1.

For example, a customer belonging to a particular product type who was part of a particular campaign type displayed Channel Propensities of 0.5 for Telemarketing, 0.3 for Email, and 0.2 for Direct Mail.

Both ORE-R and ORA are used for the propensity model.

12.4.1   Technique: Multinomial Logistic Regression

Multinomial Logistic Regression is a classification method that generalizes logistic regression to multi-class problems, that is, with more than two possible discrete outcomes. That is, it is a model that is used to predict the probabilities of the different possible outcomes of a categorically distributed dependent variable.

12.4.2   Dependent Variable

The Dependent Variable for this model is defined as the response channel through which a customer has responded.

12.4.2.1   Data Considered

The historic Period to be considered is an input parameter to the model. The data on which prediction happens is the last available month for which data is available. The data on which the model fitting happens is all the data before the prediction period.

12.4.2.2   Independent Variables

Independent variables for this model are variables that describe Customer Demographics, Campaign/Channel related statistics, Account/Customer Activity related information. The numerical independent variables are averaged over the entire historic calibration period. Categorical variables such as Gender, Product, Marital Status, and Profession are considered as segments, and calibration for the model is done group-wise for each relevant combination of these segments.

12.4.2.3   Weblog Variables

Weblog variables for these models are data elements that enhance the prediction of the models. The weblog variables are the recency and frequency of the event and the aggregate of the duration spent by the customer on the event for the entire day.

Following are the applicable list of events:

Table 35: List of Events

Event	Description
Fees and Charges	This category type lists all the Fees and Charges that are applicable to a product.
Terms and Conditions	This category type lists all the terms and conditions applicable to a product.
Application Start Page	This category type displays the first page of a multi-page activity, typically the opening of an account.
Application submission confirmation	This category type displays the last page of a multi-page activity, typically the confirmation of the opening of an account.
Calculator	This category type helps the customers to determine values such as EMI, Interest earned (in case of products like RD), and so on.
Product Details	This category type displays the page where product features are highlighted.
Customer Complaint	This category type displays the page where customers log their product or service complaints.
Financial Planning	This category type helps the customers to set and track financial goals.
Bill Payment	This category type displays the page after the customer logs in to pay bills.
Rewards/ Offers	This category type lists all the rewards and offers that are available from the bank.

12.4.2.4 Weblog Processing

Weblog processing involves processing the weblogs, which are data generated through user activity in the bank website. These weblogs are captured once a day using Apache Hadoop. The AAI platform then moves this data to the hive tables. Weblog variables for these models are data elements that enhance the prediction of the models. For more information, see the Weblog Processing for RCA section.

For detailed information on the technique and variables used, see the Model Metadata Sheet.

12.5 Product Association Modeling

Product Association Model provides a list of Product Basket - Target Product combinations that are most likely to occur based on historic data. For example, a Credit Card Product being sold to a customer who owns a Term Deposit account, a Mortgage account and an Auto loan account emerges as one of the most common combinations of Product Basket - Target Product.

12.5.1 Technique: Apriori

Apriori is an algorithm for frequent item set mining and association rule learning over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database.

12.5.2 Data Considered

Complete historic data available is used. The number of significant item sets obtained will vary based on the input parameters: Support Probability and Confidence Probability.

For detailed information on the technique and variables used, see the Model Metadata Sheet.

12.6   Account Forecast Modeling

Account Forecast Model estimates the future values of a time series. The future values are obtained for the desired number of lead periods considering a desired (but sufficient) amount of history.

12.6.1   Technique: ARIMA

Autoregressive Integrated Moving Average (ARIMA) model is fitted to a time series to predict future points in the series. The model is generally referred to as an ARIMA (p, d, q) model where parameters p, d, and q are non-negative integers that refer to the order of the autoregressive, integrated, and moving average parts of the model respectively.

12.6.2   Time Series

The Time Series for this model is the reporting line value for each reporting line of each account.

12.6.3   Data Considered

The accounts that are considered for ARIMA fall into one of the three categories:

- The account has sufficient historic data points to be considering its history for future value estimation. (Typically this period is 2 years)

- The account has insufficient historic data points to consider its data for forecasting but has enough to be estimated based on other accounts that fall into its segment. Typically if an account has been open for at least 6

  - belong to the same segment as that account

  - fall into point 1 above

  - have been opened less than 24 months ago

- The account does not have significant historic data points to be considered for future values estimation. Typically, accounts that have been opened less than 6 months ago.

For detailed information on the technique and variables used, see the Model Metadata Sheet.

For seeded techniques of Advanced Analytics Infrastructure (AAAI), the models could be of the type NAG or R. Default is set to NAG. This configuration change needs to be done in a database table in the CONFIG schema of OFSAAI to be able to see the R models. The following query needs to be run:

```
UPDATE CONFIGURATION SET PARAMVALUE='R' WHERE PARAMNAME='F_MODEL_TYPE'

/ COMMIT

/
```

13 Model Creation and Execution

Models are built based on various techniques associated with executable and related parameters based on the business purpose. In the Infrastructure system models are defined in the metadata abstraction layer using the underlying metadata objects such as Measures, Hierarchies, and Datasets along with statistical techniques.

Model Creation in the Sandbox Tab of Infrastructure system facilitates you to construct multiple models based on the required parameters and output specifications.

Figure 25: Model Creation page
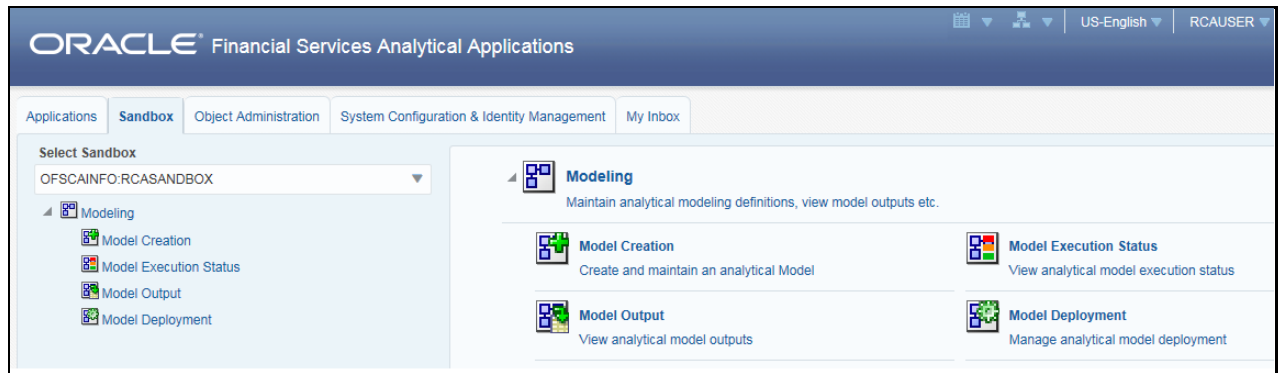
The Model Creation screen displays model definition details such as Model ID, Model Name, Version, Model Objective, Created By, and Created Date. You can also view, modify, and delete model definitions.

Figure 26: Model Management
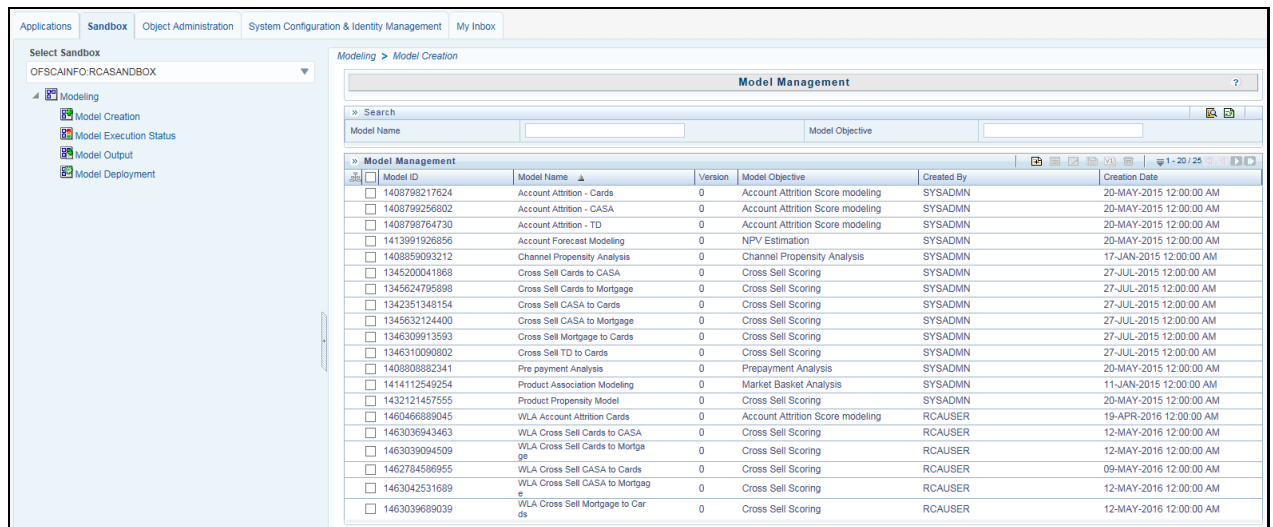
You can also make use of Search and Pagination options to search for a specific model or view the list of existing model definitions within the system.

13.1 Adding a New Model

See the following sections for adding a new model.

- Create Model Definition

- Modify Model Definition

13.1.1 Create Model Definition

To create a model definition in the Model Creation screen, follow these steps:

1. Select **Add** from the Model Management toolbar. This button is disabled if you have selected any Model ID in the grid. The Model Definition New screen is displayed.

2. Enter the details for the model:

   - Name

   - Description

   - Objective (to add a new objective, right-click on the heading and add)

   - Dataset

   - Technique: This can be defined in two ways

     — Write the technique in the Model Definition Screen itself in the Model Script section

     — Define a technique in the Technique Registration screen Navigate to Applications > Model Management > Technique Registration > Add > Script Console.

3. Write the script for technique in either the Model Script (Model Screen) / Script Console (Technique Registration) section.

4. Add the inputs and variables as applicable to the script

   > **NOTE**   For information on managing variables, see the Managing Variables section in OFS Enterprise Modeling User Guide.

5. Click **OK** and save the model.

13.1.2 Modify Model Definition

You can update the model definition details of an existing Model in the Model Definition screen.

Figure 27: Model Definition - Edit Mode
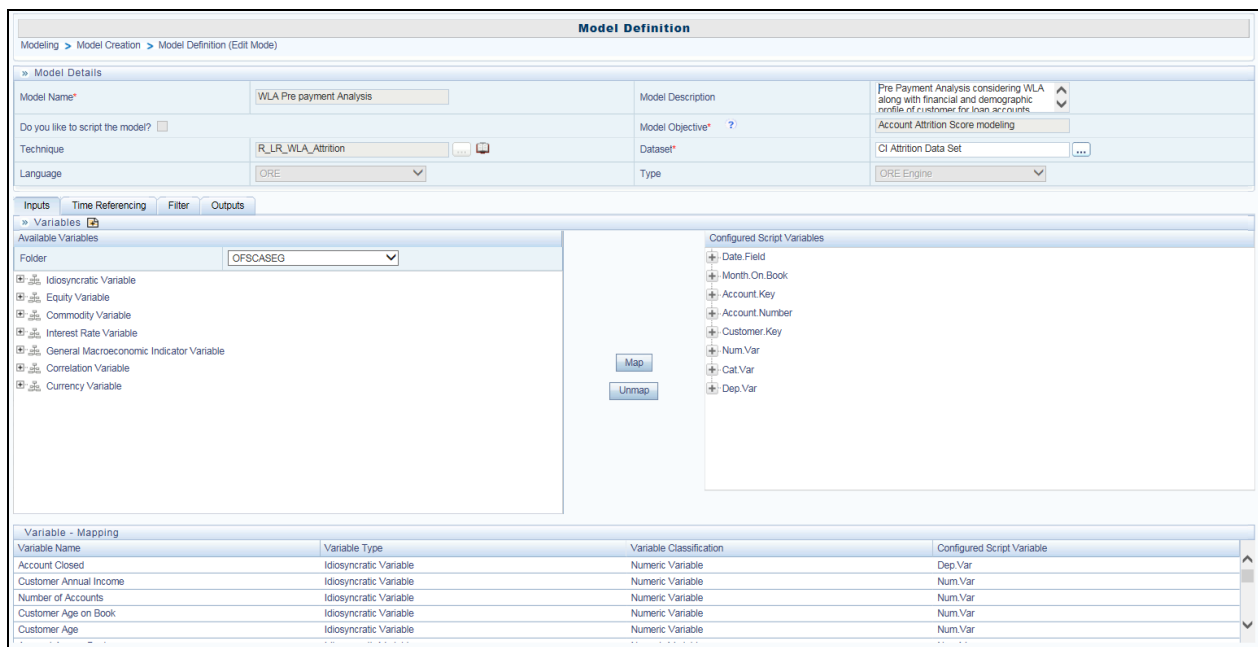
The filter providing Segment above available variables section lists all available segments for the user and works as a filter for selecting the available variables under each segment.

1. Select the check box adjacent to the Model ID whose details are to be updated.

2. Click **Edit** in the Model Management toolbar. This button is disabled if you have selected multiple Model IDs.

3. Edit the Model Definition details as required. Model Name, Technique, and Model Objective are not editable. You can update the Model Description, Dataset, and variable parameters based on the technique selected.

4. Once you have updated all the necessary details in the Model Definition Edit screen, you can:

   a. Select **Preview Data** to view the new Model Definition details before upload.

   b. Click **Save** to update the model definition details.

   c. Click **Save** and select **Execute** to process the model execution. The status of which can be verified in the Model Execution Status option by accessing Advanced Analytics Infrastructure > Modeling > Model > Model Execution Status.

An input called Model Code is added in the Input Parameters section to indicate a code that maps configurations to choose events for data models. Events relevant to a model are identified using a model code. Historic and prediction periods are also indicated in the data model configuration.

> **NOTE**   USE.WLA.DURATION
> /USE.WLA.FREQUENCY/USE.WLA.RECENCY can be used
> temporarily until the multiple data set functionality is provided.

For more information, see the *Model Management* in the OFS Analytical Applications Infrastructure User Guide.

13.2   Model Execution

Model Execution happens in two stages: Sandbox Execution and Production Execution. Sandbox Execution of a model happens on the Sandbox Schema where the data is used to calibrate the model. Ideally, the model fit in the Sandbox during Calibration is used to predict the results for the data found in Production Schema.

