

# **OFS Performance Analytics Cloud Service**

## **Administration Guide**

**Release 9.0.0.0.0**

**Feb 2020**

**ORACLE**  
Financial Services

## OFS Performance Analytics Cloud Service

Copyright © 2020 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

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

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

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

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

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

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

## Document Control

Version Number	Revision Date	Change Log
9.0	24 January 2020	First release

# Table of Contents

<b>1</b>	<b>Preface.....</b>	<b>6</b>
1.1	Intended Audience.....	6
1.2	Documentation Accessibility.....	6
<b>2</b>	<b>Cube Build Process.....</b>	<b>7</b>
2.1	List of Cubes in IPA and RPA Applications.....	7
2.2	Creating Configuration Files.....	8
2.3	Building Cubes.....	9
2.3.1	<i>Prerequisites.....</i>	9
2.3.2	<i>Tables Used by the Cube Build Component .....</i>	10
2.3.3	<i>Executing the Cube Build Task.....</i>	11
2.3.4	<i>Aggregate Data Task.....</i>	11
2.3.5	<i>Create Cube Task.....</i>	11
2.4	Checking the Execution Status .....	15
<b>3</b>	<b>What-If Analysis.....</b>	<b>17</b>
3.1.1	<i>Basic Scenario .....</i>	18
3.1.2	<i>Advanced Scenario.....</i>	20
3.2	Configuration for What-If Analysis.....	22
3.3	PACS Integration with Price Creation and Discovery (PCD) .....	27
3.4	Reporting Line Correlation Calculations.....	27
3.5	What-If Analysis Limitation .....	29
<b>4</b>	<b>Segmentation and Rules .....</b>	<b>30</b>
4.1	Segmentation in IPA.....	30
4.2	Segmentation in RPA .....	31
4.3	Creating a Rule.....	32
4.4	Editing a Rule .....	34
<b>5</b>	<b>Defining a Batch.....</b>	<b>39</b>
<b>6</b>	<b>Adding a New Dimension .....</b>	<b>40</b>
6.1	Dimension Definition Process.....	40

6.1.1	Steps to follow while using ESSBASE Source for Relationship Manager Hierarchy .....	43
6.2	Metadata .....	45
6.2.1	Technical Metadata .....	45
6.2.2	Optional Metadata .....	45
6.2.3	Business Metadata .....	46
6.2.4	Reporting Metadata .....	46
<b>7</b>	<b>Adding a New Measure.....</b>	<b>47</b>
7.1	Measure Definition Process.....	47
7.2	Build Cube.....	48
<b>8</b>	<b>Developing a New Cube.....</b>	<b>49</b>
8.1	Procedures to Develop a New Cube .....	49
<b>9</b>	<b>Run Rule Framework .....</b>	<b>50</b>
9.1	Executing a seeded run .....	50
9.2	Runs Available.....	52
<b>10</b>	<b>Loading Multiple Load Runs in OFSAA .....</b>	<b>54</b>
10.1	Features.....	54
10.2	Design Details .....	54
10.3	Data Transformations .....	55
10.3.1	Execution .....	56
10.3.2	Execution .....	56
10.3.3	Execution .....	57
<b>11</b>	<b>Steps to Upload Files to SFTP Server Shared Mount .....</b>	<b>62</b>

# **1 Preface**

## **1.1 Intended Audience**

Welcome to Release 9.0.0.0.0 of the Oracle Financial Services Performance Analytics Cloud Service Operations Guide.

This guide is intended for the users of Oracle Financial Services Performance Analytics Cloud Service. See Related Information Sources for more Oracle product information.

## **1.2 Documentation Accessibility**

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at:

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit:

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

## 2 Cube Build Process

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

This chapter discusses the following topics:

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

### 2.1 List of Cubes in IPA and RPA Applications

OFS IPA application is seeded with the following cubes:

Cube	Purpose	Dataset
Institutional Analysis	The purpose of this cube is to provide analysis of various Account related measures across dimensions like Product, Line of Business, Vintage, and so on.	This cube is based on the FCT_COMMON_ACCOUNT_SUMMARY, FCT_CRM_ACCOUNT_SUMMARY, FCT_COMMON_CUSTOMER_SUMMARY, and FCT_CRM_CUSTOMER_SUMMARY fact tables.
RM P and L Cube	The purpose of this cube is to provide details of Profit and Loss statement of a Relationship Manager across dimensions like Line of Business, Product, Organizational Unit, and so on.	This cube is based on the FCT_ACCOUNT_PROFITABILITY and FCT_ACCOUNT_MGR_REL fact tables.

OFS RPA application is seeded with the following cubes:

Cube	Purpose	Dataset
Retail Analysis	The purpose of this cube is to provide analysis of various Account related measures across dimensions like Product, Line of Business, Vintage, and so on.	This cube is based on the FCT_COMMON_ACCOUNT_SUMMARY and FCT_CRM_ACCOUNT_SUMMARY fact tables.

Cube	Purpose	Dataset
Customer Summary	The purpose of this cube is to provide analysis of various Customer related measures across customer profile dimensions like Income Band, Age, and so on.	This cube is based on the FCT_COMMON_CUSTOMER_SUMMARY and FCT_CRM_CUSTOMER_SUMMARY fact tables.
Cards Balance Summary	The purpose of this cube is to provide analysis of various Cards Balance Category related measures across dimensions like Time, Customer Profile by Income, Age, Gender Age on Book, Vintage, and so on.	This cube is based on the FCT_CARDS_BALANCE_SUMMARY fact table.
Account Profitability	The purpose of this cube is to provide analysis of Financial Reporting Lines related measures across dimensions like Time, Line of Business, Customer Profile by Income, Age, Gender Age on Book, Vintage, and so on.	This cube is based on the FCT_ACCOUNT_PROFITABILITY, FCT_COMMON_ACCOUNT_SUMMARY, FCT_CRM_ACCOUNT_SUMMARY, FCT_COMMON_CUSTOMER_SUMMARY, and Fct_CRM_Customer_SUMMARY fact tables.
RM PnL Cube	The purpose of this cube is to provide details of Profit and Loss statement of a Relationship Manager across dimensions like Line of Business, Product, Organizational Unit, and so on.	This cube is based on the FCT_ACCOUNT_PROFITAIBILITY and FCT_ACCOUNT_MGR_REL fact tables.

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

## 2.2 Creating Configuration Files

Each cube has a configuration file that contains the details of dimensions and measures which are part of the cube. Essbase outline is created using the configuration file. Configuration files for seeded cubes are available as part of the installer. However, if there are any changes to cube definition then configuration files are recreated during saving of the cube definition.

Follow these steps:

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



**NOTE**

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

## 2.3 Building Cubes

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

- Generating an aggregate DATA file containing the measure values for each dimension leaf that are part of the cube definition. This is performed by the AGGREGATE DATA component task within the batch definition.
- Creating the cube outline on Essbase server. This is performed by the CREATE CUBE component task within the batch definition.
- Loading the data to the cube. This is performed by the CREATE CUBE task within the batch definition

### 2.3.1 Prerequisites

Following are the prerequisites for creating a cube:

1. All the post install steps mentioned in the OFSAA Infrastructure installation guide and Solution installation manual have been completed successfully.
2. Parentage files need to be created for BI hierarchies after dimension data is loaded. 'Resave Metadata' process is used to create the parentage files.
3. OFSAAI application user needs to have the required functions mapped to the user for doing Resave Metadata and accessing the Home> Unified Metadata Manager > Business Metadata Management screens and executing a batch from Application batch operations screen
4. Execute Save Metadata by navigating to the following screen on the OFSAAI framework LHS Menu.
  - Go to **Home > Administration > Save Metadata**.
  - Choose all the available metadata under Hierarchy and move it to the right by using the '>>' button.
  - Click **Save** and might take a few minutes for the saving to complete.
  - Click **Show Details** to view the log for the Save operation.
5. Ensure that the following services are running on the application server before doing a cube build:
  - Iccserver
  - Router
  - AM
  - Messageserver

- Olapdataserver
6. Batches need to be created for executing, which is explained in the Executing the Cube build section.
  7. All the required tables for dataset need to be populated before you execute the cube batches, such as Dimension Population, Time Dimension population, Account Summary Population and Fact Ledger Population.
  8. The dataset for the cube should return some rows in the database for the cube build to happen. To check the same, perform the following steps:
    - a. Navigate to **Home > Unified Metadata Manager > Business Metadata Management > Data Sets**.
    - b. Click **Search**.
    - c. Click any dataset in the pop up which opens and click **Ok** to return to the data set screen.
    - d. Click the button on right of ANSI Join text box. Enter the required expression or click the below button to define an expression using the Expression screen.
    - e. Click **OK** to return to the data set screen.
  9. Perform the same for Join/Filter Condition and Date filter.
  10. Frame a SQL query like this:
 

```
Select count(1) from <Enter the part you obtained from Ansi join
part above>where<Enter the part you obtained from Join/Filter
Condition & Date filterparts>
```

This query should show record count greater than zero when you fire this from SQL prompt in the database.
  11. Essbase Reserved Keywords: Essbase has few reserved keywords and 'Missing' is one of them. So in case of cube building using dim\_bands, it fails as it tries to create a cube with the reserved keyword in the table data. None of the reserved keywords should be used while creating data.

**NOTE**

Essbase does not identify/ support newline character. Ensure that you do not use them.

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

[https://docs.oracle.com/cd/E51367\\_01/financialsop\\_gs/FASVU/apbs02s02s03.html](https://docs.oracle.com/cd/E51367_01/financialsop_gs/FASVU/apbs02s02s03.html)

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

## 2.3.2 Tables Used by the Cube Build Component

Tables that are part of the dataset need to be populated before executing the cube build component. In addition, REV\_BIHIER table in atomic database schema stores the hierarchy data for Business

Intelligence-enabled hierarchies for cube build. This table gets populated when a hierarchy is saved using Save Metadata screen.

### 2.3.3 Executing the Cube Build Task

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

### 2.3.4 Aggregate Data Task

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

### 2.3.5 Create Cube Task

1. In the batch created in Aggregate Data task above, click New Task ('+' symbol in Task Details container).
2. Enter the Task ID and Description.
3. In the Component drop down, choose Create Cube.
4. Select the following from the Dynamic Parameters List and then click Save:
  - Datastore Type: Select the appropriate datastore from the list.
  - Datastore Name: Select the appropriate name from the list.
  - IP address: Select the IP address from the list.
  - Cube Parameter: Choose the cube code to be built from the drop down list.
  - Operation: Choose All from the drop down list.

## 5. Execute the batch created in the above step.

### NOTE

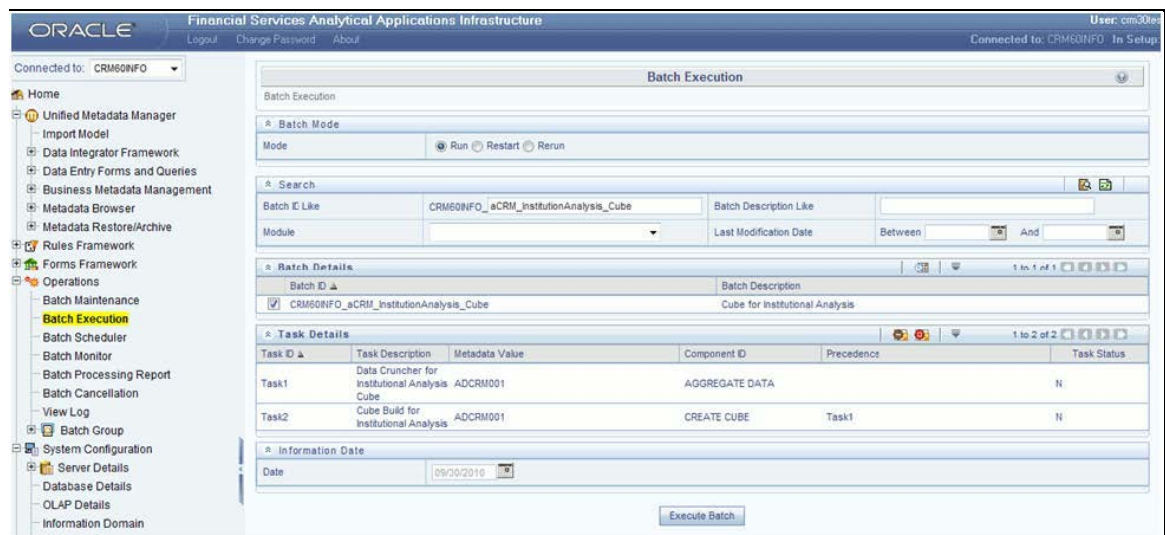
A common issue in the Aggregate task is Data Set not having records for which the steps mentioned in the prerequisites have to followed or the SQL query in Data Cruncher log file has to be checked on the database (Location of log file mentioned in the 'Checking the Execution Status' section below). In the Create Cube task one common error is the hierarchy member being the same for two different dimensions which are part of the same cube (Error message: 'Duplicate Alias' in the Create Cube log file). In this case, you can try appending a string to the Hierarchy member code so that it is unique across the cube or changing the hierarchy data to make the node unique across the cube.

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

### 2.3.5.1 IPA Cubes

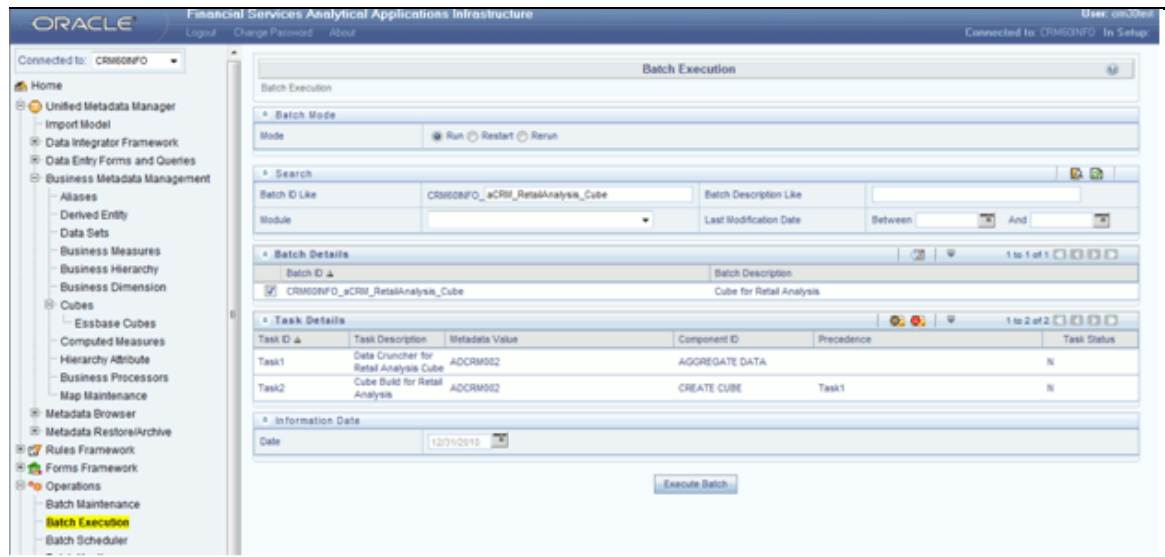
- **Institutional Analysis:** Seeded batch <INFODOM\_aCRM\_InstitutionAnalysis\_Cube is provided with the installer. Execute the batch for the required MIS Date.
- **RM P and L Cube:** Seeded batch <INFODOM\_ReIn\_Mgr\_Cube> is provided with the installer. Execute the batch for the required MIS Date.

•

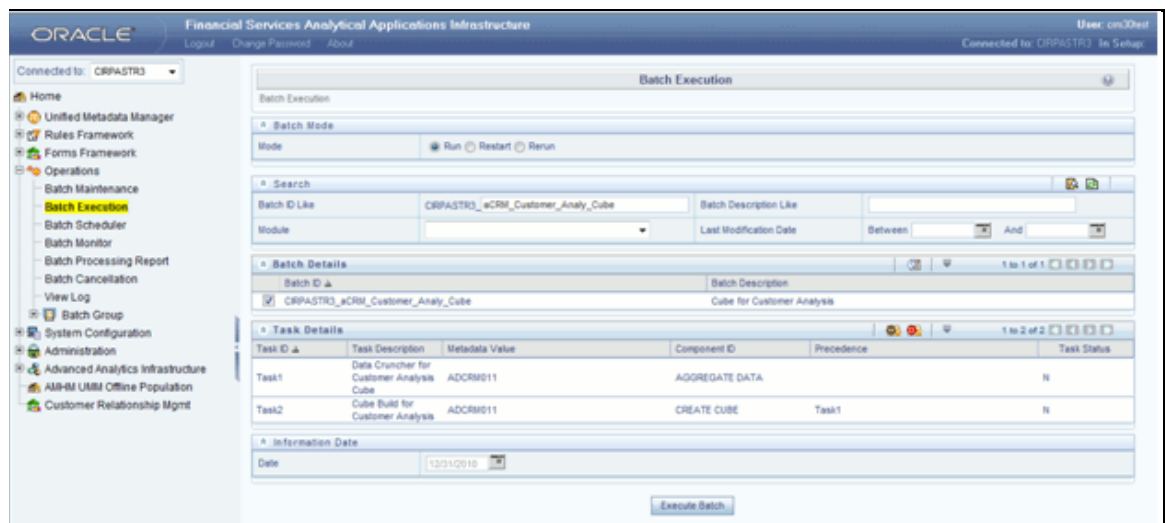


### 2.3.5.2 RPA Cubes

- **Retail Analysis:** Seeded batch <INFODOM>\_aCRM\_RetailAnalysis\_Cube is provided with the installer. Execute the batch for the required MIS Date.



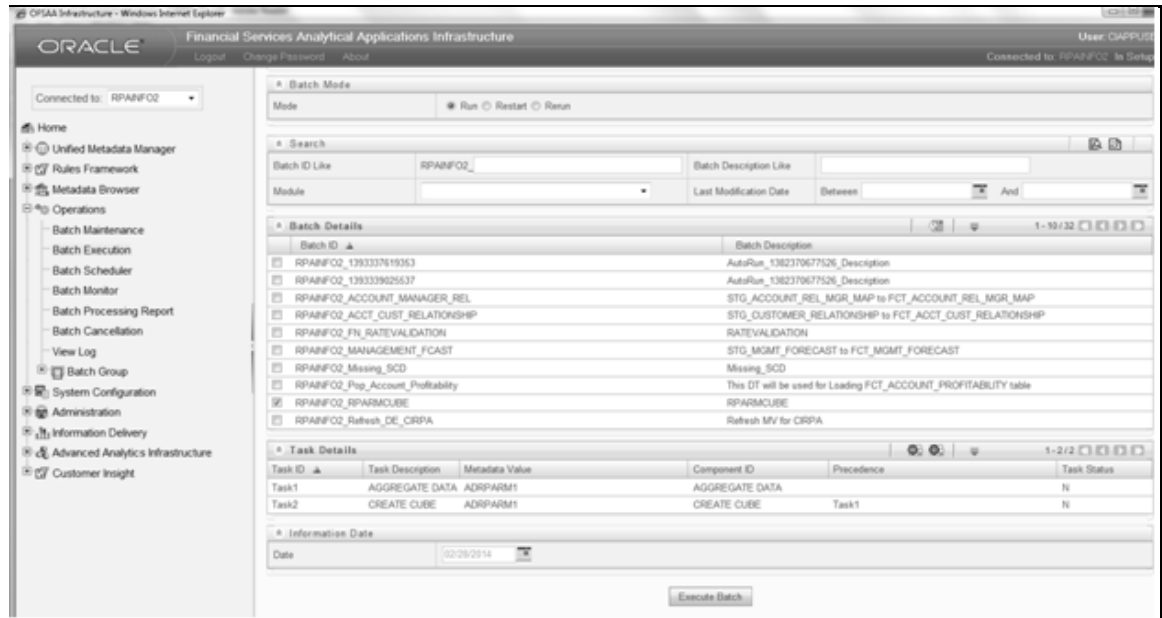
- **Customer Summary:** Seeded batch <INFODOM>\_aCRM\_Customer\_Analy\_Cube is provided with the installer. Execute the batch for the required MIS Date.



- **Cards Balance Summary:** Seeded batch <INFODOM>\_aCRM\_Cards\_Balance\_Summ\_Cube is provided with the installer. Execute the batch for the required MIS Date.

- **Account Profitability:** Seeded batch <INFODOM>\_aCRM\_Account\_Proft\_Cube is provided with the installer. Execute the batch for the required MIS Date.

- **RM PnL Cube for RPA:** Seeded batch <INFODOM>\_RPARMCUBE is provided with the installer. Execute the batch for the required MIS Date.



## 2.4 Checking the Execution Status

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

### NOTE

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

The status messages in Batch Monitor are:

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

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

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

**NOTE**

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



### 3 What-If Analysis

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

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

User could define the variations through the UI, which when imposed on the income statement provide the resulting net income. The resulting income statement post applying the variations is called as a 'Scenario'. The projections are by default created for a period of 5 years, but the change in projection may not necessarily be applied for the entire 5 years. The tenure for which the specified variation is applicable can also be defined while specifying the variation. The magnitude of variation being applied can be specified either in 'percentage' or 'absolute'. If the variation is specified as percentage then the value of the component to which variation is being applied changes by the corresponding percentage value for the specified time. Similarly, when variation is applied in absolute terms the value of the component to which variation is being applied changes by the corresponding absolute value for the specified time.

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

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

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

Income Statement Variation												
Time run: 3/28/2016 12:20:16 PM												
Amount in Millions (USD)												
	2013		2014		2015		2016		2017		2018	
	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement	Projected Movement	Revised Movement
Net Income Before Taxes	-246.27		-330.95		-333.85		-336.55		-339.09		-85.13	

The following input parameters are applicable for OFAPACS:

V_PARAMETER_NAME	V_PARAMETER_VALUE
VAR_STRT_DT_IPA	This input parameter indicates the start date for the variance calculation.
VAR_END_DT_IPA	This input parameter indicates the end date for the variance calculation.

V_PARAMETER_NAME	V_PARAMETER_VALUE
TSHLD_FCTR_IPA	This input parameter indicates the threshold factor for replines.
EXCL_LMT_IPA	This input parameter indicates the maximum outlier exclusion percentage.