13.2.1   Model Execution (Sandbox)

To execute the model, follow these steps:

1.   Click the **Execute** Button (as shown below) and select Baseline.

Figure 28: Baseline (when the technique is directly written in the Modeling Screen)
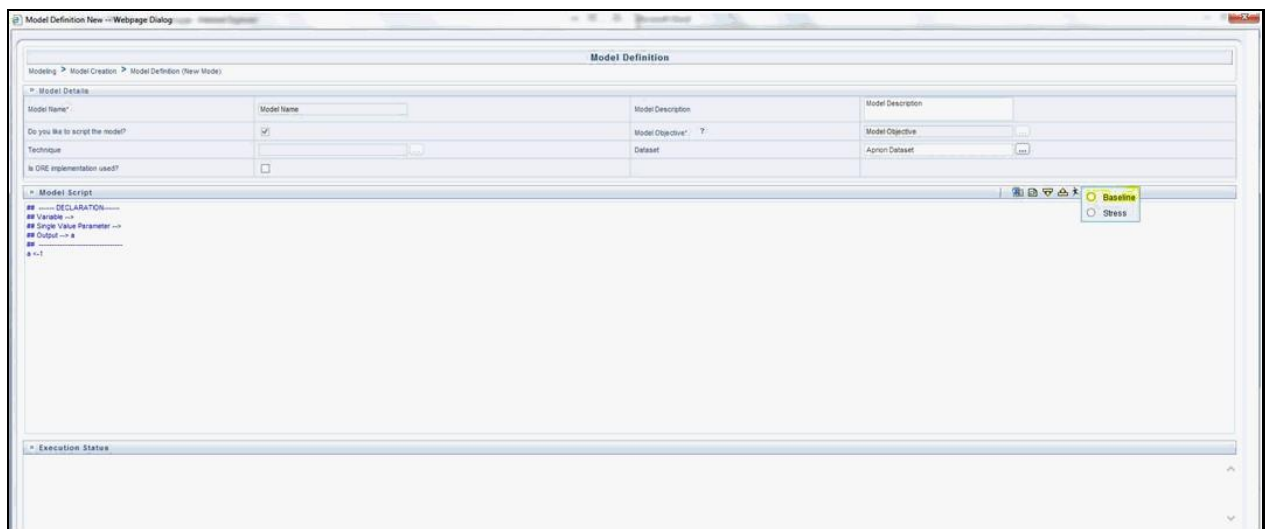
Figure 29: Baseline (when the technique is written in Technique Registration and selected in the Technique screen)
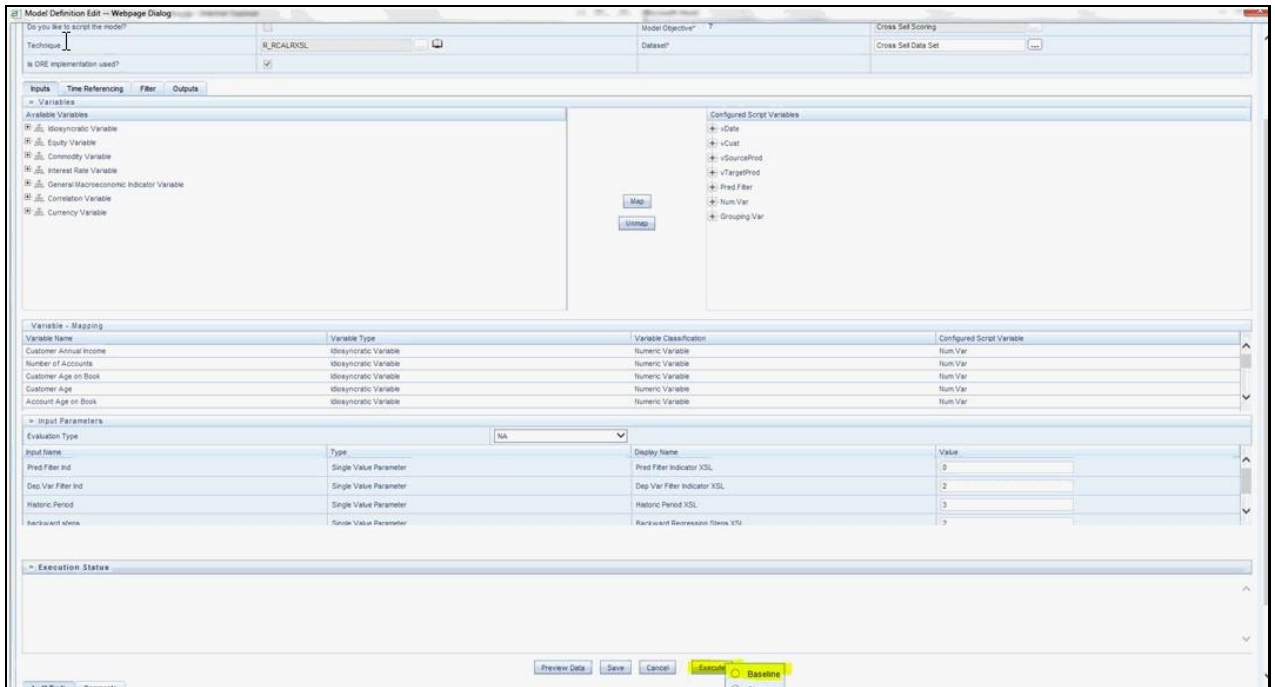
2. After the model execution is triggered, the following message appears: *Successfully triggered the model execution*.

3. Check the model execution status in the Model Execution Status window (sort descending by date for the status of the latest execution).

Figure 30: Model Execution Status
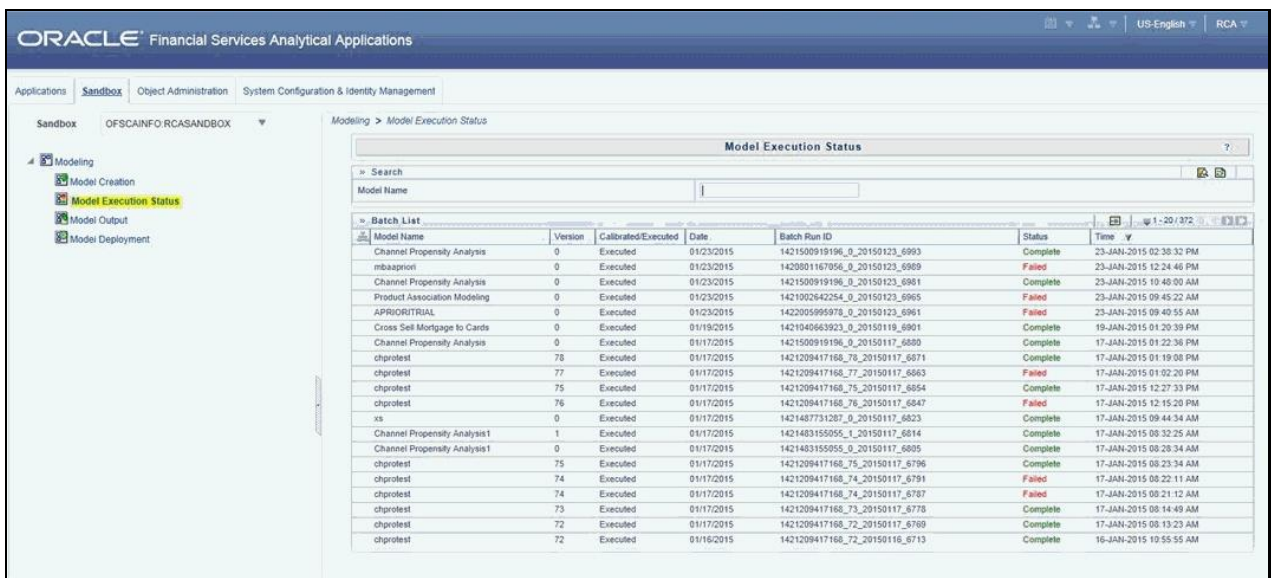
4. After the model execution is successful, Model Output can be viewed in the Model Output screen.

Figure 31: Model Output
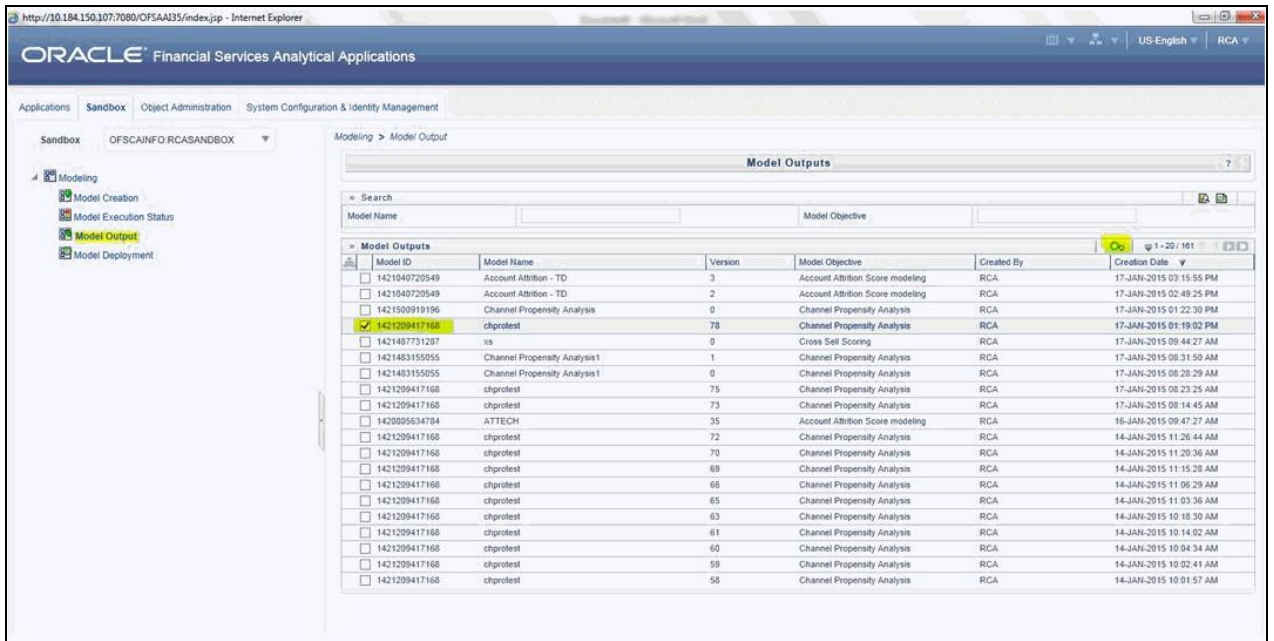
5. Select the execution ID for which the Output is to be viewed.

Figure 32: Outputs Summary
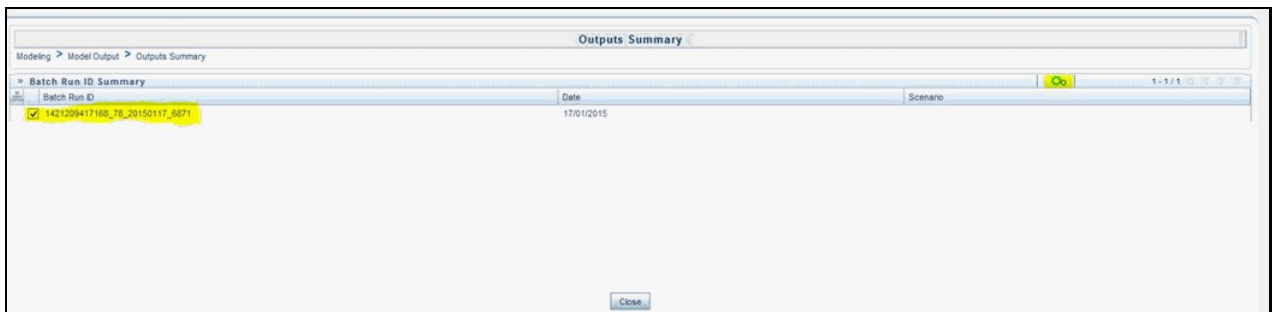
6. Click the Model Output desired to be viewed.

Figure 33: View Sample
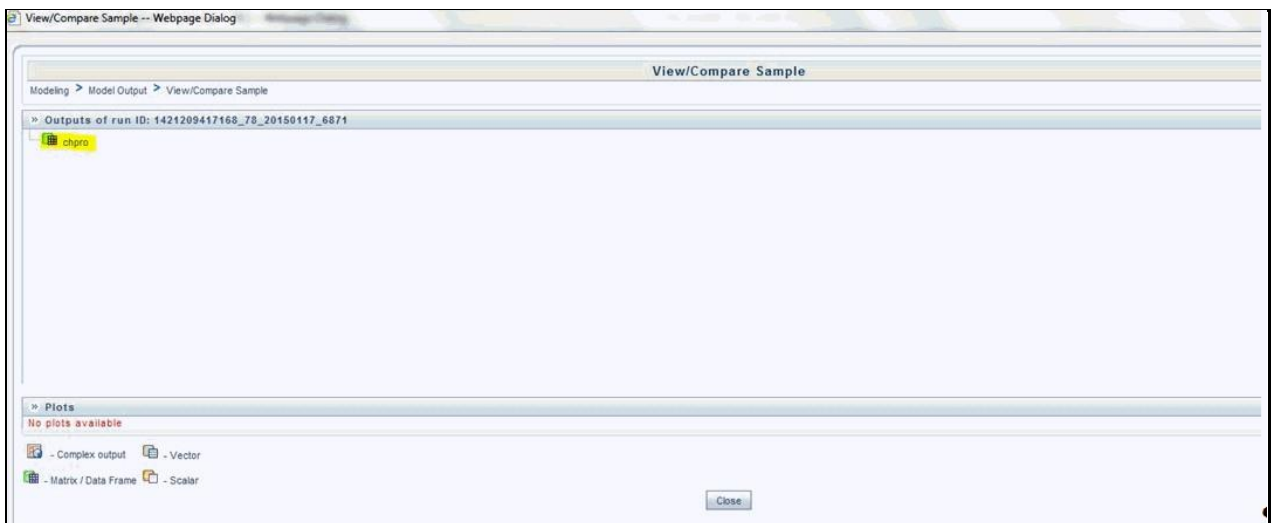
7. After the model execution is successful and it is confirmed that the model script is ready to be executed in the Production Schema, navigate to Model Deployment.