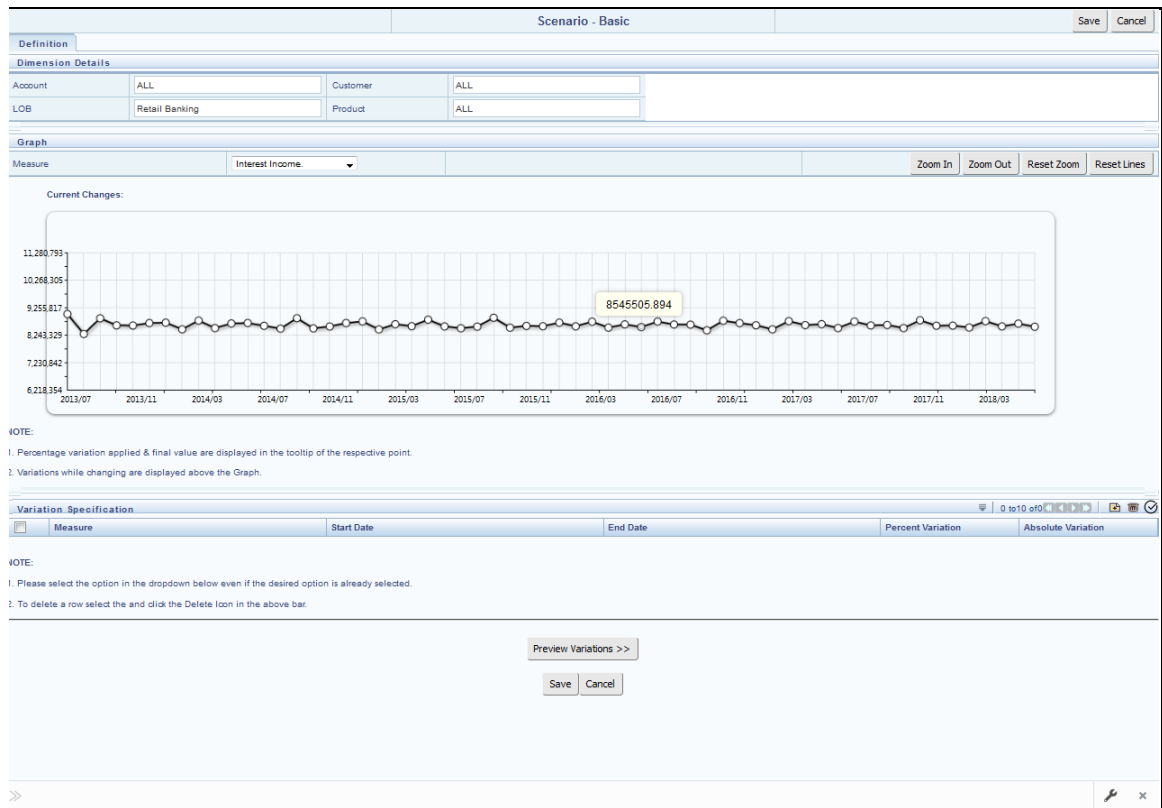
**NOTE**

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

### 3.1.1 Basic Scenario

The following procedure describes the steps to create a variation:

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



8. Apply the necessary **Dimension Details** for the following:
  - Account
  - Customer
  - LOB
  - Product
9. Select the relevant repline **Measure** from the dropdown list to which you want to apply the variation.
10. Select a point on the graph and drag to apply the desired variations. Percentage variation and applied, final values in the dropdown are displayed on the graph and as a tooltip on the point that is being changed.
11. Select the relevant details for the following under **Variation Specification** section of the screen.
  - Measure
  - Start Date
  - End Date
  - Percent Variation: Enter the % value. If you enter this, you cannot enter the Absolute Variation.
  - Absolute Variation: Enter the absolute value. If you enter this, you cannot enter the Percent Variation.

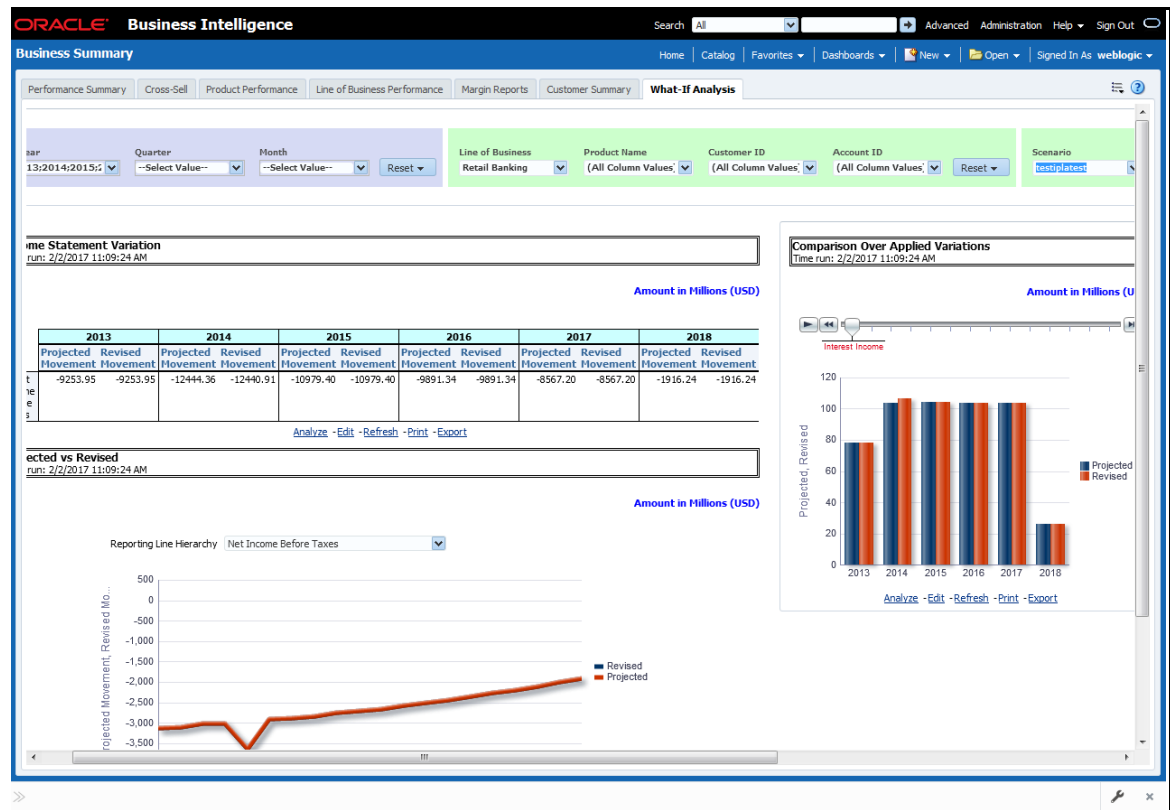
If there is an overlap in dates among various variation specification rows, the same can be overridden to apply the respective change.

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

12. Click **Preview Variations** to see the results.

13. Click **Save** after confirming the variations. After this, you will be redirected to the OBIEE screen where the applied variations can be seen and analyzed further.

The OBIEE screen is displayed as shown below:



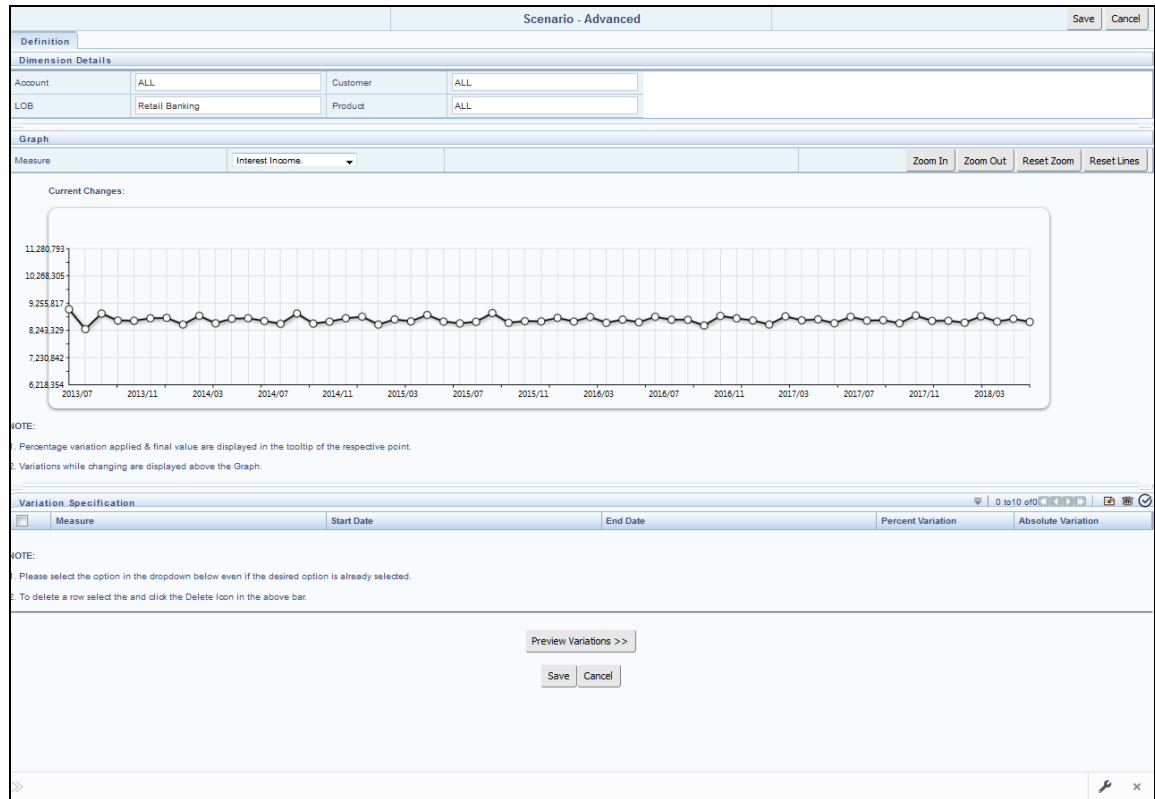
### 3.1.2 Advanced Scenario

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

The following procedure describes the steps to create a variation:

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

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



15. Apply the necessary **Dimension Details** for the following:

- Account
- Customer
- LOB
- Product

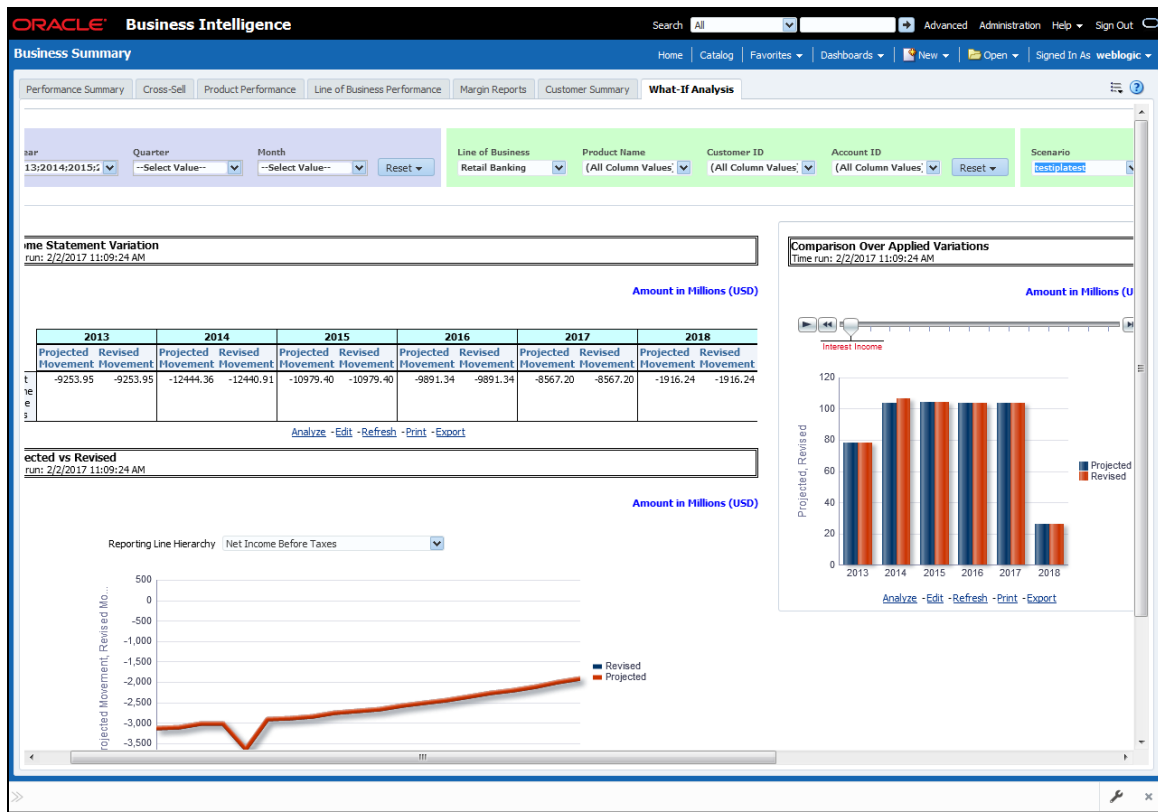
16. Select the relevant repline **Measure** to which you want to apply the variation.

17. Select a point on the graph and drag to apply the desired variations.

18. Click **Preview Variations** to see the results.

19. Click **Save** after confirming the variations. After this, you will be redirected to the OBIEE screen where the applied variations can be seen and analyzed further.

The OBIEE screen is displayed as shown below:



This chapter discusses the following topics:

- [Configuration for What-If Analysis](#)
- [PACS Integration with Price Creation and Discovery \(PCD\)](#)
- [Reporting Line Correlation Calculations](#)
- [What-If Analysis Limitation](#)

## 3.2 Configuration for What-If Analysis

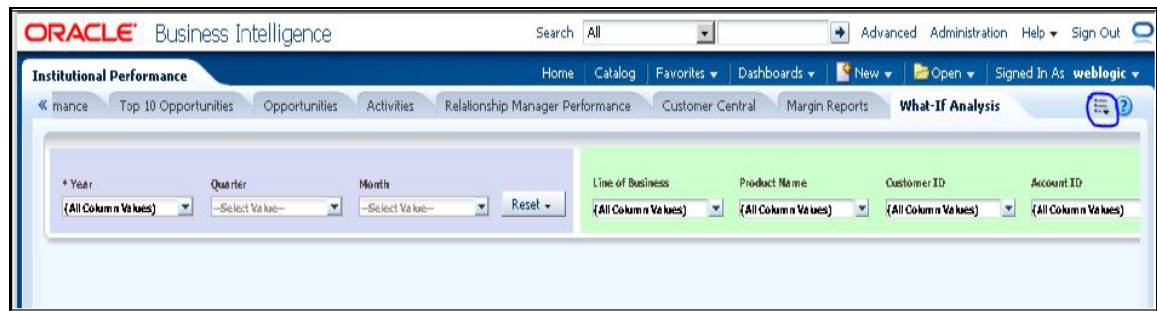
Configure connection to the What-If Variation application page in OFSAAI in the Create Scenario Analysis. OFSAAI is Oracle Financial Services proprietary tool which uses Java to enable users to apply variations on the projected data.

Example: Assume that the ofsaa hostname is 10.184.150.107 and the OBIEE analytics port is 7001. Hence, the

OBIEE analytics access url would be: <http://10.184.150.107:7001/PFT805>.

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

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

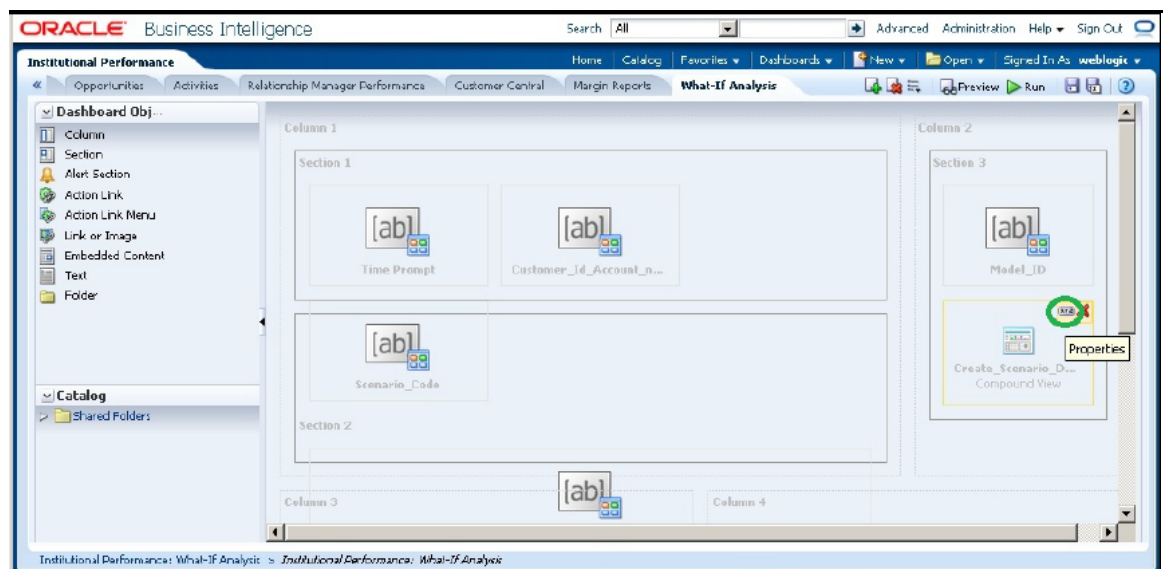


Edit the analysis Create Scenario.

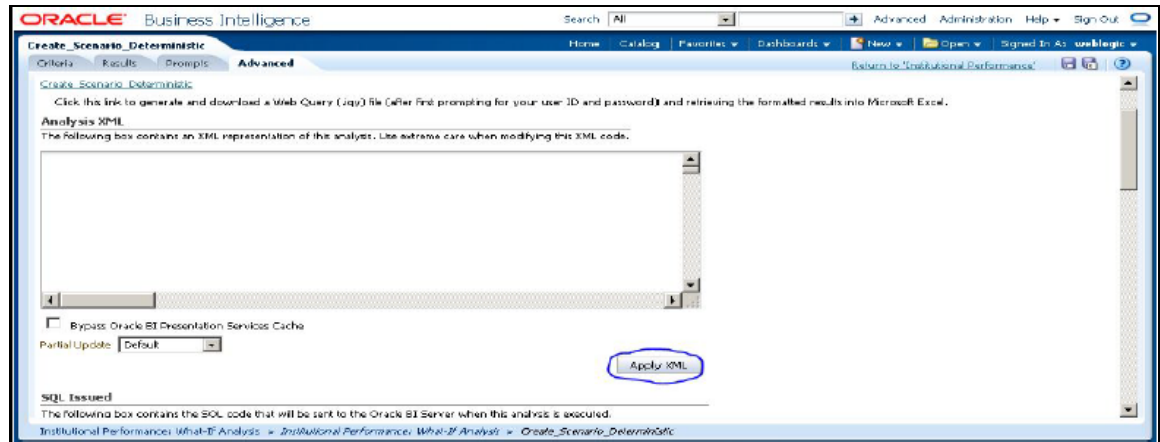
### NOTE

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

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



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



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

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

`http://10.184.150.107:7001/analytics`

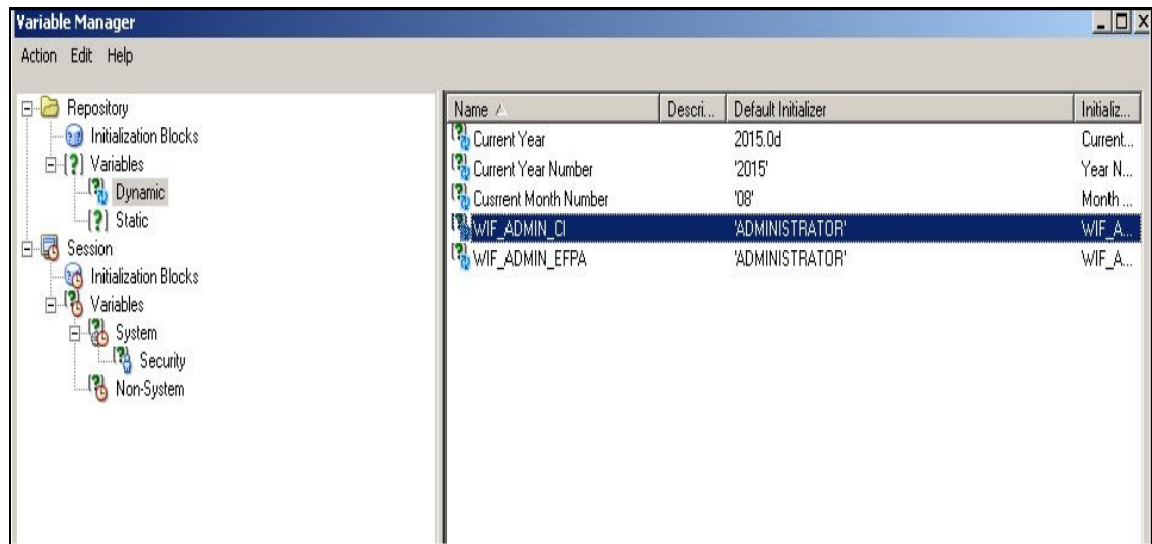
To configure these details to the What-If analysis framework, the user needs to execute the following update on the atomic schema:

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

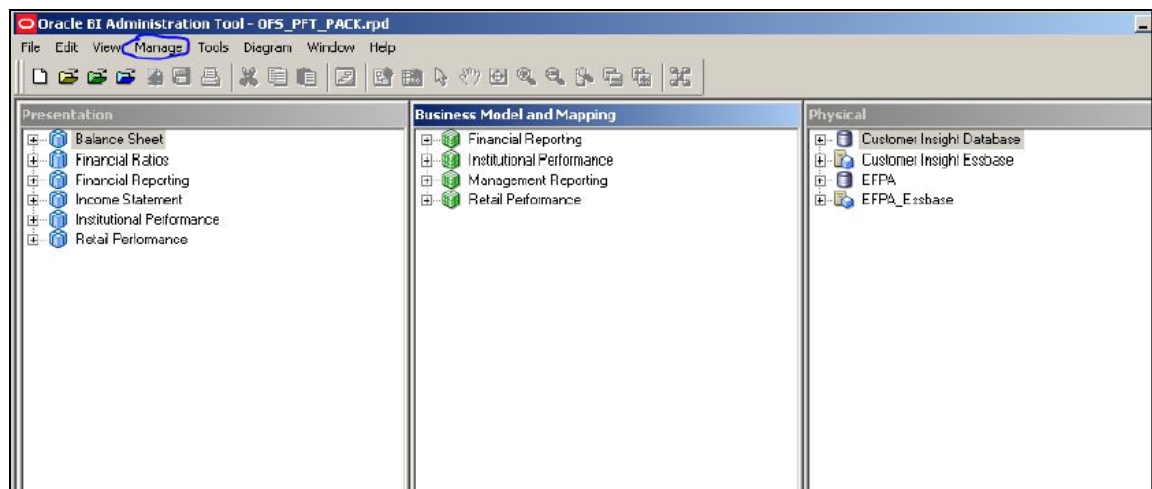
Where the hostname and port replaced would be the user's corresponding hostname and port instead of the examples mentioned above.