Figure 34: Model Deployment
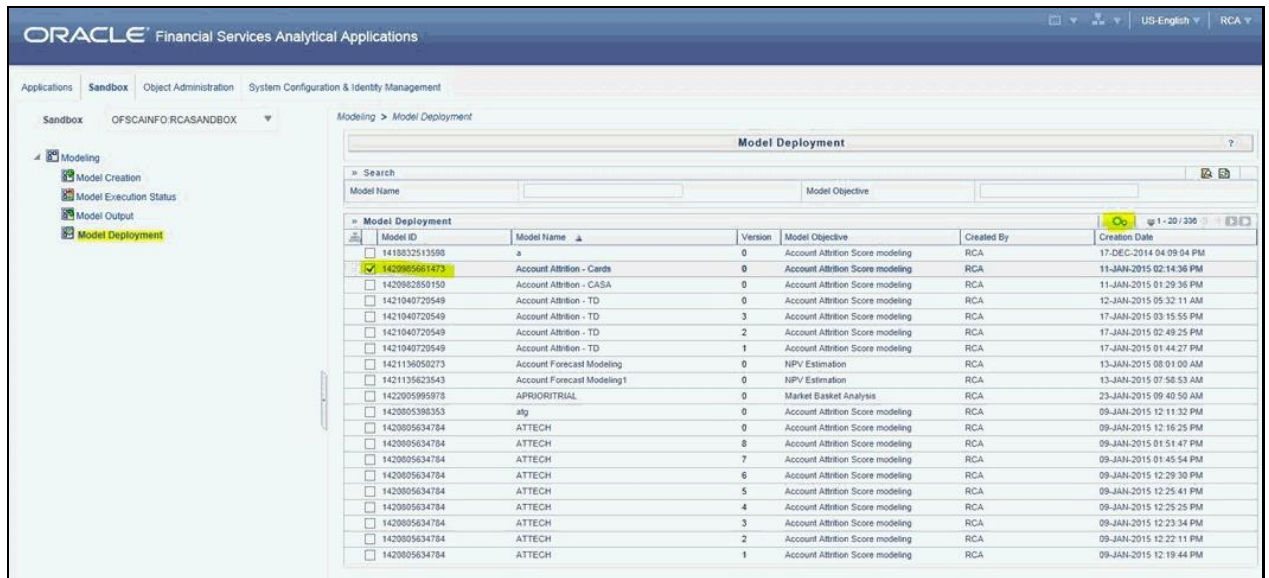
8. Authorize and Deploy the model to Production Schema.

Figure 35: Model Deployment Authorization

13.2.2   Model Execution (Production)

A Model becomes available for execution in the Production Infodom after it has been authorized and deployed in the sandbox. Once a request is processed for the execution in the Production Infodom, a Batch is registered for the Model. This relevant Batch can be executed to obtain the results.

1. Navigate to Applications, select Model Execution, and then select Request for Model Execution.

Figure 36: Request for Model Execution

2. Select the check box in the Register Batch field and click **Save**. A Batch ID is created for the Model Execution.

13.2.2.1 Batch Execution

To execute the batch, follow these steps:

1. Navigate to **Operations**, select **Batch Execution**, and then select the module **Enterprise Modeling**.

2. Find the Batch ID beginning with the same name as the Model ID in the following format:
    <INFODOM>_<MODEL_ID>_<MODEL_VERSION>

3. Select the desired execution date and execute the batch.

13.2.2.2 Error Logging

To verify the status of the Model Execution, navigate to the following path:

`$ORACLE_HOME/hs/admin`

where ORACLE_HOME is the home of the database server hosting Oracle R Enterprise. A file is created with the batch execution ID of the batch.

13.2.2.3 Loading Data to the Target Table

After the Infodom Execution is complete, the results are recorded in the following tables with sequence numbers leading back to the execution process:

*   MF_MODEL_ORE_OUTPUT (Execution Summary)
*   MF_MODEL_ORE_DETAILS (Actual Output Values/Queries to produce Output Values)

Based on the sequence and reference numbers in these tables, the target table is updated. For this purpose, a Batch is triggered. The Batch is built on a process that can backtrack the relevant combination of primary keys and update the relevant records in the target table's target column(s).

It is important to have the required records in FSI_MODEL_PARAMETERS. See the *Model Metadata Sheet>Techniques* sheets for information on what information to add to this table in case a new model is created.

There are two ways the results are expected in the Reporting Tables:

- All the required rows are already present and the column corresponding to the score/probability needs to be updated.

- The relevant rows are all supposed to be inserted into the table afresh.

13.2.2.4 Result Update

For updating the result into the Fact table which already has the required rows, the name of the Batch registered for this purpose is ##INFODOM##_Model_Fact_Update - Task2. This batch accepts the input parameter list and based on this list it identifies the specific Model to update.

The list of parameters required are the following: (This is listed in the exact order)

Table 36: Input Parameters

Parameter	Sample Value/Column name from MF_MODEL_ORE_OUTPUT
Batch ID	'Sample_Batch'
Execution Date	FIC_MIS_DATE
Run ID	''
Process ID	''
Run Execution ID	V_BATCH_RUN_ID
Run Key	''
Model ID	V_MODEL_ID
Model Version	N_MODEL_VERSION
Sample ID	''
Object Name	V_OBJECT_NAME
Run Key	-1

Execute the Batch created to update data in the target table from the temporary table.

Figure 37: Batch Maintenance page
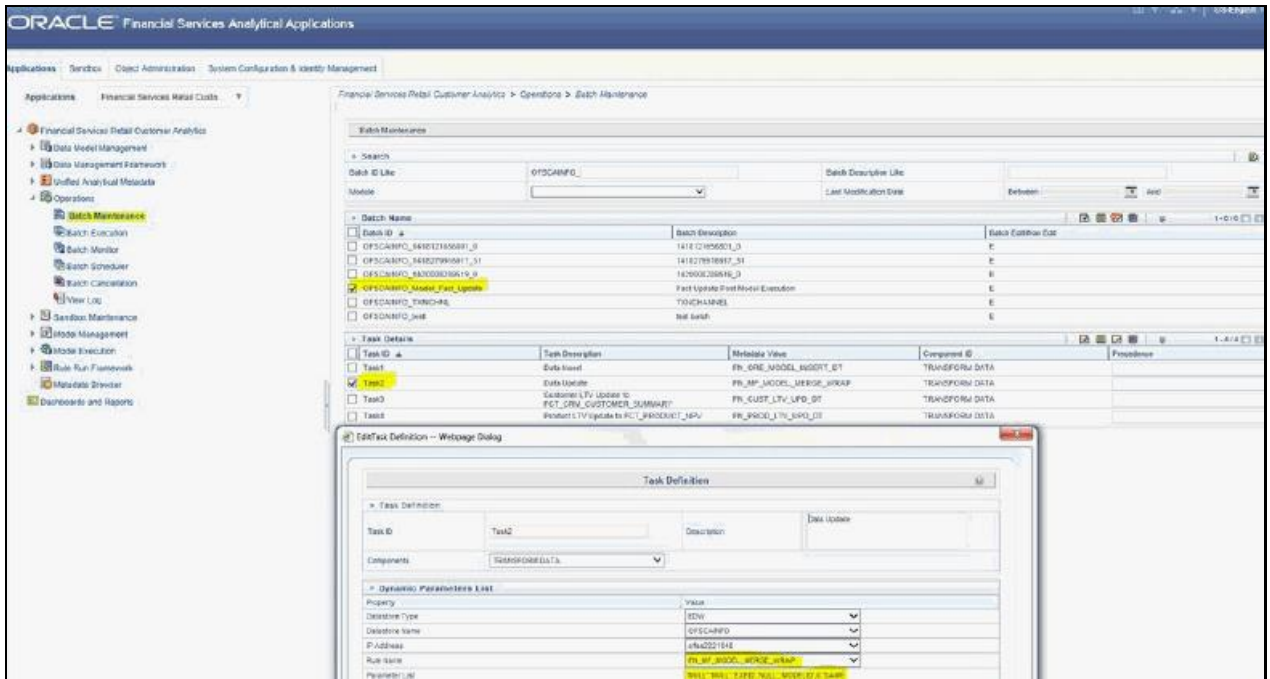
The parameters list must be updated with details relevant to the specific Model. These details are obtained from the temporary table - MF_MODEL_ORE_OUTPUT as mentioned in the table above.

Example

The following is an example of a successful model execution from MF_MODEL_ORE_OUTPUT.

Figure 38: MF_MODEL_ORE_OUTPUT Example

13.2.3 Result Insert

For inserting result data into the Fact table afresh, the name of the Batch registered for this purpose is ##INFODOM##_Model_Fact_Update - Task1. This batch accepts the input parameter list and based on this list it identifies the specific Model to update.

The list of parameters required are the following (This is listed in the exact order)

Table 37: Model_Fact_Update - Task1 Parameters

Parameter	Sample Value/Column name from MF_MODEL_ORE_OUTPUT
Batch ID	'Sample_Batch'
Execution Date	FIC_MIS_DATE
Run ID	''
Process ID	''
Run Execution ID	V_BATCH_RUN_ID

Parameter	Sample Value/Column name from MF_MODEL_ORE_OUTPUT
Run Key	"
Model ID	V_MODEL_ID
Model Version	N_MODEL_VERSION Run Key

13.3 Update Bands in Fact Tables

You have to update the band values based on the scores in certain cases. For instance, a predictive model's execution derives the score values, which are updated to the fact tables. Based on the new score values, it is necessary to have the new band values updated in the fact tables. A Data Transformation "Update_Bands" is seeded to update the bands in fact tables. Update of bands in fact tables makes use of a setup table FSI_BAND_SETUP_DETAILS.

Table 38: Update Bands

Column Name	Data Type	Column Description
TABLE_NAME (PK)	VARCHAR2(30)	This stores the name of the table of the source and the target column.
SRC_COLUMN_NAME (PK)	VARCHAR2(30)	This stores the name of the source column based on which the bands would be updated in the target column.
TGT_COLUMN_NAME (PK)	VARCHAR2(30)	This stores the name of the target column where the bands are updated.
BAND_TYPE	VARCHAR2(30)	This stores the band type which has to be used from the DIM_BANDS table.

Seeded entries into the FSI_BAND_SETUP_DETAILS table are provided with the installer to update the attrition score band in the table FCT_CRM_ACCOUNT_SUMMARY and product propensity score band & product propensity segment band in the FCT_XSELL_SCORE table.

Figure 39: Task Definition
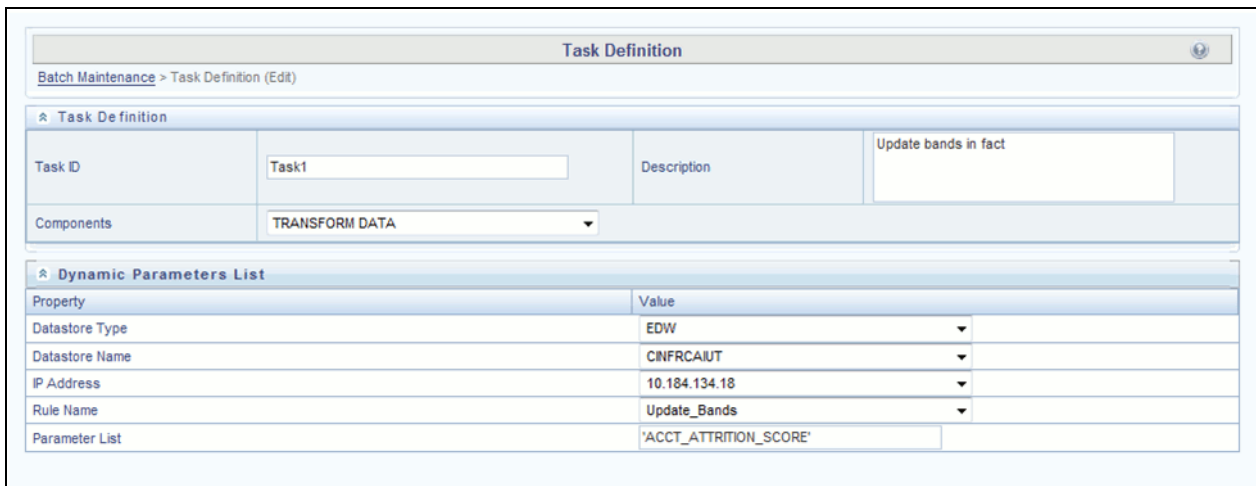
Execute the seeded batch <Infodom>_Update_Bands. The parameters passed to DT " Update_Bands" are:

- Batch Run ID: This is passed internally to the DT from the Batch in Operations modules of OFSAAI.
- FIC MIS Date/As of Date: This is passed internally to the DT from the Batch in Operations modules of OFSAAI.
- Band Type: You have to provide the values in the Parameter List of Batch Maintenance screen.

The following values can be entered:

Table 39: Band Type Parameters

Band Type to be updated	Parameter to be passed in DT
Account Attrition Score Band	ACCT_ATTRITION_SCORE
Product Propensity Score Band	PRODUCT_PROP_SCORE
Product Propensity Segment Band	PRODUCT_PROP_SEG

Figure 40: Batch Execution
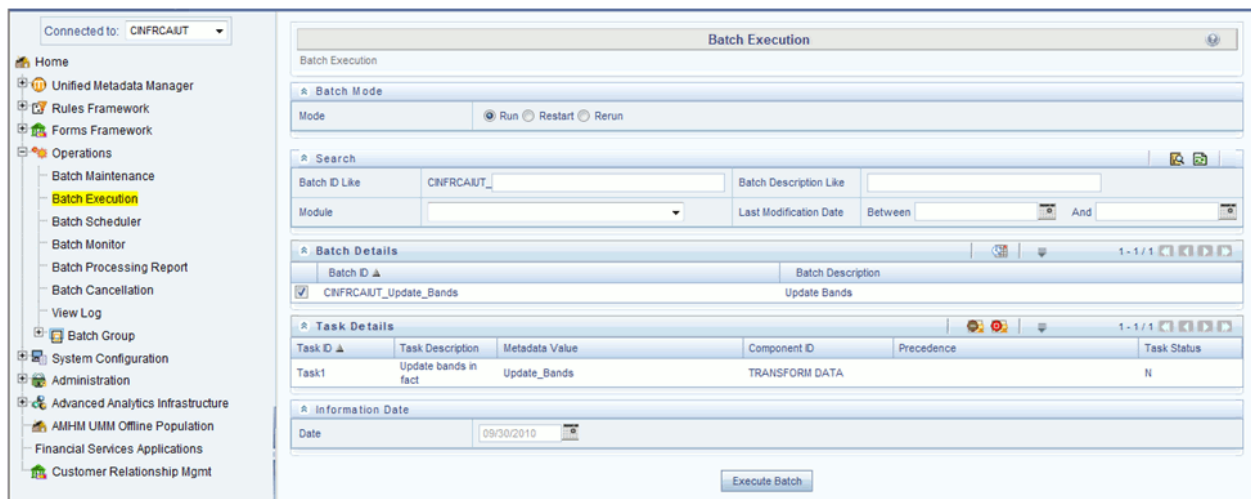
You can also define a new Batch and an underlying Task definition from the Batch Maintenance window of OFSAAI.

For more information on defining a new Batch, see How to Define a Batch.

14 Visibility

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

Visibility has been implemented using two security models:

- OBIEE Security
- Data Security

14.1 OBIEE Security

This has been implemented using the Roles and Privileges settings, the dashboard level, the Report level, and the object level.

The dashboard visibility for the OFS CA application is restricted as per the following table:

Table 40: Dashboard Visibility for OFS CA

Subject Area	Dashboard Group Name	Dashboard Name	Analysis/ Report Name
Customer - Channel Analytics	Customer View	Customer Profitability and Engagement	Top Serviced Customers
Customer - Channel Analytics	Channel Analytics	Service	Customer Complaint and Follow up Action Report
Customer - Channel Analytics	Channel Analytics	Service	Detail Summary of Service Requests

The Reports visibility for the different roles has to be handled by setting proper catalog Permissions.

Those users who have access to any of the above dashboards with PII columns should also be mapped to the 'Data Security Group' in OFSAA using SMS.