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

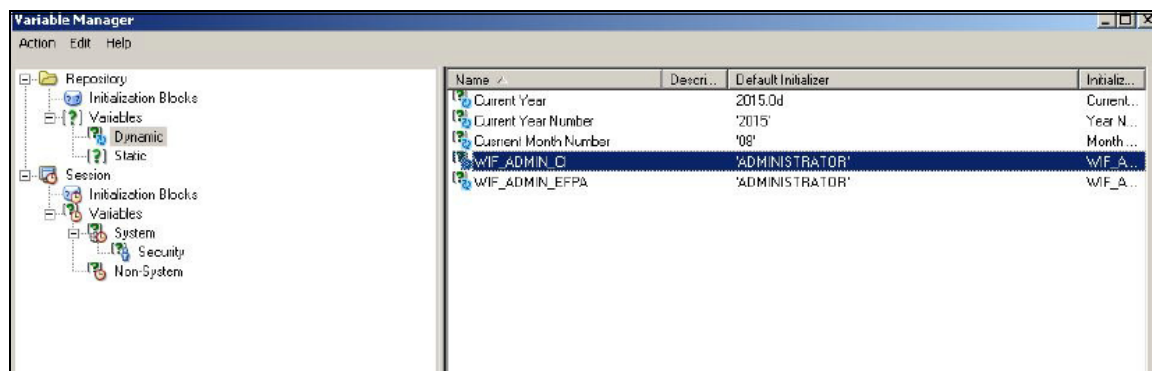




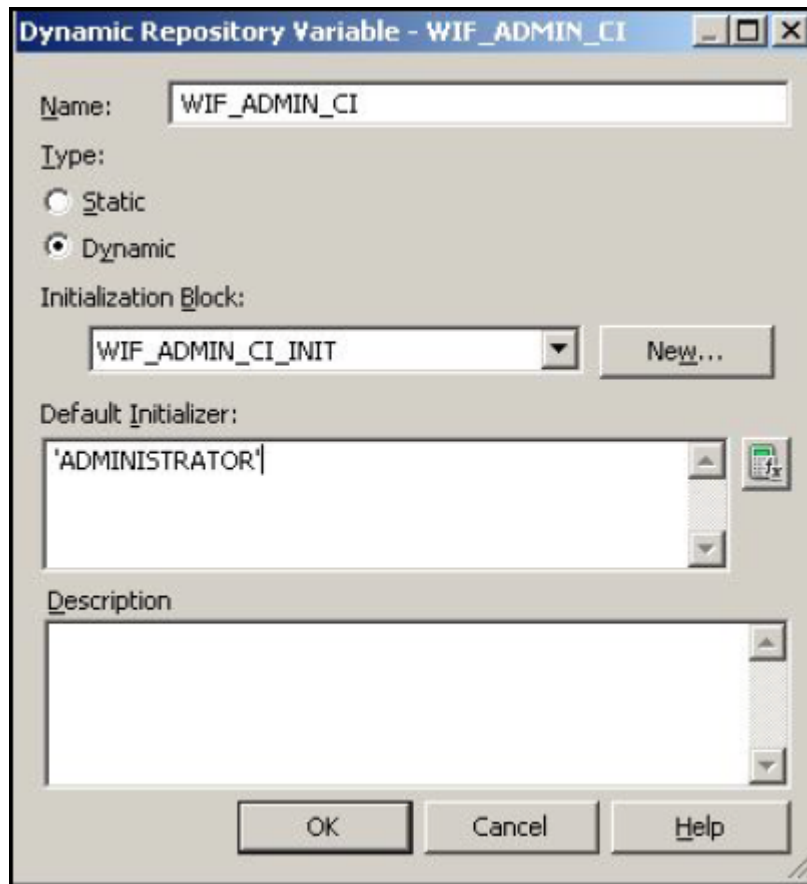
- Open the OFS\_PFT\_PACK RPD and navigate to the Variable Definition Screen.



- Modify the WIF\_ADMIN\_CI Repository Dynamic Variable:



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



10. If the web server is Tomcat of version  $\geq 8.0.18$ , following additional configuration needs to be done to avoid Performance Issues while performing What-If Analysis:

Add the following tags in the server.xml file under tomcat\_folder/conf/:

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

```
<Loader delegate="true"/>
```

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

```
removeAbandonedOnBorrow = "true"
```

```
removeAbandonedOnMaintenance="true"
```

Example :

```

<Context path="/PFT" docBase="/scratch/ofsaapp/tomcat-7.0.19/webapps/PFT" debug="0" reloadable="false" crossContext="true">
  <Loader delegate="true"/>
  <Resource auth="Container"
    name="jdbc/FICMASTER"
    type="javax.sql.DataSource"
    driverClassName="oracle.jdbc.driver.OracleDriver"
    username="pftconf30"
    password="ofsaasx"
    url="jdbc:oracle:thin:@10.184.153.87:1521:DEV12C"
    maxActive="1000"
    maxIdle="30"
    maxWait="10000" removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
    removeAbandonedOnBorrow="true" removeAbandonedOnMaintenance="true"/>
  <Resource auth="Container"
    name="jdbc/OFSPTINFO"
    type="javax.sql.DataSource"
    driverClassName="oracle.jdbc.driver.OracleDriver"
    username="pftatn30"
    password="ofsaasx"
    url="jdbc:oracle:thin:@10.184.153.87:1521:DEV12C"
    maxActive="1000"
    maxIdle="30"
    maxWait="10000" removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
    removeAbandonedOnBorrow="true" removeAbandonedOnMaintenance="true"/>

```

### 3.3 PACS Integration with Price Creation and Discovery (PCD)

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

### 3.4 Reporting Line Correlation Calculations

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

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

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

The following parameters are available in the FSI\_MODEL\_PARAMETERS table:

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

The following steps are used in repline correlation calculation:

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

### Excluding Outliers

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

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

### Testing for Stationarity

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

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

Results are reported and used in the cointegration test.

### Testing for Cointegration

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

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

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

## 3.5 What-If Analysis Limitation

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

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

## 4 Segmentation and Rules

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

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

This chapter covers the following topics:

- [Segmentation in IPA](#)
- [Segmentation in RPA](#)
- [Creating a Rule](#)
- [Editing a Rule](#)

### 4.1 Segmentation in IPA

Following are the several segmentation types that are being supported:

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

**NOTE**

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

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

Segmentation is done using the following dimensions:

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

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

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

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

## 4.2 Segmentation in RPA

Following are the several segmentation types that are being supported:

- Demographic Segmentation
- Profitability segmentation
- Risk Based Segmentation
- Behavioral Segmentation
- Channel Usage Segmentation
- Affluence Segmentation

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

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

- Location
- Credit Rating
- Currency
- Geography
- Industry
- Product
- Age
- Income
- Gender

When Price Creation and Discovery is integrated and is installed with IPA, user is required to update column V\_PRODUCT\_TYPE in FSI\_SEG\_REP\_LINE\_MAP table accordingly to the match the product type used in the Price Creation and Discovery Application.

Average of the measures present in Account Profitability for Account Segments across Reporting Lines and Month on Book for the account are reported into the following table:

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

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

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

DIM\_SEGMENT\_TABLE would be populated using SCD Process (Map reference number 267) and source would be a view DIM\_SEGMENT\_V for which data would be from DIM\_SEGMENT\_B/TL and FSI\_SEGMENT\_TYPE\_CD/MLS tables. FSI\_SEGMENT\_TYPE\_CD/MLS table stores list of Segment Types used in IPA. For example, Corporate Tracker, Behavioral, Profitability segments, and so on.

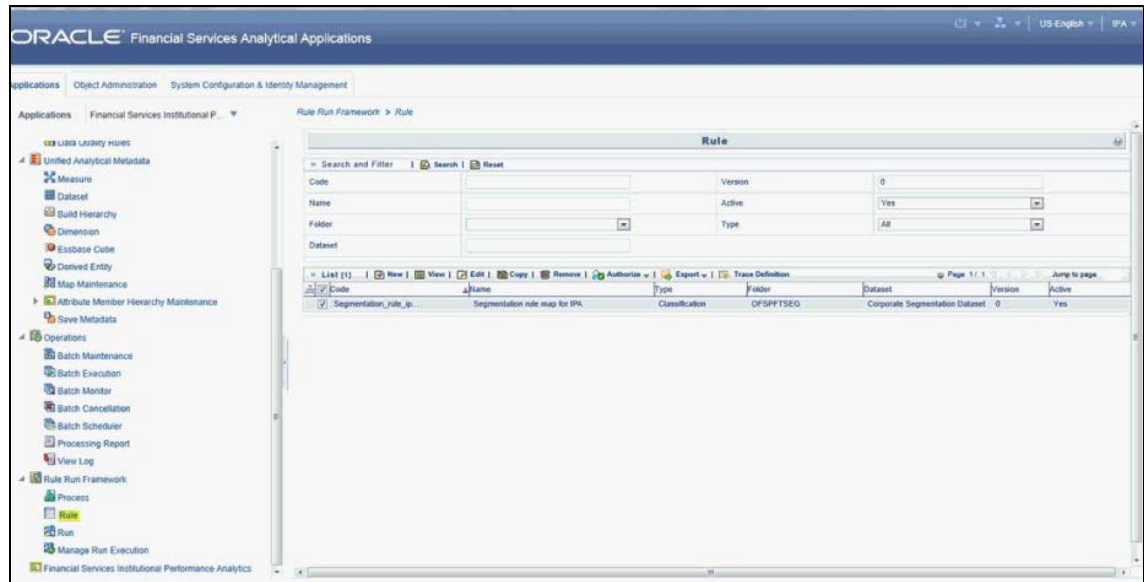
The Segment score will not be updated. The rule will update only the n\_segment.

## 4.3 Creating a Rule

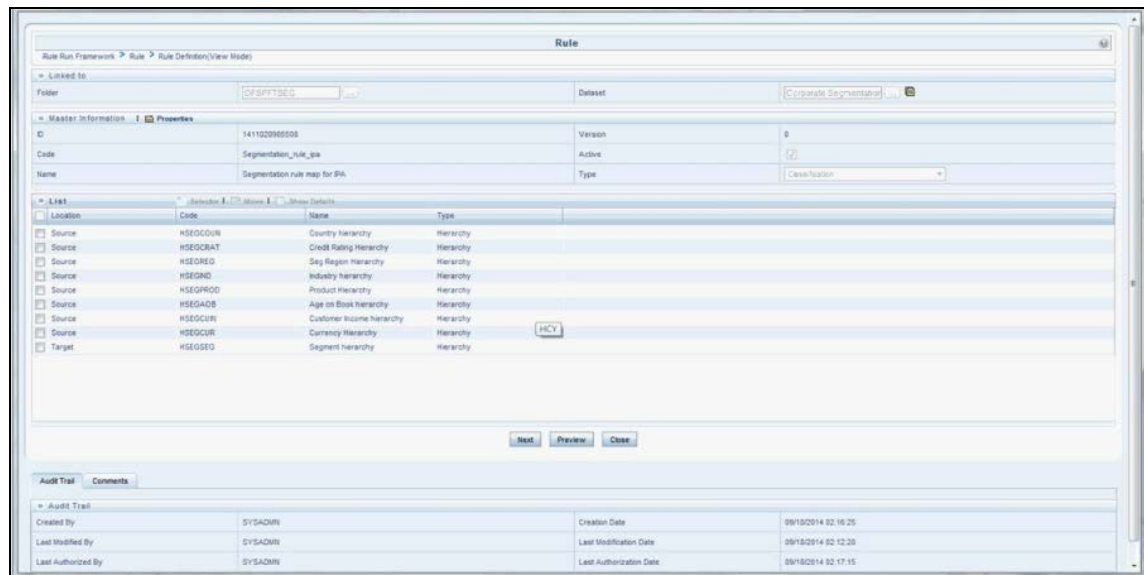
To define a rule, follow these steps:

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





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



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

Rule Run Framework > Rule > Rule Definition (View Mode)

Rule

Linked to: Folder: OFSPFTSEG Dataset: Corporate Segmentation

Master Information Properties

ID: 1411020905508 Version: 0

Code: Segmentation\_rule\_iss Active: ☒

Name: Segmentation rule map for IPA Type: Data Position

List

Location	Code	Name	Type
Source	HSEGCCOIN	Country hierarchy	Hierarchy
Source	HSECCORAT	Credit Rating Hierarchy	Hierarchy
Source	HSEOREG	Seg Region Hierarchy	Hierarchy
Source	HSEORND	Industry Hierarchy	Hierarchy
Source	HSEGRROG	Product Hierarchy	Hierarchy
Source	HSEGACB	Age on Book hierarchy	Hierarchy
Source	HSEGCIN	Customer Income hierarchy	Hierarchy
Source	HSEGCUR	Currency Hierarchy	Hierarchy
Target	HSEGSEG	Segment hierarchy	Hierarchy

Audit Trail

Created By	Creation Date
SYSADMIN	09/16/2014 02:16:25

Last Modified By	Last Modification Date
SYSADMIN	09/16/2014 02:12:20

Last Authorized By	Last Authorization Date
SYSADMIN	09/16/2014 02:17:15

## 4.4 Editing a Rule

To edit a rule, follow these steps:

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

Rule Run Framework > Rule > Rule Definition (Edit Mode)

Rule

Linked to: Folder: OFSPFTSEG Dataset: Corporate Segmentation

Master Information Properties

ID: 1411020905508 Version: 0

Code: Segmentation\_rule\_iss Active: ☒

Name: Segmentation rule map for IPA Type: Data Position

List

Location	Code	Name	Type
Source	HSEGCCOIN	Country hierarchy	Hierarchy
Source	HSECCORAT	Credit Rating Hierarchy	Hierarchy
Source	HSEOREG	Seg Region Hierarchy	Hierarchy
Source	HSEORND	Industry Hierarchy	Hierarchy
Source	HSEGRROG	Product Hierarchy	Hierarchy
Source	HSEGACB	Age on Book hierarchy	Hierarchy
Source	HSEGCIN	Customer Income hierarchy	Hierarchy
Source	HSEGCUR	Currency Hierarchy	Hierarchy
Target	HSEGSEG	Segment hierarchy	Hierarchy

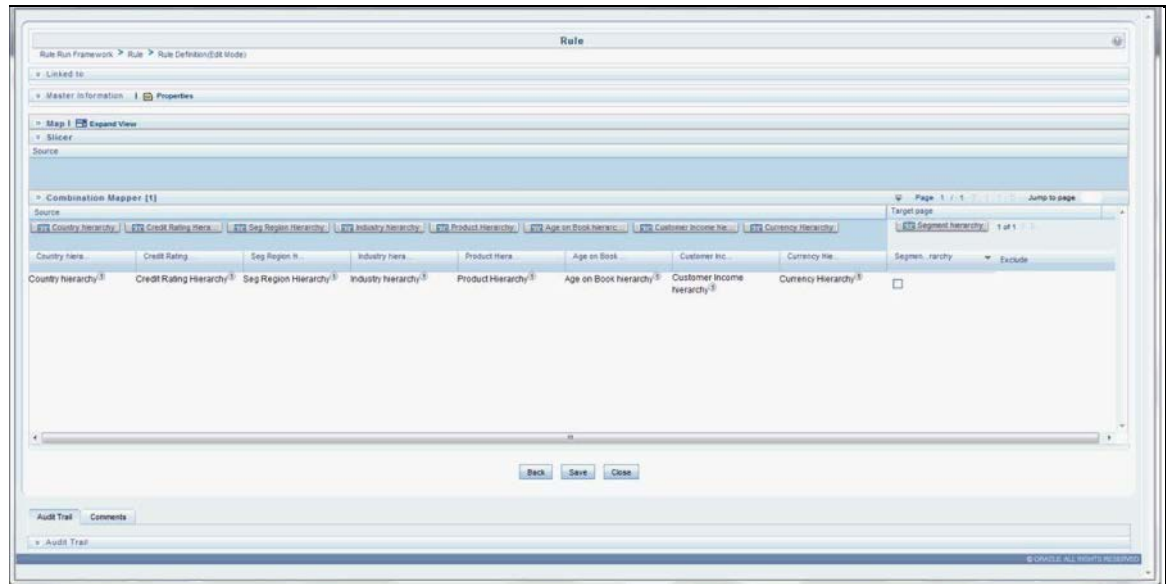
Audit Trail

Created By	Creation Date
SYSADMIN	09/16/2014 02:16:25

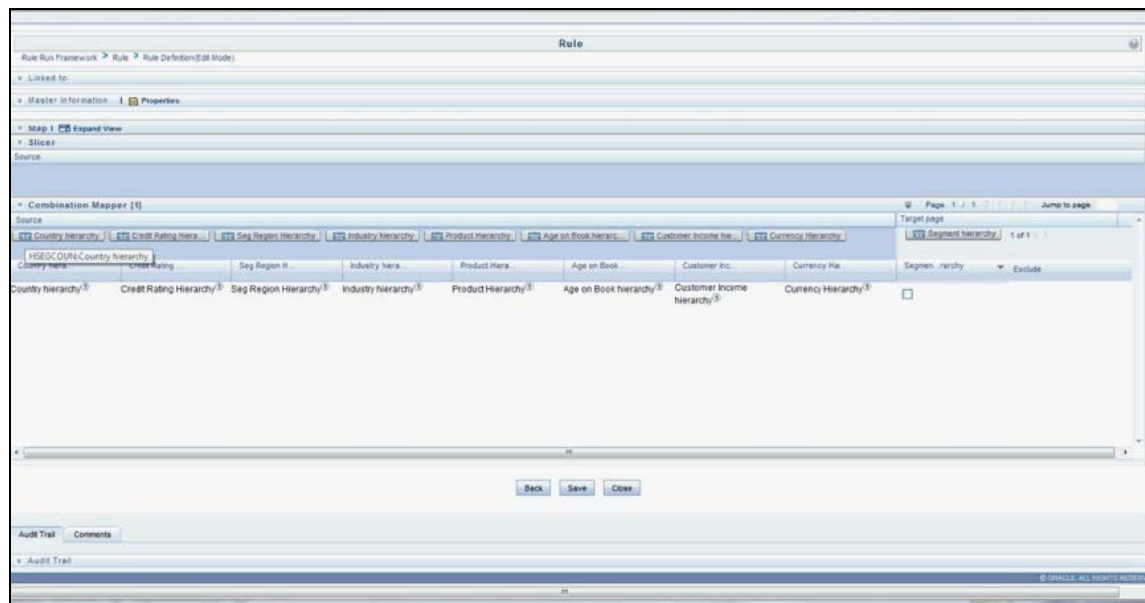
Last Modified By	Last Modification Date
SYSADMIN	09/16/2014 02:12:20

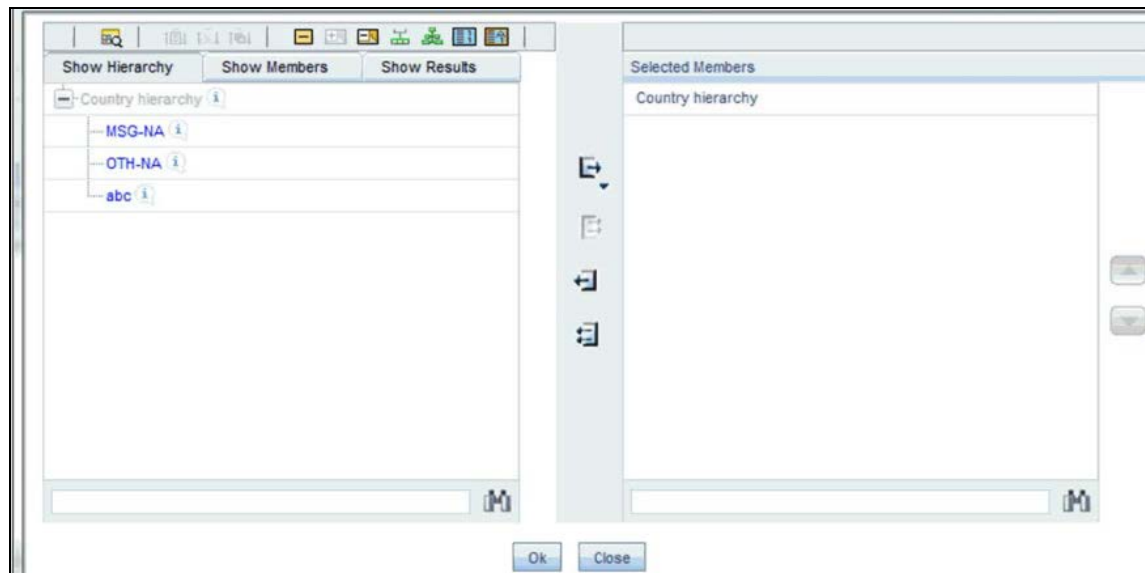
Last Authorized By	Last Authorization Date
SYSADMIN	09/16/2014 02:17:15