If PII entitlements change for a given user; then you need to either clear the cache through OBIEE admin or refresh the report.

14.2 Data Security

This has been implemented with a sequence of tables used for controlling the data access to the user.

The set of tables are:

- **FSI_M_USER**: This table stores all the users who are not relationship managers and are business users who have access to data at different levels. The user id in this table should match the user's login id of OBIEE.

- **FSI_M_USER_MANAER_MAP**: This table stores all the users who are relationship managers. V_User_name should hold the Obiee login Id of the user who is a relationship manager. The Manager Code column should match with the entry in dim_management.

- **FCT_ACCT_MANAGER_REL**: This table restricts the user who is a relationship manager to a certain account of customer/Customer. This defines the user at the lowest granularity.

- **DIM_CUSTOMER**: This table is to define if the user has access to all the accounts the customer holds. This is again to define the relationship manager visibility. This data will be moved from dim_party. Dim_party will be sourced from stg_party_master.

- **FSI_USER_DATA_ACCESS**: This is a mapper table enabled on AAI Mapper that provides UI for the user to set the visibility. The visibility of the user can be set at the following levels using the mapper - Product, Branch, Legal Entity, and Line of Business.

- **FSI_USR_CTRL_ACCESS**: This table contains all the records for each user and the access available to the user for every date. The data is sourced from FSI_M_USER_MANAGER_MAP, FSI_USER_DATA_ACCESS, DIM_MANAGEMENT, FCT_COMMON_ACCOUNT_SUMMARY, FCT_ACCT_MANAGER_REL, and DIM_CUSTOMER. The Parent-Child hierarchies (derived entities) need to be refreshed before this table loads. The names of the hierarchies are MGRPC and CUSTPC. The User has access to all the child nodes in the manager Hierarchy and all the customer hierarchies the user is managing, and the customer hierarchies managed by the child node managers as well.

- **CTRLACC**: This is a materialized view on the table FSI_USR_CTRL_ACCESS giving the distinct user access to accounts, customers, products, line of business, and legal entity. This view is used for applying visibility on the rpd. This is created as a derived entity and there is a job to refresh this derived entity.

NOTE	Users insertion in FSI_M_USER and FSI_M_USER_MANAGER_MAP has to be done directly into the table. For example, in presence of a Single Signon System, these tables need to be loaded with data from a single-sign-on system directly.

15 Marketing Triggers

Marketing Triggers enable you to identify accounts or customers based on the respective dimensions or measure values.

15.1 Creating New Marketing Triggers

To create new marketing triggers, follow these steps:

1. Log into the OBIEE client.

2. Navigate to **New** and select **Segment**.

3. Select **Target Level**. The marketing trigger window is displayed.

Figure 41: Marketing Triggers
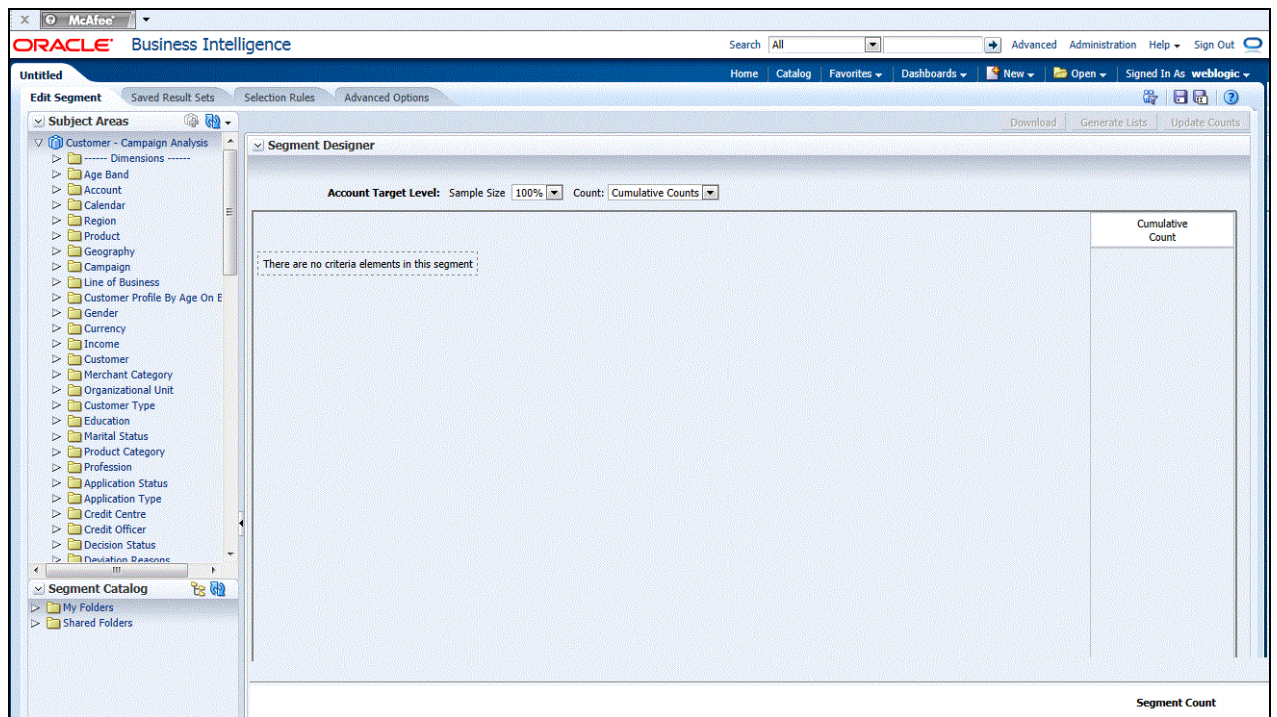
4. Expand the subject area and double-click any entity for which trigger needs to be created.

5. A New Filter window is displayed. Enter the appropriate filter conditions and values.

Figure 42: New Filter
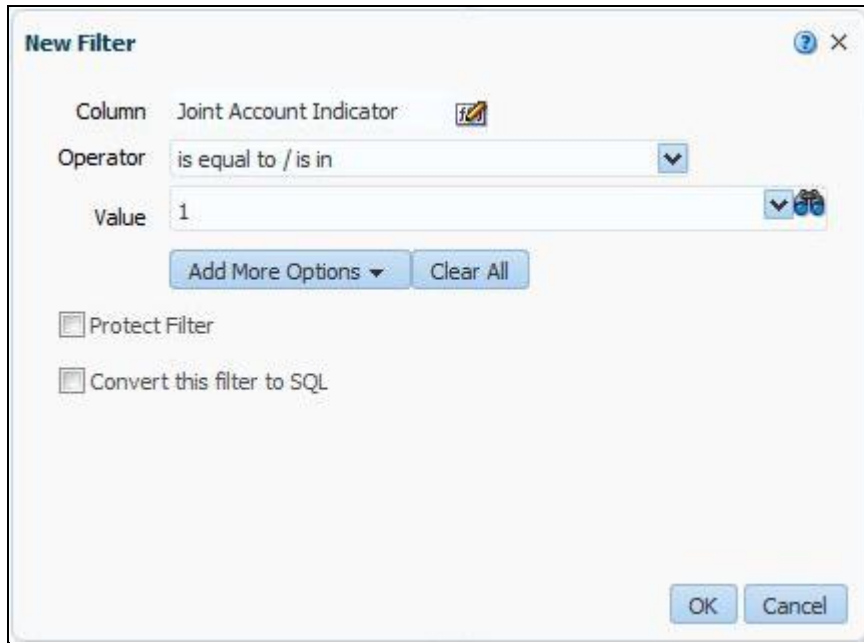
6. Click **OK**. A new filter is added.

Figure 43: New Filter Values
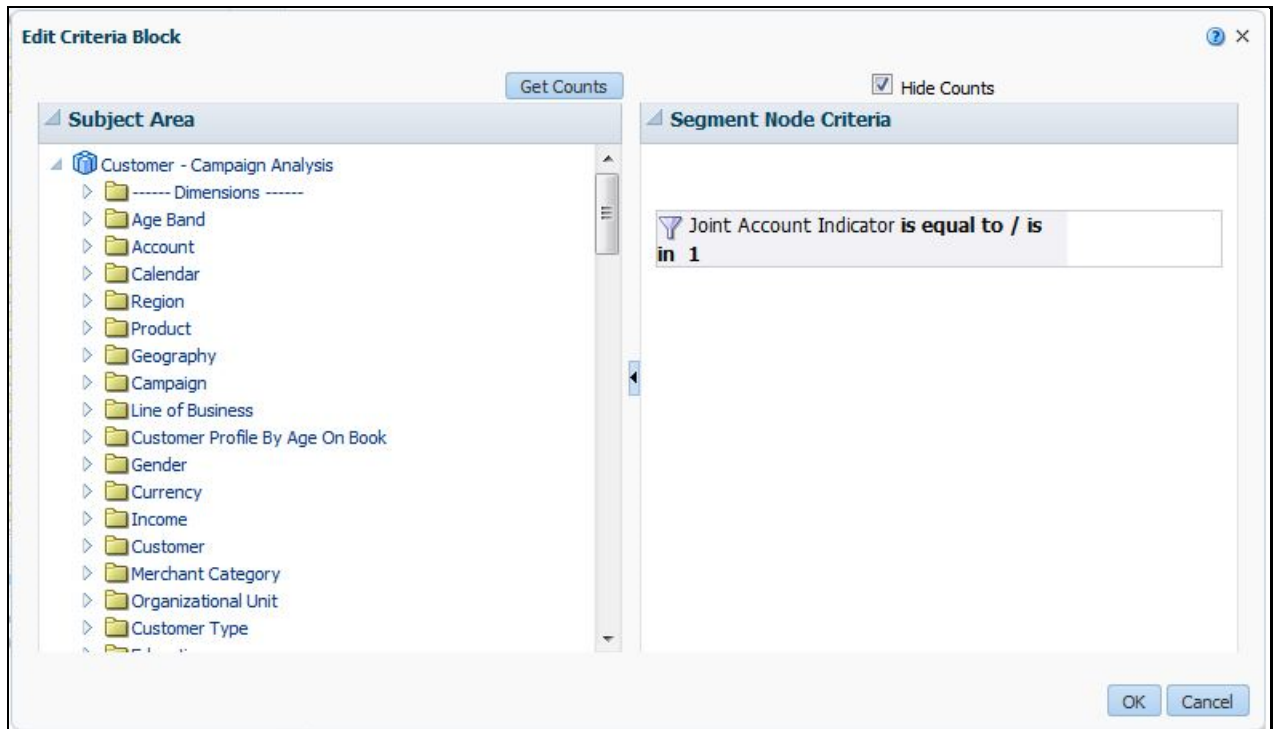
7. Click **Update Count** to update the filter.

Figure 44: Update Counts
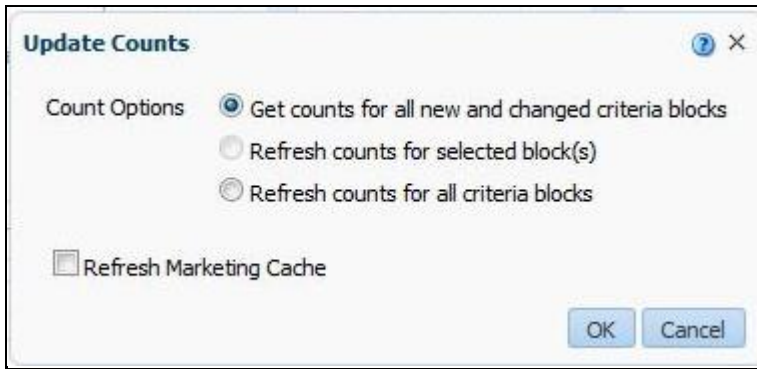
8. Click **OK**. The count will be displayed highlighted as shown in the below screenshot.

Figure 45: Updated Count
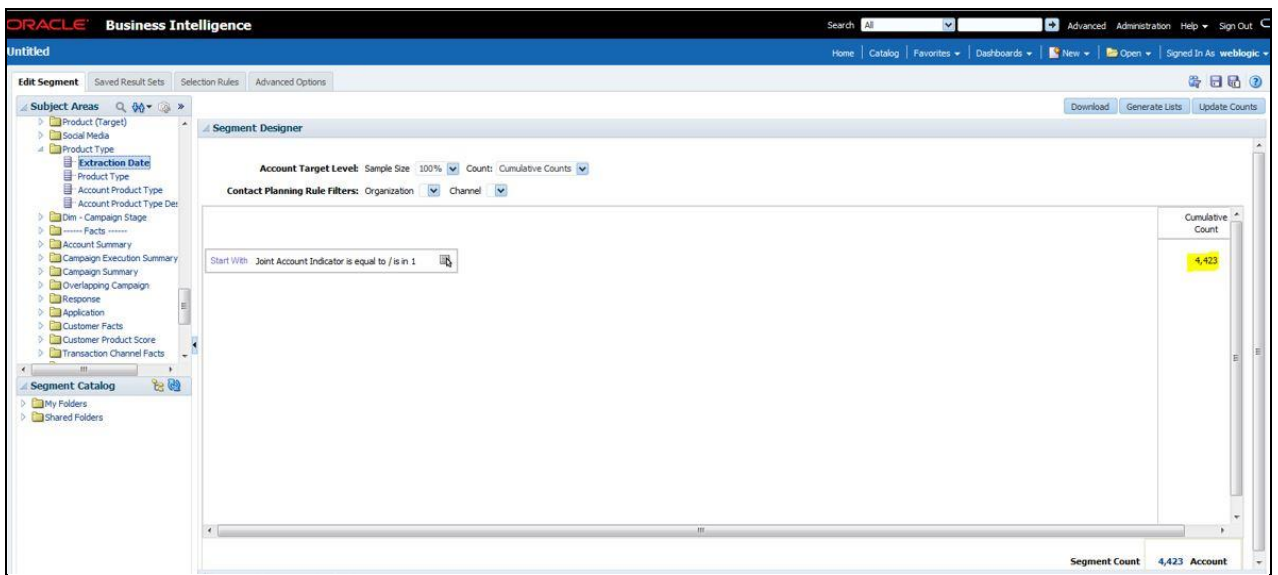
9. To add another filter, double-click any entity for which trigger needs to be created.

Figure 46: Additional Filter
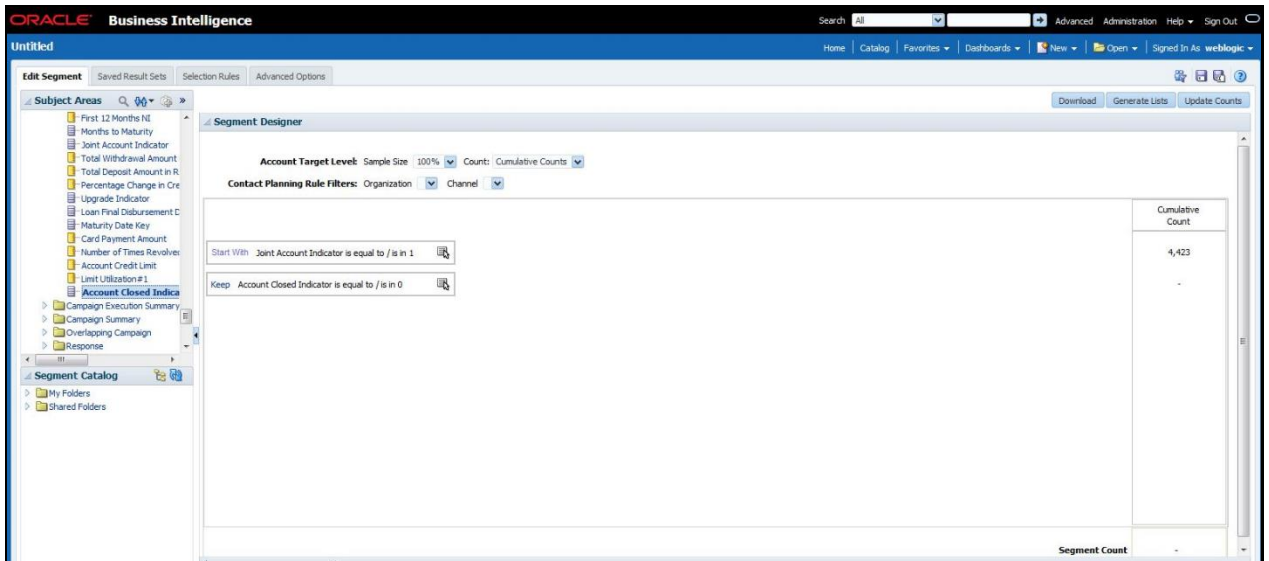
10. Click Update Count to update the filter.

Figure 47: Update Count
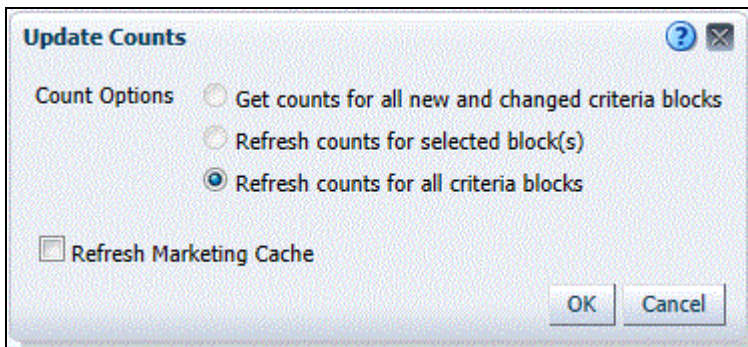
11. Click **OK**. The updated count is displayed.

Figure 48: Updated Count
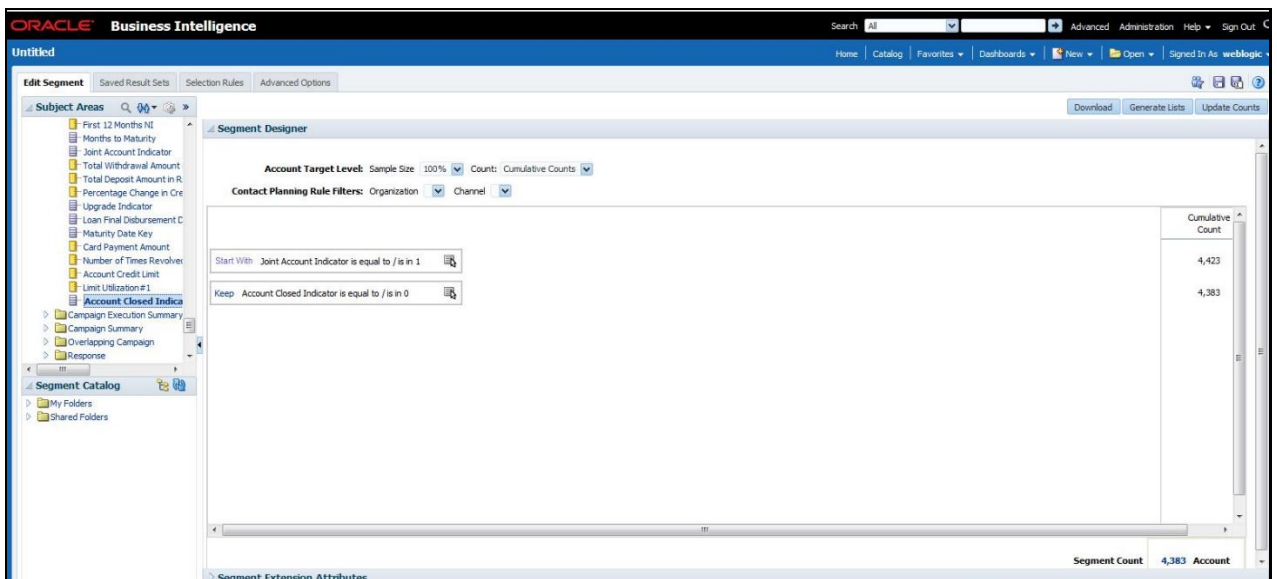
12. Click the **Advanced Options** tab and click **Browse** for the field **List Preview File Format**.

Figure 49: Advanced Options Report
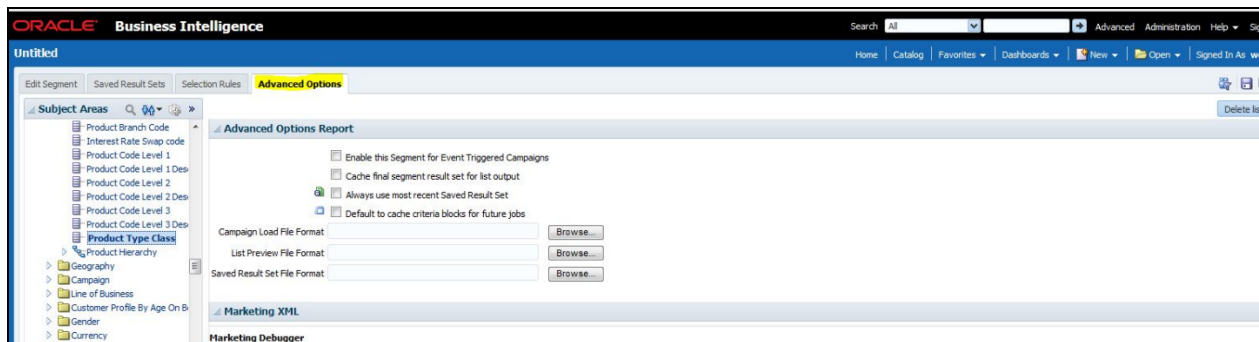
13. Select Account level list format as given in the below screenshot, and then click **OK**.

Figure 50: Account Level List Format
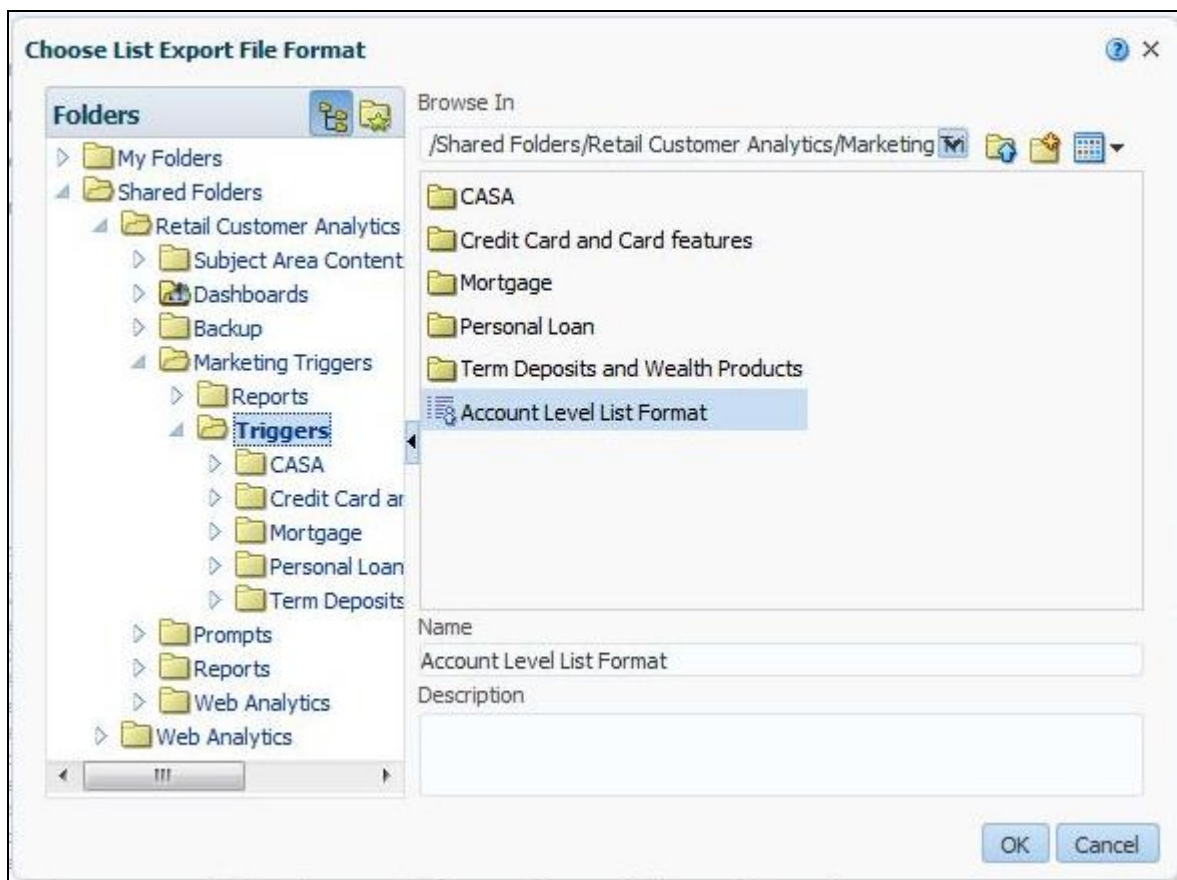
14. Save the list.

Figure 51: Save As
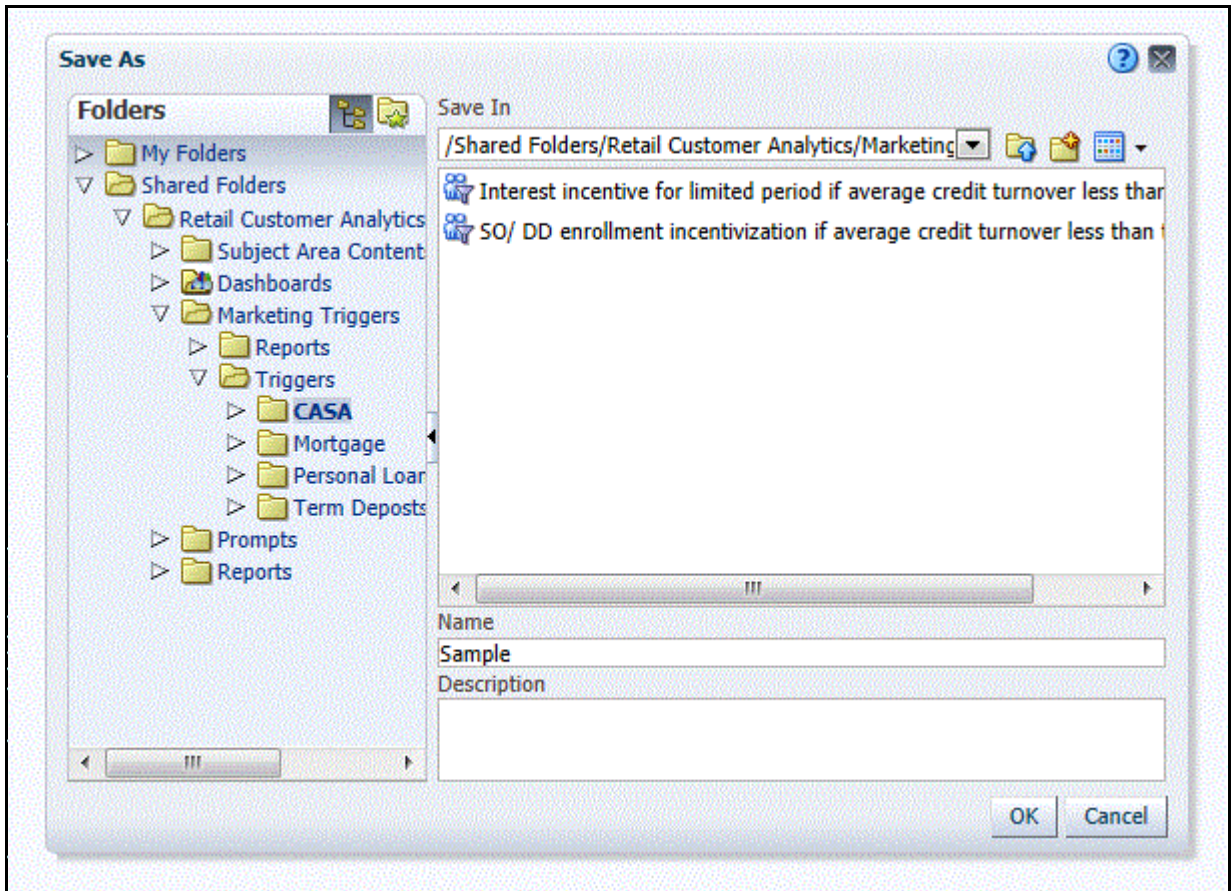
15. In the Edit Segment tab, click **Generate List** to view the filter in list-mode.