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

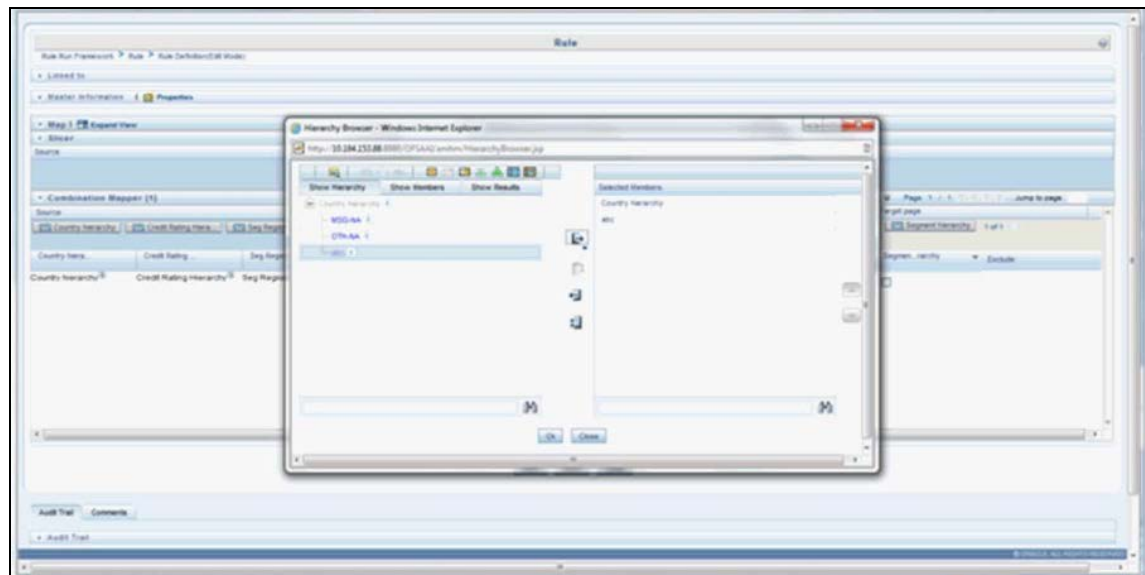


3. Click **Hierarchy** as shown in the following figure and the hierarchy screen opens up.

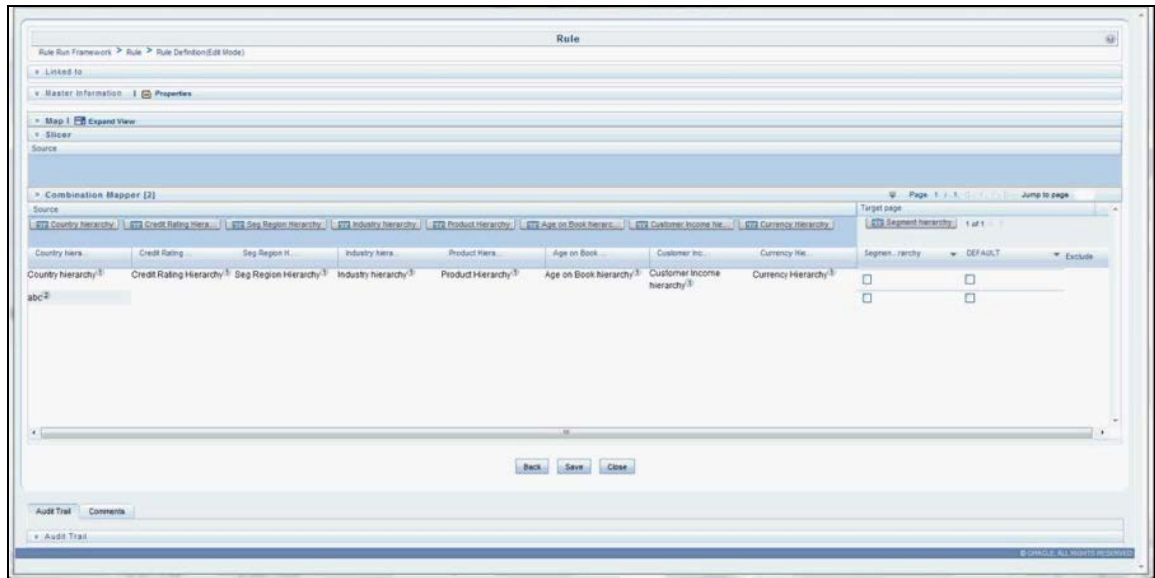




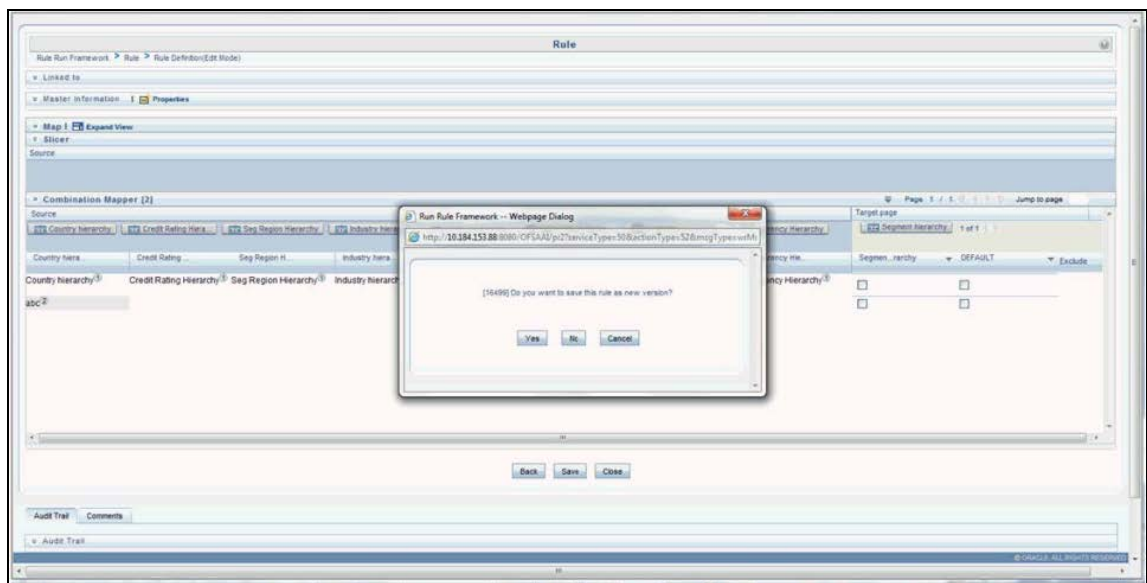
4. Select the hierarchy and click **OK**.



The selected node appears in the rule.



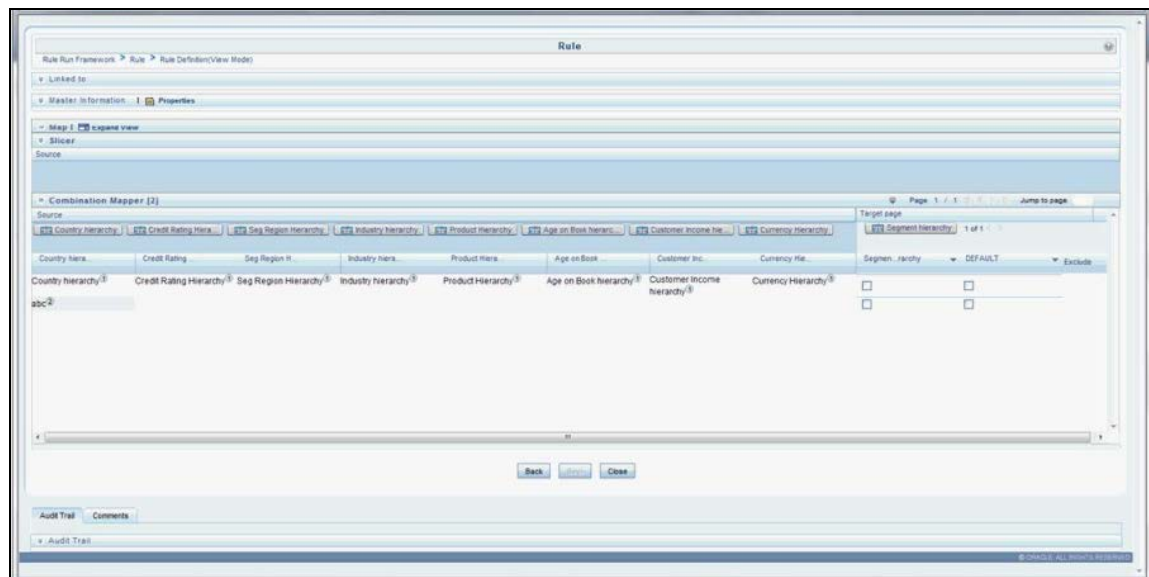
- Similarly, select all the nodes that need to be considered for the rule and assign it to the target hierarchy. Click **Save**. A confirmation message appears as shown in the following.



- On clicking **Yes**, the following message appears: *Save action with authorize was successful on following definitions Segmentation\_rule\_ipa or Save action with authorize was successful on following definitions Segmentation\_rule\_rpa.*



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



## 5 Defining a Batch

Batch refers to a set of executable processes based on a specified rule. Batch Maintenance framework within OFSAAI facilitates you to create and maintain the Batch Definitions. You can process the Batch scheduled for execution from Batch Maintenance and also from other modules.

You need to have Data Centre Manager function role mapped to access the Operations framework within OFSAAI. You can access Batch Maintenance by expanding Operations section within the tree structure of LHS menu. The Batch Maintenance window displays a list of Batches scheduled for maintenance with the other details such as Batch ID, Batch Description, and the editable state of the Batch.

You can create a batch from the Batch Maintenance screen as mentioned below:

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

The New Batch Definition window is displayed.

Enter the Batch details as given in the following table.

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: <ul style="list-style-type: none"> <li>• The Batch Name should be unique across the\ Information Domain.</li> <li>• The Batch Name must be alpha-numeric and should not start with a number.</li> <li>• The Batch Name should not exceed 41 characters in length.</li> <li>• The Batch Name should not contain special characters "." and "-".</li> </ul>
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 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.

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

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

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

## 6 Adding a New Dimension

This section explains the steps to be performed by the user for adding a new dimension to the cube.

As a prerequisite, dimension tables should be added in the data model and the fact table needs to have the referential key with the dimension table. These dimension tables will hold dimension members and can be level-based or parent-child.

Level based dimension tables contain columns for each level of the hierarchy, while parent-child dimension tables contain columns for storing the relationship between the parent and child members. These dimension tables can be loaded from external systems or can be maintained within the Dimension Management component of OFSAAI.

If user intends to maintain the dimension within OFSAAI, see Data Model Utilities Guide for adding dimension tables under "Object Management" chapter.

This chapter discusses the following topics:

- [Introduction to Dimensions](#)
- [Dimension Definition Process](#)
- [Metadata](#)

### 6.1 Dimension Definition Process

#### Step 1 - Add Business Hierarchy

1. To define a new **Business Hierarchy**, go to **Unified Metadata Manager**, select **Business Metadata Management** and choose the type of hierarchy.

Hierarchy Types are:

- **Regular:** For representing non-time and non-measure dimensions in a hierarchical format. For example, this type are Product, Organization Unit, and so on.
- **Measure:** For representing the measures in the hierarchical format. This corresponds to a ACCOUNT hierarchy within the ESSBASE. An example of this type is Management Reporting Line.
- **Time:** For representing the calendar or date dimension in a hierarchical format. This corresponds to a TIME hierarchy within Essbase. An example of this type is Calendar hierarchy.



2. Choose Hierarchy subtype. Hierarchy SubTypes are:
  - **Non Business Intelligence Enabled:** For representing the hierarchy with underlying data store containing just leaves and nodes are built within the metadata of the hierarchy. This subtype is useful for modelling bucket/range, ragged, and non-additive hierarchies.
  - **Business Intelligence Enabled:** For representing the hierarchy with underlying data store as level-based dimension table. This subtype is useful for modelling balanced hierarchies.
  - **Parent Child:** For representing the hierarchy with underlying data store as a parent-child dimension table. This subtype is useful for modelling ragged hierarchies.
3. Select the "Total Required" property, if a TOTAL is required to be included as the root node of the hierarch and select the "List" property, if hierarchy is a flat list of members without any levels.
4. Choose the entity and attribute on which the hierarchy is based. The components for hierarchy definition differ for each subtype of the hierarchy.
  - If subtype is "Non Business Intelligence Enabled", then the user can add nodes and order in which the node should appear in the hierarchy (sort-order). Node identifiers
  - are SQL expressions that are specified for leaf members and data is classified based on the node identifiers.
  - If sub-type is "Business Intelligence Enabled", then the user can specify the levels and SQL expression for each level within the hierarchy.
  - If sub-type is "Parent Child", then the user can specify the column that contains the parent member and that contains the child member.

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

## Step 2 – Add Business Dimension

A Business Dimension is a structure of one or more logical grouping (hierarchies) that classifies data. It is the categorization across which measures are viewed. A dimension can have one or more hierarchies. Business Dimension facilitates you to create a logical connection with measures. It gives you various options across which you can view measures.

1. To define a new **Business Dimension**, go to **Unified Metadata Manager**, select **Business Metadata Management**.
2. Choose the **Dimension Type**. Dimension Type is same as Hierarchy Type and helps to filter the hierarchies that will be part of the dimension. A dimension will contain one or many hierarchies. Choose the hierarchies that are part of the dimension.