Figure 52: Generate Lists
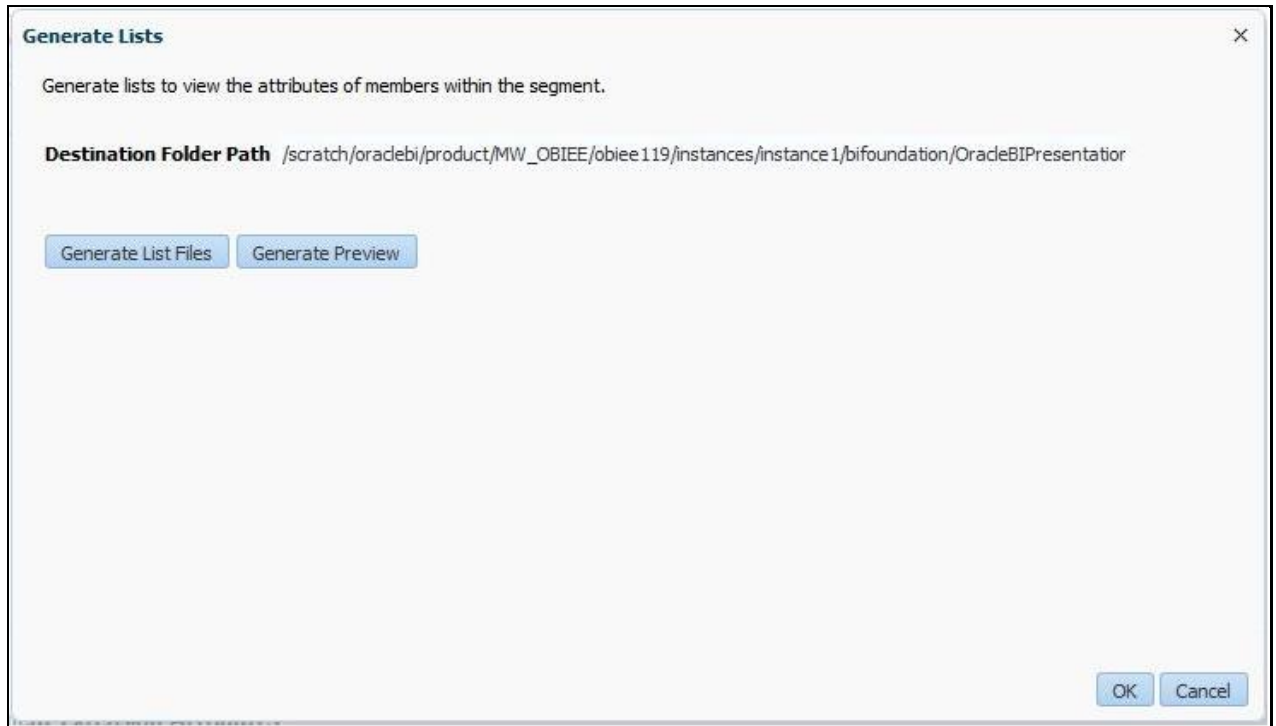
16. Click **Generate List Files** to download the file into a local folder.

Figure 53: Generate Lists

17. Click the path under **File Path** to open the file and then save it.

Figure 54: File Path
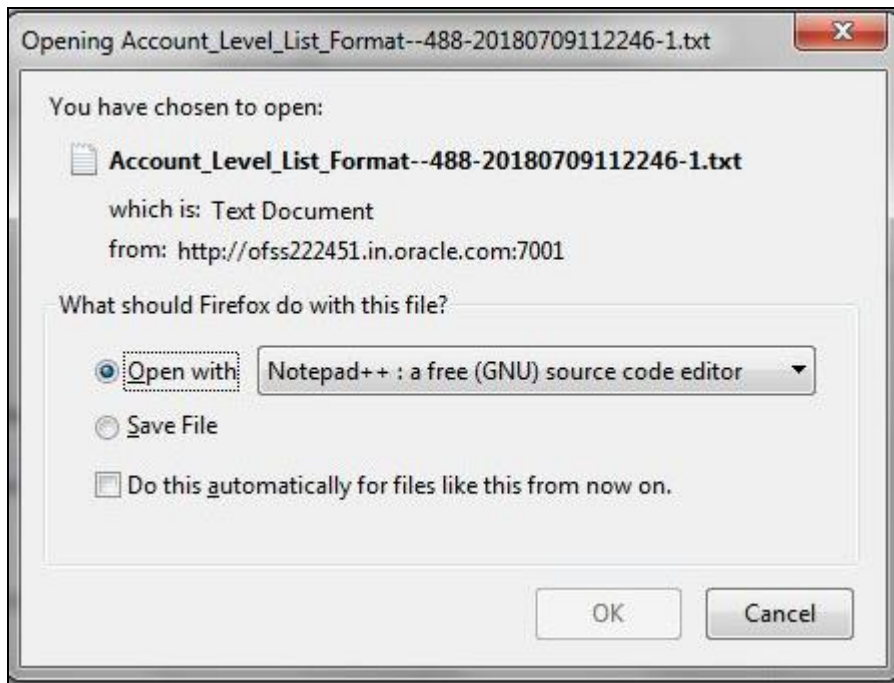
18. Also, these lists will be saved in the server in the following location:

    `$OBIEE_HOME/instances/instance1/bifoundation/OracleBIPresentationServicesC`
    `omponent/coreapplication_obips1/lists`

19. You can copy the path from the generated list and paste it into the server as shown in the following screenshot.

Figure 55: Generate Lists

Figure 56: Server Path
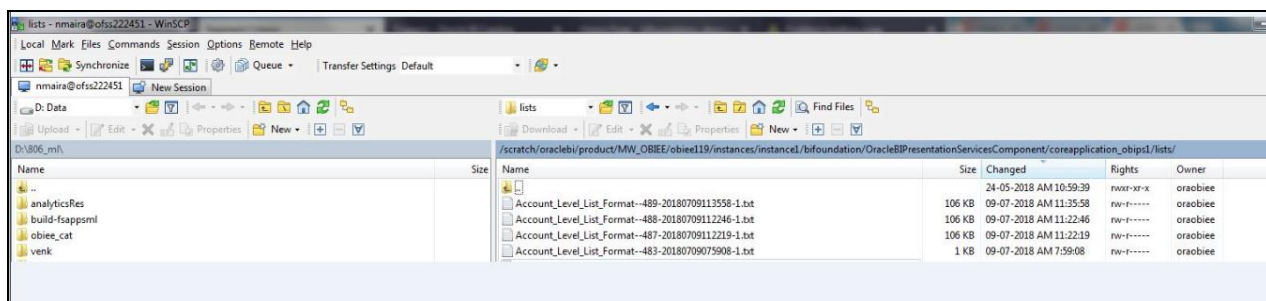
15.2   Viewing Default Marketing Triggers

To view the default Marketing Triggers, click **Catalog** to view default marketing triggers.

Figure 57: Catalog
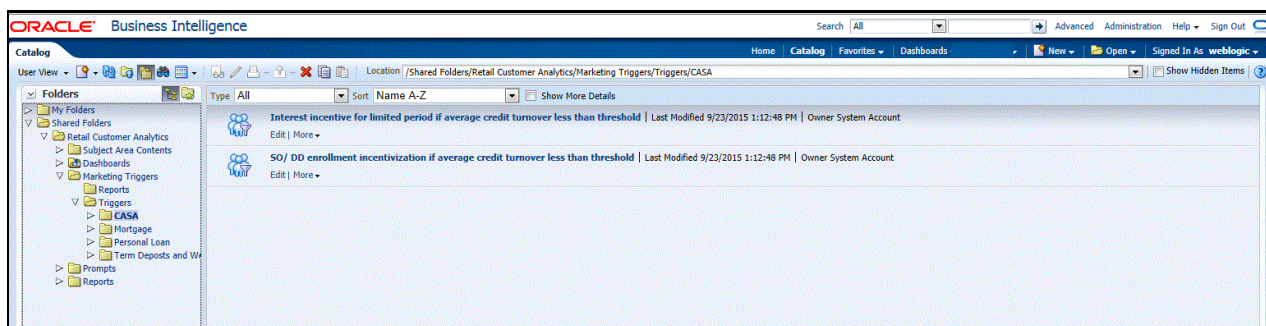
15.3   Using Default Marketing Triggers

OFS RCA allows you to use the default marketing triggers.

To use the default marketing triggers, follow these steps:

1. Navigate to Catalog, select **Shared Folder**, select **Retail Customer Analytics**, and then select **Marketing Triggers**.

2. Click **Triggers**. You can access the triggers for different products here.

Figure 58: Seeded Triggers
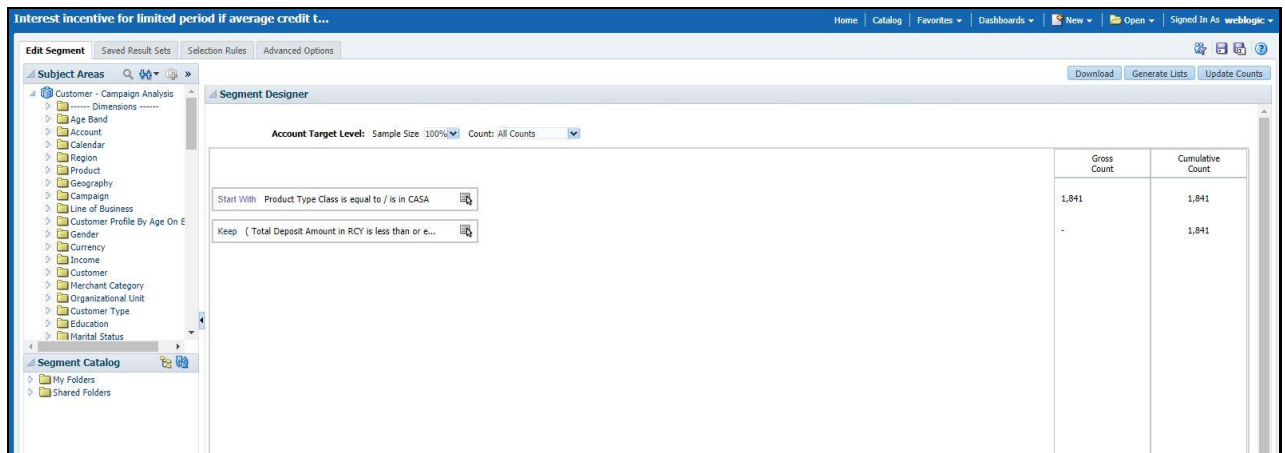
3. You can access directly or edit the Out-of-the-box triggers (tweak).

4. Click **Update Counts** to generate counts.

5. Click **Download** to generate an excel with counts.

6. Click **Generate Lists** to save the file in a local folder.

Out-of-the-box, OFS RCA is released with the following Marketing Triggers:

- CASA
  - Interest incentive for a limited period if average credit turnover is less than the threshold
  - SO/ DD enrollment incentivization if average credit turnover is less than the threshold
  - If the balance is close to the next interest rate tier
  - proposition upgrade of customer

- Credit Card and Card Features
  - Credit or debit card spend in a foreign location (above a certain threshold amount)
  - First time revolver – BT Offer
  - Forex Spend - price incentive on debit\/ credit card
  - Joint Account – Supplementary card
  - Payment of Overlimit Fees – CLI Offer
  - Revolver to transactor behavior in the last 6 months and high Credit score
  - Spends on specified categories like dining or retail department stores or apparel or airlines
  - Low denomination Fuel spends on Credit or Debit Cards

- Mortgage
  - Change in residential or mailing address for the customer during a specified period
  - Recurring investments like SIP maturing in the next month
  - Term deposit balance exceeding the threshold

- Personal Loan
  - 50th Birthday/ Anniversary: Holiday Loan Offer

- EMI payments to competitor Bank

- High card or overdraft utilization

- Home Improvement or furnishing loan on full disbursement of construction linked mortgage loan

- Increase in CASA credit turnover

- Joint Account accompanied by a rise in credit turnover

- Maturing personal loan – Top up loan

- Significant withdrawal from CASA

- Spend in furniture or home improvement stores on credit or debit card

- Term Deposits and Wealth Products

  - 21st Birthday: Recurring Deposit for Mortgage deposit offer

  - Fixed Deposit Maturity in next 3 months

  - Opening of Kid Account: Recurring Deposit offer

  - Significant deposit into CASA or matured deposit in CASA

  - Tax filing date approaching

16   Social Media and Influencer Score

Word of mouth marketing does a lot of business without your knowledge. This type of marketing promotes and sells your products and services through other people, in other words, influencers who can affect your buying habits because of their real or perceived knowledge, position, or relationship.

Influencer Score can help in a couple of ways: It helps identify social media authors that might become your brand ambassadors or product endorsers. For example, bloggers who post about your bank's products and services on social media.

This functionality considers the customer's social media behavior in the social media websites namely Facebook, LinkedIn, Google+, and Twitter, and computes the Influencer score. The influencer score will be calculated as a number rating, a higher number indicating a higher rating. Influencer score is calculated on the following aspects:

- Calculated based on the behavior on the social media in last 90 days. This period is configurable and can be defined at your bank.

- Tracks and gathers the social media data from Facebook, Twitter, Google+, and LinkedIn for the calculation of the score. This data includes the following elements:

- Tweets/ retweets/ shares/ comments/ depth of comments, etc.

- Amount of quality content created - Number of posts, quality of posts (measured by shares/ retweets)/ tweets)

- Higher the number of topics, the higher the influence.

17 Next Best Offer

This feature enables the bank to offer the customer the best offer based on various criteria. This service consists of the following components:

- NBO request-response using a restful web-service
- NBO process that consists of the back-end implementation for NBO request-response endpoint
- NBO process will follow OFSAA optimized collaborative filtering
- OFSAA optimized cluster algorithm is used as a seeded technique for Customer segmentation to find customer peer set.
- Customer and peer set web browsing pattern is used in conjunction with product holding to assign weights to rank preferred NBO offers against the customer in a segment through batch processing.
- Filtering is applied to prevent offers of products that are already owned by the customer.

17.1 NBO Web Service Description

The URL to access the NBO web service:

```
http://<<host>>:<<port>>/<<web context>>/rest-api/rca/nbo/
```

17.1.1 Request

The following is the JSON web service Request for NBO.

```
{"reqkey": "1",
"customerCode": "OBIB4C5",
"accountNumber": "OBIBG1A016",
"nboinfodom":"OFSCAINFO",
"nbodepth": "1",
"pageKeyPayload": ["1004"]
}
```

The Request header details for the JSON web service are as follows:

Table 41: Request Header Details

Header	Value
authorization	Basic authorization (<Created username and password>)
content-type	application/json
userid	<Created User>

17.1.2   Expected Response

The expected response for the Request JSON is as follows:

```
{
    "NBOPayload": [
      "22"
    ],
    "reqkey": 35,
    "processComments": "#>#Infodom Added as OFSCAINFO#>#Process call
    information retrieval success"
}
```

NBO Request description:

The following are the descriptions for the NBO Request:

- Reqkey: Only 10 digit numeric value accepted.

- customerCode: Accepts 20 digit alphanumeric values.

- accountNumber: Accepts 20 digit alphanumeric values.

- Nboinfodom: Accepts 20 digit alphanumeric values.

- Nbodepth: Only numeric value accepted.

- pageKeyPayload: Accepts alphanumeric values.

17.1.3   Error Response

The following are the NBO Response errors:

If JSON parameter is not proper:

```
{
    "reqkey": 0,
    "processComments": "<Parameter_Name> is not in correct format."
}
```

If default infodom is not set and infodom name is not provided as part of the request:

```
{
    "reqkey": 36,
    "processComments": "#>#[WARNING]Infodom was not part of request,default
    Infodom Added as #>#Infodom Added as INFDM#>#[ERROR] Process call
    failed.#>#sqle.getMessage()"
}
```

18   Random Forest

Random forest, which is a Machine Learning technique has been introduced as an additional method to predict customer propensity for cross-selling and attrition, where weblog data is present in addition to the existing customer data for processing the output. In the case of high dimensionality and size of data, Random forest is a highly recommended algorithm for enhanced accuracy.

19 Boxed Models

Before applying the Release 8.1.1.0.2 one-off patch, upgrade the versions of ORDS and APEX to the following versions:

- ORDS: 21.1.3.153.1102
- APEX: 21.1

19.1 Post-installation Steps

After applying the one-off patch Release v.8.1.1.0.2, perform the following steps.

19.1.1 Give Object Privileges

Connect to the ATOMIC schema and provide the grants to the CONFIG user on the respective tables. Update the `ofsaa_bxmodel_atm_grants.sql` file available at the location `FIC_HOME/dumps/apps/` and execute in the ATOMIC schema.

19.1.2 Create Views in CONFIG Schema

The following tables are in the atomic schema. Create views for those tables in config schema so that we can access these tables' information in config schema.

Update the `ofsaa_bxmodel_views.sql` file available at the location `FIC_HOME/dumps/apps/` and execute on the CONFIG schema.

19.1.3 Re-importing the APEX Application

To reimport the APEX application, follow these steps:

1. Login to OFSAAWS_APEX workspace as OFSAAWS_ADMIN (admin user).
2. Click on App Builder.
3. Click on Import.
4. Select the new file from $FIC_HOME/dumps/apps.
5. Click next with default selections.
6. Click Next.
7. Select build status as **Run Application** for production environment else, **Run and Build Application**.

Figure 59: Install Database Application
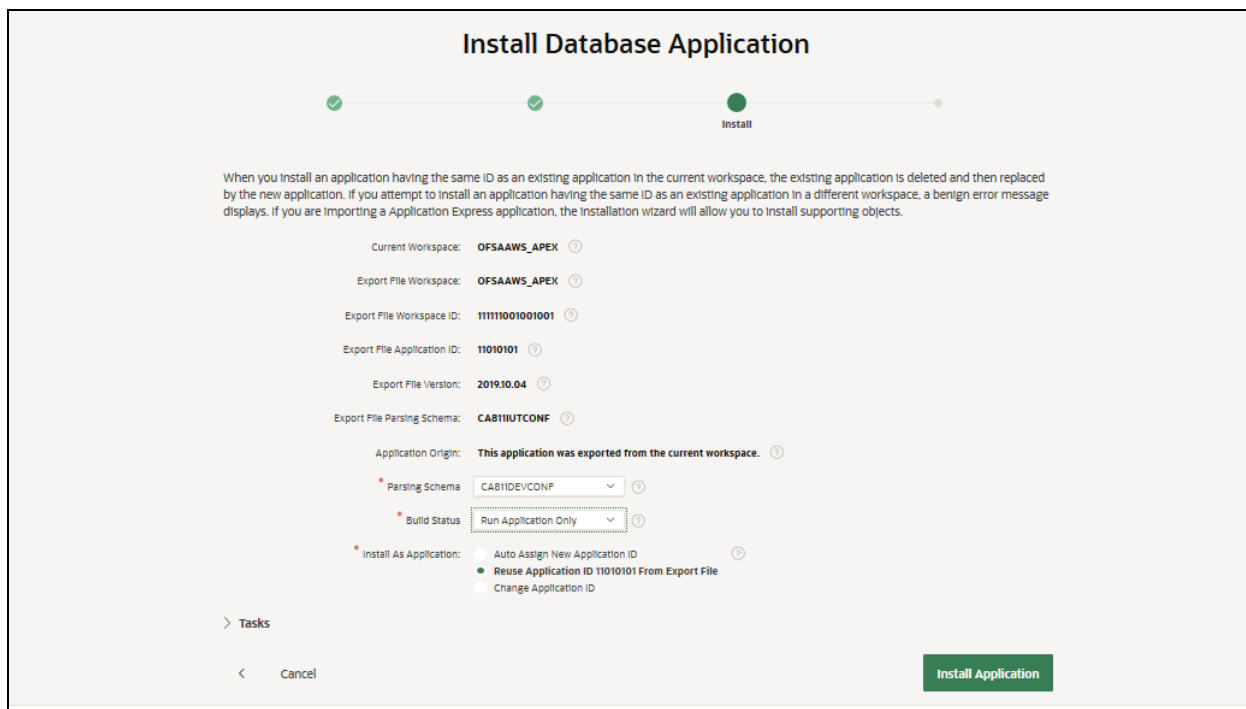
8. Select Reuse Application ID.

9. Click **Install Application**.

19.2   Boxed Model Management

The Propensity and Attrition models follow a boxed approach. The boxed models can be launched from the LHS menu. The user interface allows the model administrator in the bank to make some minor changes to the models based on applicability.

To launch the Boxed Models, from the LHS menu, select **Financial Services Retail Customer Analytics**, select **Model Management**, and then select **Boxed Models**.

This launches the **Boxed Model Management** screen in another browser window.

Figure 60: Boxed Model Management

The Boxed Model Management screen lists all the Propensity models, Attrition models, and Cross-Sell models. This is a View Only screen and does not allow you to add new models or delete existing models. However, you can view and edit the existing models.

Click the **Run Summary** button to open Model Output screen. This screen displays the Model Output in terms of the following plots, milestones, and metrics as applicable:

- Outliers

- Model Variable Distribution

- Variables Corelation Map

- Paiplot

- Countplot

- Dependable Variable Distribution

- Metric Density Plot

- Confusion Matrix

- Significant Variables

You can select the **Model MIS Date** for the **Prediction** that you want to see.
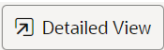
You can click on the [Detailed View] button to see an enlarged view of the selected plot or milestone or metric.

Figure 61: Model Plots Tab
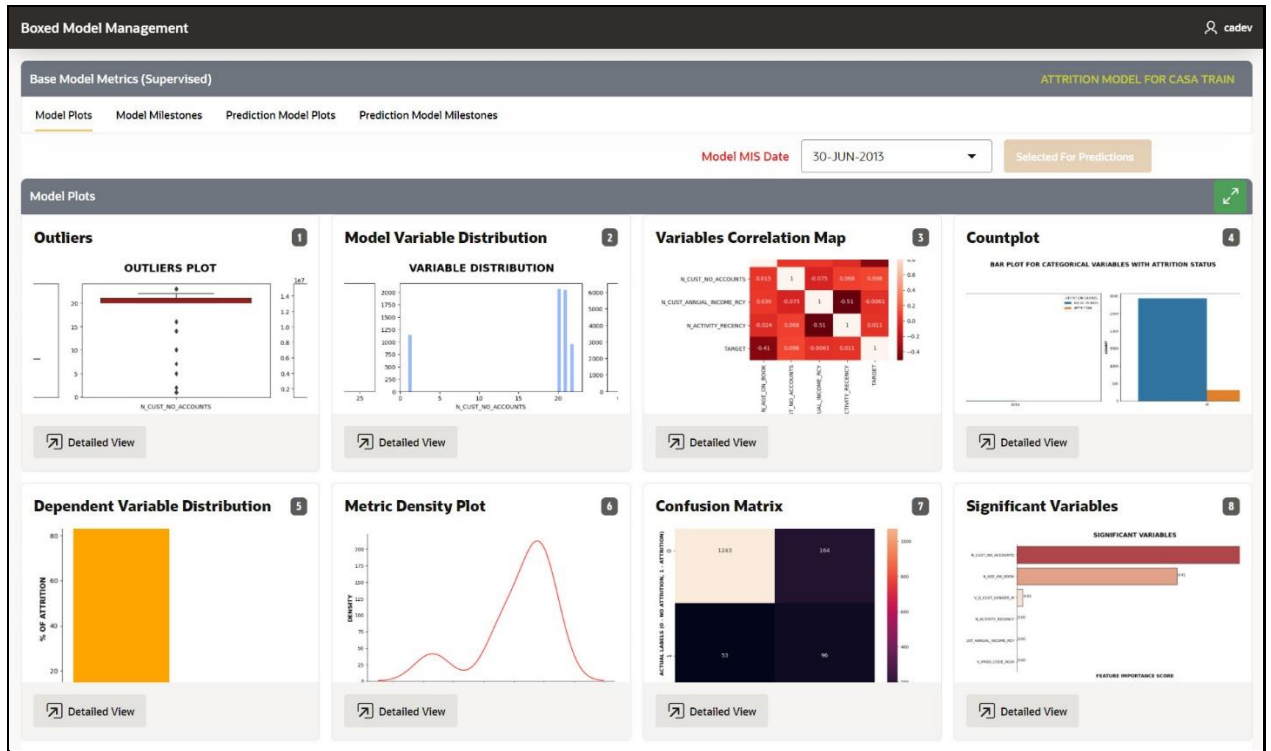
19.3   Editing the Model

To edit the model, follow these steps:

Click the ✎ icon to edit the parameters for the selected model.

These models follow a screen flow with different widgets in a train as follows:

1. **Model Metadata**: This widget displays the information related to the selected model. This screen is View Only screen.

Figure 62: Model Metadata widget

2. **Model Data Source Definitions**: This widget displays the model data source details. You can click the ✎ icon to edit the Data Source Entry for the selected model. When you click the ✎ icon, the Edit window opens where you can change the **Data Source Entry** and Description. Click the **Save** button to save the details. The table name that you enter for Data Source Entry must be in the Oracle standard table name format.

Figure 63: Edit window

You can navigate to the Model Data Variables widget by clicking the Next >> or clicking the ↗ icon.

3. **Model Data Variables**: This screen displays the data variables for the selected model. You can edit the select and de-select the variables and add new variables to the model.

Figure 64: Model Data Variables widget

a. Click **Add New Variable** to add a new variable.

Figure 65: Edit window

b. Enter the following details (all fields with * are mandatory):

— Data Source Code

— Variable Code (must be in Oracle standard format and must consist only alphabetics and _)

— Variable Type

— Variable Expression (Not NULL)

— Description

c. Click **Save** to save the details.

You can see the kay feature values that are integral to the model by clicking the **KEYFEATURE Values** button. The Key Feature Variables are View Only parameters.

a. You can navigate to Model Run Time Parameters widget by clicking the Next >> button.

b. **Unmapping and Mapping a Model**: When you select a model and click the Unmap button, the model will not be considered for the modelling. This will be listed at the bottom of the widget page under the **Inactive Data Source** section. If you want the un-mapped model to be considered for modelling, you can select the model from the Inactive Data Source list and click the Map button. The model will be listed again under the main list of models and will be considered for modelling.

4. **Model Run Time Parameters**: The Model Run Time Parameters window allows you to tune the model parameters. This widget displays the Data Parameters, Other Model Parameters, and Random Forest Parameters.

Figure 66: Model Run Time Parameters widget

a. Click the ✎ icon adjacent to the Product Type List items displayed to open the Edit window.

Figure 67: Edit window

b. Enter the following parameters:

— Parameter Label

— Parameter Values

— Description

c. Click **Save** to save the details.

You can also change the Description and Parameter Value for the parameters.

d. You can navigate to the Environment Runtime Parameters widget by clicking the Next >> button.

5. **Environment Runtime Parameters**: This widget displays the model log parameters. You can edit and change the log parameters.

Figure 68: Model Environment Runtime Parameters widget

a. Click the ✎ icon adjacent to the Parameter Label to open the Edit window.

Figure 69: Edit window

b. Enter the following parameters:
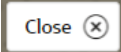
— Parameter Label

— Parameter Value

— Description

c. Click **Save** to save the details.

6. Click the button to close the window and navigate back to the Model Management Summary screen.

20   Appendixes

20.1   Requesting and Authorizing to Populate Sandbox

This option is not available for the logical sandbox.

To request and authorize to populate sandbox in the Sandbox Maintenance window, follow these steps:

1. Select the sandbox which you want to populate and click the **Edit** button in the Sandbox Maintenance toolbar. The Edit button is disabled if you have selected multiple checkboxes. The Sandbox Maintenance Edit window is displayed.

2. In the Request Action tab, select **Complete for Populate Sandbox** to copy the required table data from the Production infodom to the Sandbox infodom based on the sandbox definition.

3. Click the Authorize tab, and select the **Populate Sandbox - Complete/ Incremental** checkbox to authorize sandbox population. This tab will be enabled only if your user role is mapped to the function SANDBXAUTH.

4. Click **Save** to confirm changes.

After authorization, a Sandbox-Populate batch is registered in the OFSAA Infrastructure Operations. The batch will be available in the Batch Scheduling window with the Sandbox ID. This batch must be triggered from the Batch Scheduling window to complete the data population.

20.2   How to 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 the 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.

To create a batch, follow these steps:

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

2. In the Batch Maintenance window, select the '+' button from the Batch Name toolbar. The New Batch Definition window is displayed.

3. Enter the Batch details shown in the following table.

Table 42: Batch Details

Field	Description
Batch Name	The Batch Name is auto-generated by the system. You can edit to specify a Batch name based on the following conditions: • The Batch name must be unique across the Information Domain. • The Batch Name must be alpha-numeric and should not start with a number. • The Batch Name must not exceed 41 characters in length. • The Batch Name must not contain special characters "." and "-".
Batch Description	Enter a description for the Batch based on the Batch Name.
Duplicate Batch	(Optional) Select the checkbox to create a new Batch by duplicating the existing Batch details. On selection, the Batch ID field is enabled.
Batch ID (If duplicate Batch is selected)	It is mandatory to specify the Batch ID if 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.

20.3   Weblog Processing for RCA

Big data, in the form of weblog data, is used in RCA to enhance the predictive power of the attrition and propensity models. The big data functionality is an option within RCA and is included to analyze weblogs resulting from click-stream data, which are logs of user activity on a bank's website and internet banking.

> **NOTE**   The propensity and attrition models enhanced with weblog variables have an option to run either or RDBMS or HDFS.

The following steps provide a high-level description of weblog processing:

1. Weblogs are captured from all the weblogs available in the MIS folder and moved to the staging table in HDFS using the L2H (log-to-hive) process.

2. Weblogs are moved from the staging table through a series of fact tables to the final fact table in HDFS using the H2H (hive-to-hive) process.

3. Weblogs are moved from the final fact table in HDFS to RDBMS using the H2T (hive-to-table) process.

The data from RDBMS is calibrated using the propensity and attrition models.

20.3.1   Validating a Data Model Generated from a Log File

To generate meaningful data from weblog files, you need to validate a data model generated from the weblog file. Following are the steps to generate and validate the data model using the Source Model Generation (SMG) utility:

1. Create a weblog base folder and generate a source data model.

Figure 70: Source Data Model Generation
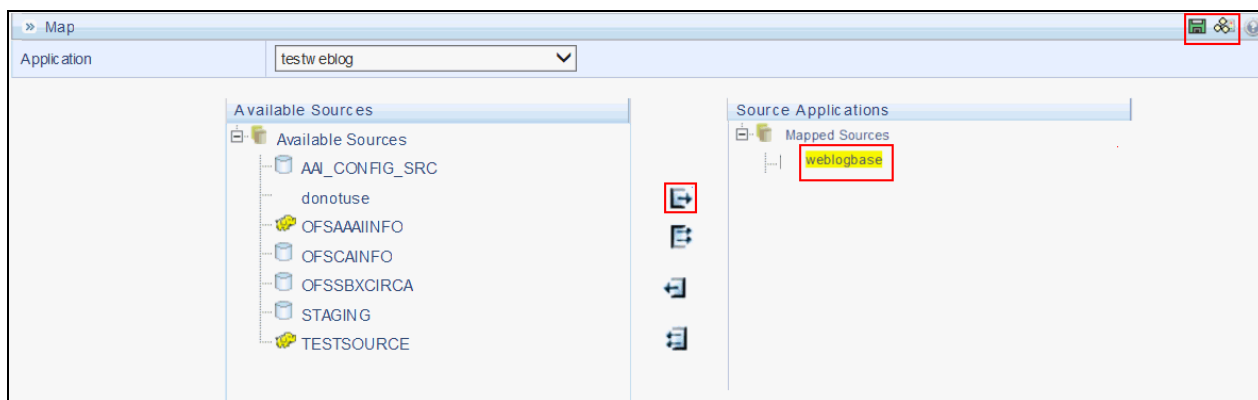
2. In the SMG utility, select a sample weblog file from the weblog base folder. Once the file is read, the utility maps the file to a known weblog type. The following weblog types are supported:

- Apache

- Nginx

- Microsoft-IIS

- LiteSpeed

- Google Servers

- Custom

> **NOTE** Select the custom option only if the data is delimited. You can then specify the delimiter by selecting the **Delimited** check box.

Figure 71: Sample Weblog File Selection
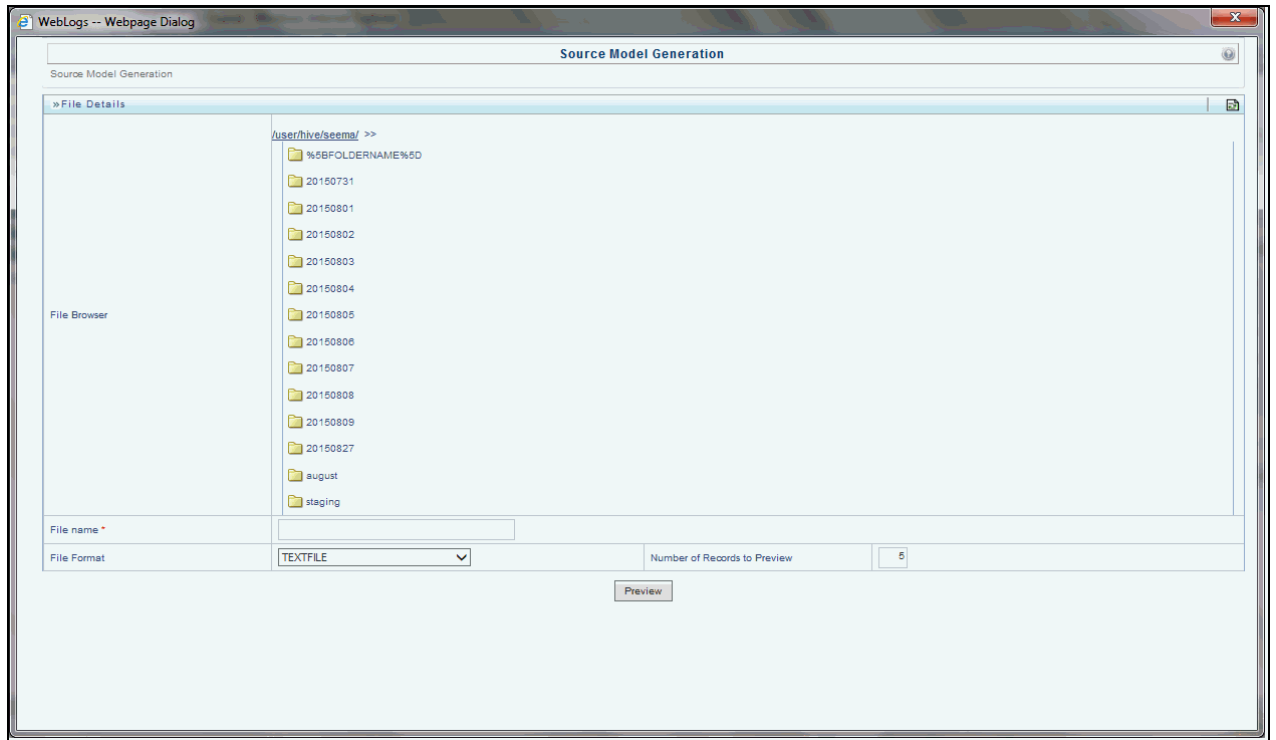
| NOTE | If the source folder is Hadoop-based, ensure that the weblog file is present in the Hadoop folder before generating the source model. |

Figure 72: Data Model Generation
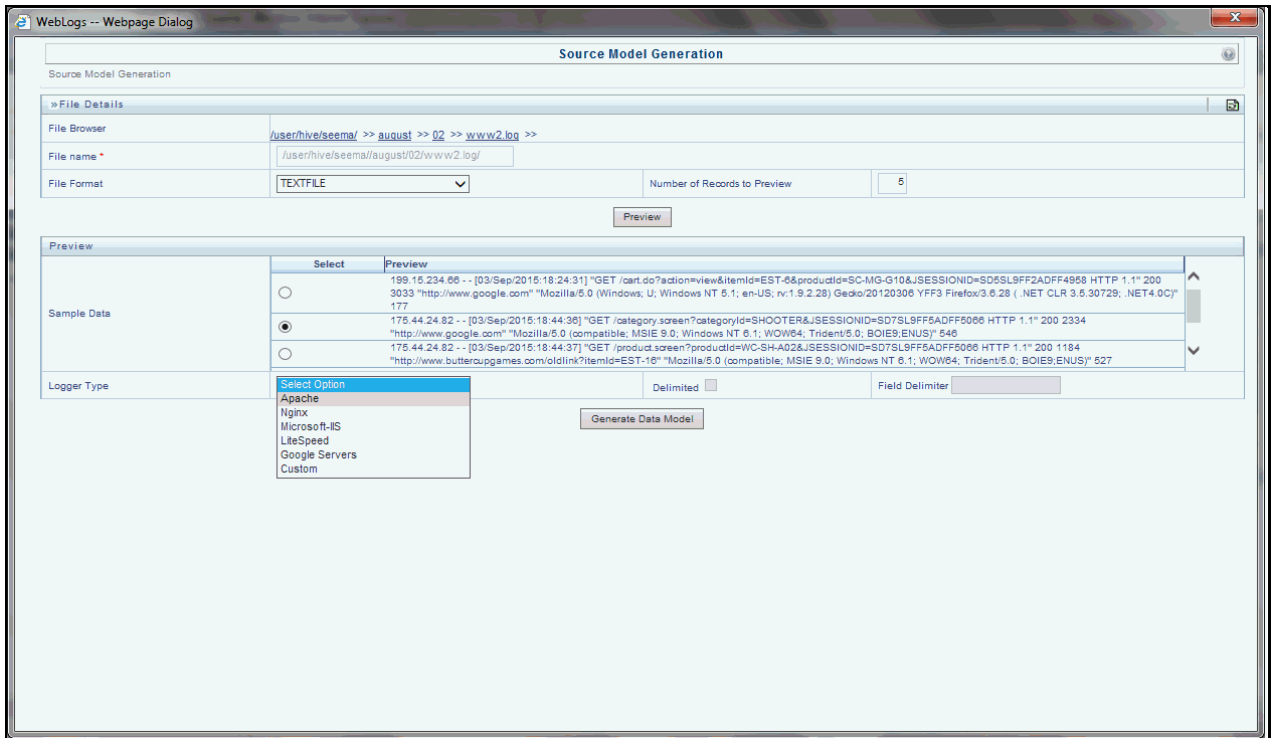
You can also select the number of records that you want to use to validate the data model from the weblog file.

3. Generate a preview of the data model by clicking **Generate Data Model**. You can edit the column names and data types only for weblogs that are not of the custom type.

Figure 73: Data Model Validation
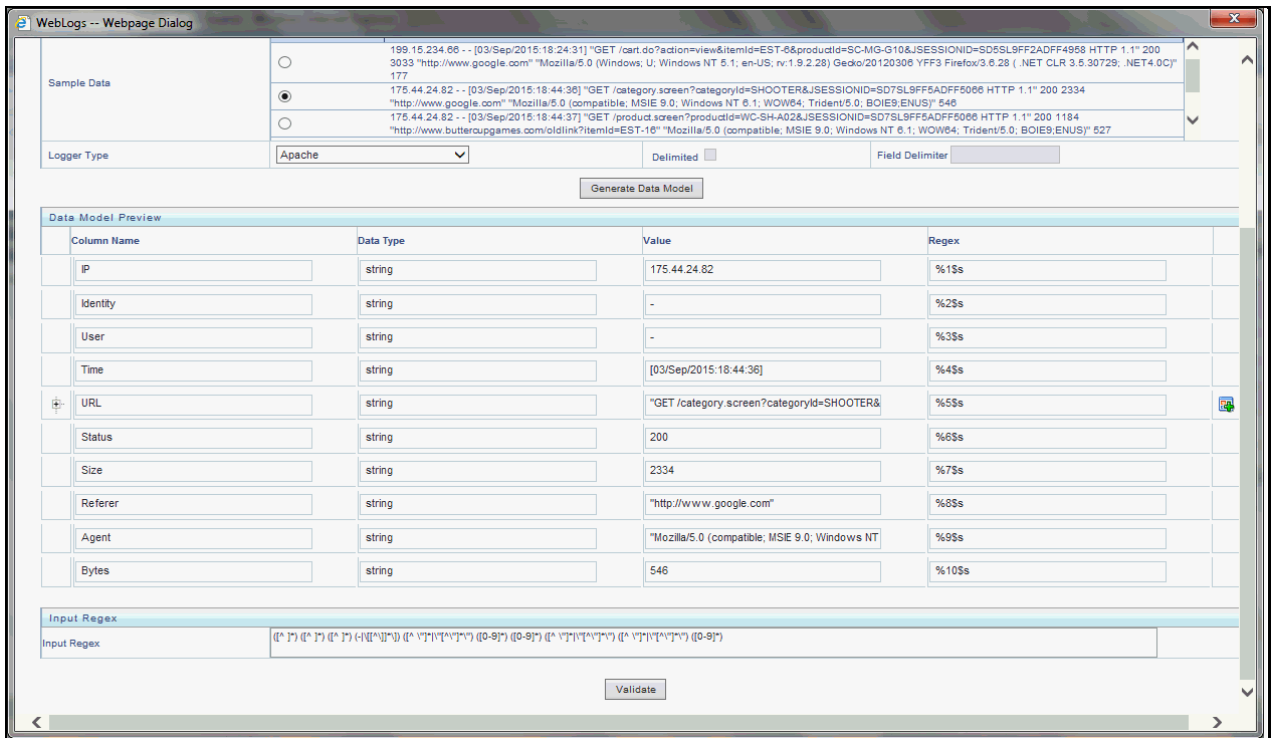
4. Validate the data model by clicking **Validate**. The selected records are validated against the model, and the results are displayed in the **Data Validation** section.

Figure 74: Data Validation Records

| | NOTE | You can save the data model even if some records cannot be validated, although this will lead to incorrect data being displayed. In this case, you can make necessary changes to the input expressions and re-validate the data model. |

20.3.2   Customer Identification

Customers can be identified only if they log on to the net banking portal of the bank's website. When a customer logs on from a web browser or a device, a device cookie is created.

Following are the various scenarios involved in customer identification:

- If the customer logs on through a device, the resultant cookie contains the customer's user ID. The user ID is then mapped to the account ID, which is then mapped to the customer ID.

- If the customer has not logged on through a device but the device cookie contains the user ID, the customer is mapped to the existing user ID and the MIS date, which is then mapped to the customer ID.

- If the customer has not logged on and there is no information in the device cookie, searches are done hierarchically for matching identifiers such as email IDs, mobile numbers, and social network IDs. If the searches do not provide any results, the customer is mapped to a random customer ID.

- To capture identifiers, the bank's weblog capture tool is expected to capture weblogs as shown in the following example:

| | NOTE | The data provided must be included in the weblog as space-delimited in the same order. |

Table 43: Sample Apache Valve Configuration

Attribute	Description	Sample Apache Valve Configuration
Client IP	Remote IP address	%a
Server IP	Local IP address	%A
Response size without header	Bytes sent, excluding HTTP headers, or '-' if zero	%b
Response size with header	Bytes sent, excluding HTTP headers	%B
user name in a cookie	User name provided with a cookie with name 'name'	%{name}c
Process Duration millis	Time is taken to process the request, in millis	%D
Client hostname	Remote hostname (or IP address if enableLookups for the connector is false)	%h
Request protocol	Request protocol	%H
User-Agent	Incoming header with name User-Agent	%{User-Agent}i
username identd	Remote logical username from identd	%l
Request method	Request method (GET, POST, and so on)	%m
Outgoing header	The outgoing header for the server	%{Server}o
HTTP/s port	The local port on which this request was received	%p
URI*	Method and request URI	%r
HTTP status	HTTP status code of the response	%s
Request timestamp	Date and time, in Common Log Format [26/Feb/2016:11:26:14 +0530]	%t
Process Duration seconds	Time is taken to process the request, in seconds	%T
username authenticated	A remote user that was authenticated (if any), else '-'	%u
URL path	Requested URL path	%U
Server name	Local server name	%v

Attribute	Description	Sample Apache Valve Configuration
session ID	User session ID	%S
uid	ServletRequest attribute for uid	%{uid}r
Domain name	Domain name as string	in.oracle.com

- The following attributes are needed in the query string:

Table 44: Attributes for Query String

Attribute	Query string key
Facebook Id	Facebook
Twitter Id	Twitter
Google Plus Id	Google Plus
Phone Number	PhoneNo
Email Id	email

- Based on Apache tomcat 8.0.18, the following valve configuration needs to be done in the server.xml file so the seeded definitions can process the log data successfully:

```
<Valve className="org.apache.catalina.valves.AccessLogValve"
directory="logs"

               prefix="localhost_access_log" suffix=".txt"

               pattern="%a %A %b %B %{name}c %D %h %H %{User-Agent}i %l %m
%{Server}o %p %r %s %t %T %u %U %v %S %{uid}r in.oracle.com"/>
```

```
An L2H process is mentioned in the following example as a sample:
```

```
10.184.227.76 10.184.151.91 - 0 sroy 2 10.184.227.76 HTTP/1.1 Mozilla/5.0
(Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0 - GET - 8080
GET
/CaptureLog/success.html?Facebook=roysourav91&Twitter=roysourav91&GooglePl
us=roy.sourav9 HTTP/1.1 304 [26/Feb/2016:11:26:16 +0530] 0.002 -
/CaptureLog/success.html 10.184.151.91 71DB9426663439ED028E77AFAAC79F3A -
in.oracle.com
```

```
10.184.227.76 10.184.151.91 703 703 sroy 4 10.184.227.76 HTTP/1.1
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0 -
GET - 8080 GET /CaptureLog/logout.jsp HTTP/1.1 200 [26/Feb/2016:11:26:16
+0530] 0.004 - /CaptureLog/logout.jsp 10.184.151.91 - - in.oracle.com
```

OFSAA Support

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to the 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 My 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 or recently released documents.