The User Info grid at the bottom of the screen displays the metadata information about the Business Dimension created along with the option to add comments.

3. Click **Save** in **Add Business Dimension** screen to save the details.

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

## Step 3 – Modify Data Set

To modify **Data Sets**, go to **Unified Metadata Manager --> Business Metadata Management**.

Identify data sets that are based on the modified fact table. Open the data set definition. Include the new dimension table in the data set. Modify the data set JOIN to include the join clause between the fact table and new dimension table. Save the data set.

**Edit Data Sets**

Data Sets > Data Set Definition (Edit mode)

**Data Set Details**

Code \* DSFVWCP

Short Description \* Account Fair-Value Inception

Long Description Hedge Management Inception Dataset for Account FV

**Entities** 1 to 5 of 5

**Selected Entities**

DIM\_DATES

DIM\_FCST\_RATES\_SCENARIO

DIM\_HEDGE

FCT\_ACCOUNT\_FAIR\_VALUE

FSL\_HM\_HEDGE\_INSTRU\_MAP

**Data Set Definition**

ANSI Join

Join/Filter Condition DIM\_HEDGE.N\_HEDGE\_ID = FSL\_HM\_HEDGE\_INSTRU\_MAP.HEDGE\_ID AND FCT\_ACCOUNT\_FAIR\_VALUE.N\_ID\_NUMBER = FSL\_HM\_HEDGE\_INSTRU\_MAP.ID\_NUMBER

Date Filter

Order By

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

#### Step 4 – Modify Cube Definition

Modify "Cubes" in **Unified Metadata Manager -> Business Metadata Management**.

Identify the cube that needs to be modified. Open the cube definition. Add the new dimension. Map the measures to the newly added dimension and Save the cube definition.

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

#### Step 5 – Build Cube

Assuming that the dimension table and fact table is loaded with relevant data, cube can be built. Define batch to execute the CREATE CUBE component that will build the outline and load data in ESSBASE.

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

### 6.1.1 Steps to follow while using ESSBASE Source for Relationship Manager Hierarchy

The following are the steps to follow while using ESSBASE Source for Relationship Manager Hierarchy.

1. When creating a Parent Child hierarchy using ESSBASE, ESSBASE creates two additional parents to the existing hierarchy. For example:

Relationship Manager Hierarchy	
▽ Relationship Manager Dimension	
▽ HEPMRM02:HEPMRM02:ND	
▽ A	
B	
▽ C	
▽ D	
▽ E	
F	

Relationship Manager Hierarchy	
▽ A	
B	
▽ C	
▽ D	
▽ E	
F	

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

2. In the context of using Relationship Manager Hierarchy for Institutional Performance, there is a concept of visibility of data implemented. This means that while using a cube source, D can see A listed as a manager in the hierarchy. However, D does not have the privilege to view the data (revenue, movement, and so on) related to A but can view the data for all the child nodes of D, for example, E and F. As a result, if Relationship Manager Hierarchy is selected along with Direct Movement, no results are displayed.

Selected Columns

Double click on column names in the Subject Areas pane to add them to the Selected Columns pane. To delete by clicking or hovering over the button next to its name.

Dim - Management

Fact - Account Profitability

Relationship Manager Hierarchy

Direct Movement

Compound Layout

No Results

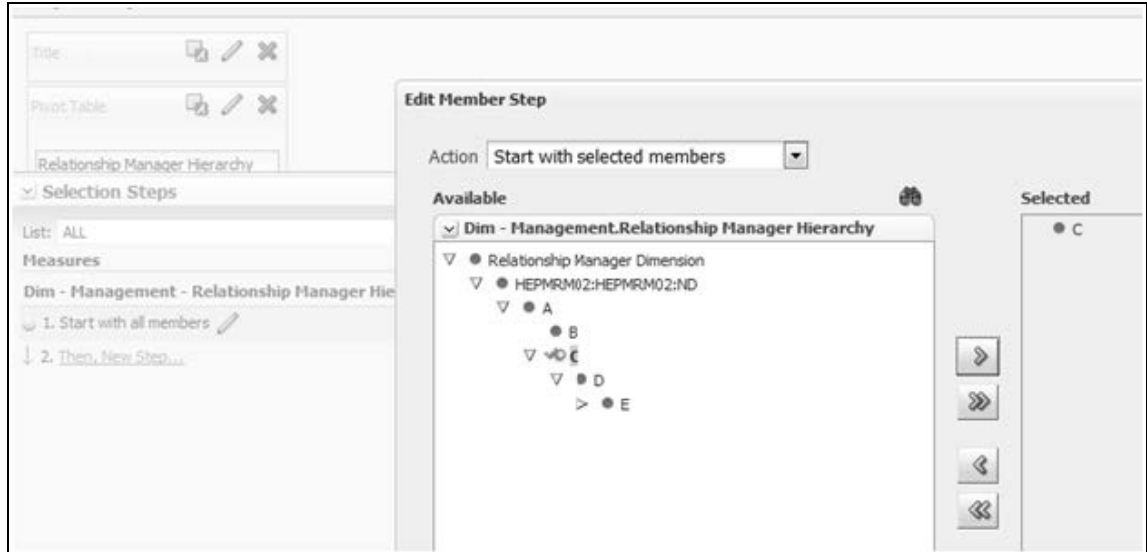
The specified criteria didn't result in any data.

Refresh

3. To view results for the logged-in Relationship Manager, the user must choose the Relationship Manager who is mapped to the user. In this case, the logged-in user is weblogic. From FSI\_M\_USER\_MANAGER\_MAP, the following is seen:

	V_USERNAME	V_MANAGER_CCDE	D.V_AM_ACCT_MANAGER_FIRST_NAME
1	RELATIONSHIP MANAGER ...	A01	A
2	SALES REPRESENTATIVE ...	A02	B
3	weblogic	A03	C

- Thus the user must start the hierarchy with C.



As a result, the user will be able to see the data related to the manager.

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

## 6.2 Metadata

### 6.2.1 Technical Metadata

- Sheet\_for\_DIM\_STG\_MAP.XLS excel sheet lists the SCD's packaged in the IPA application.
- OFS\_IPA\_Technical\_Metadata.xls lists the Institutional Performance Analytics technical metadata.

### 6.2.2 Optional Metadata

- PFT Acc\_Sum\_tech.xlsx lists the technical metadata related to PFT account summary.
- FTP Acc\_Sum\_tech.xlsx lists the technical metadata related to FTP account summary.

### 6.2.3 Business Metadata

- OFSIPA Business metadata.xlsx lists the Oracle Financial Services Institutional Performance Analytics BI 6.0 Business Metadata.

### 6.2.4 Reporting Metadata

- Customer Attributes - IPA.xlsx lists the Customer Attributes.
- IPA-RPD\_webcat.xlsx lists the IPA-RPD-Webcat metadata.

## 7 Adding a New Measure

This section details the steps to be performed by the user for adding a new measure to the cube. As a prerequisite, the fact table needs to have the column that holds values for the new measure.

This chapter discusses the following topics:

- [Measure Definition Process](#)
- [Build Cube](#)

### 7.1 Measure Definition Process

#### Step 1 – Add Business Measure

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

Add Business Measures	
Business Measures > Business Measure Definition (Add mode)	
Business Measure Details	
Code *	MEPM001
Short Description *	EOP Balance
Long Description	End of period balance
Business Measure Definition	
Aggregation Function	SUM
Roll up	<input checked="" type="checkbox"/>
Entity	
Attribute	
Business Exclusions	
Filter Expression	
DataType	Decimal

For more information on Business Measures, see Business Measures under Unified Metadata Manager chapter in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

## Step 2 – Modify Cube Definition

1. From **Unified Metadata Manager**, select **Business Metadata Management**, then select **Cube**.
2. Identify the cube that needs to be modified.
3. Edit the cube definition.
4. Add the new measure.
5. Map the measures to the to the required dimensions.
6. Save the cube definition.

## 7.2 Build Cube

Assuming that the dimension table and fact table is loaded with relevant data, cube can be built.

Define batch to execute the CREATE CUBE component that will build the outline and load data in ESSBASE.

For more information on executing the batch, see *Oracle Financial Services Analytical Applications Infrastructure User Guide*.



## 8 Developing a New Cube

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

### 8.1 Procedures to Develop a New Cube

#### Step 1 – Add Cube

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

#### Step 2 – Include Dimensions

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

#### Step 3 – Specify Variations

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

#### Step 4 – Specify Dataset

Specify dataset corresponding to the selected dimensions and measures. Data set will supply required data to the cube.

#### Step 5 – Specify Node Level Formula

If node level formula's are required to be specified for the nodes within the hierarchy, then they can be specified in this UI.

#### Step 6 – Save and Build

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

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

## 9 Run Rule Framework

In cases where data is required to be loaded for fact tables in multiple runs, the OFSAAI Rule Run Framework comes in handy. For example, for population of FCT\_CRM\_ACCOUNT\_SUMMARY, the parameters of the batch execution include a parameter \$RUNSK = -1. (Refer to the parameters of the batch ##INFODOM##\_aCRM\_CRM\_Acc\_Summ, Task1).

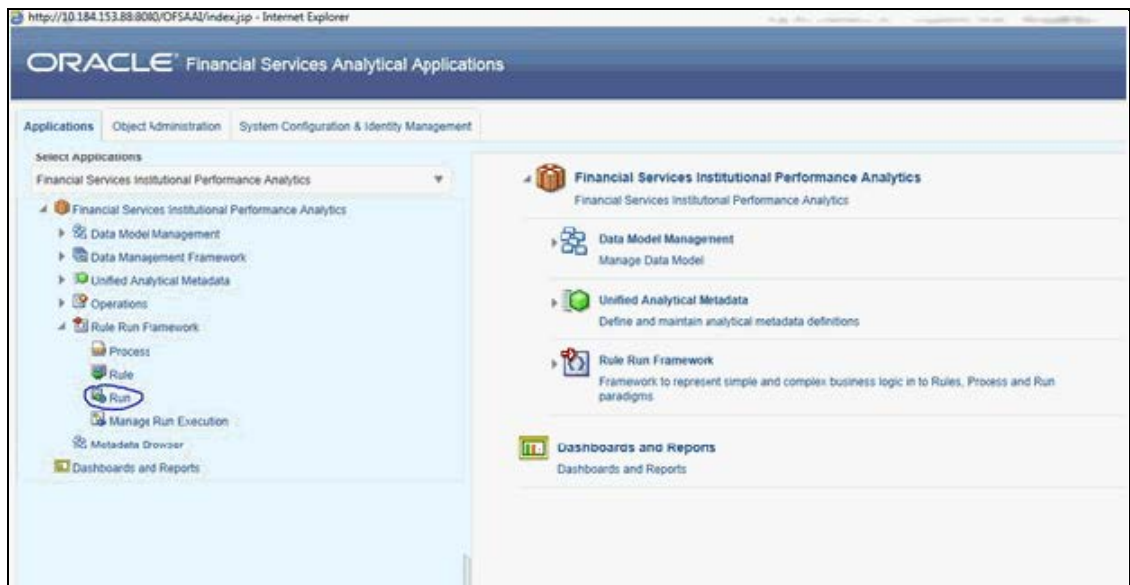
This batch execution loads the column N\_RUN\_SKEY in FCT\_CRM\_ACCOUNT\_SUMMARY as -1. This will be a default run from the seeded batch. In order to be able to enter data for multiple runs, the batch tasks can be defined in Rule Run Framework. This will then create a batch internally which will load data for a different run into the fact table.

This chapter discusses the following topics;

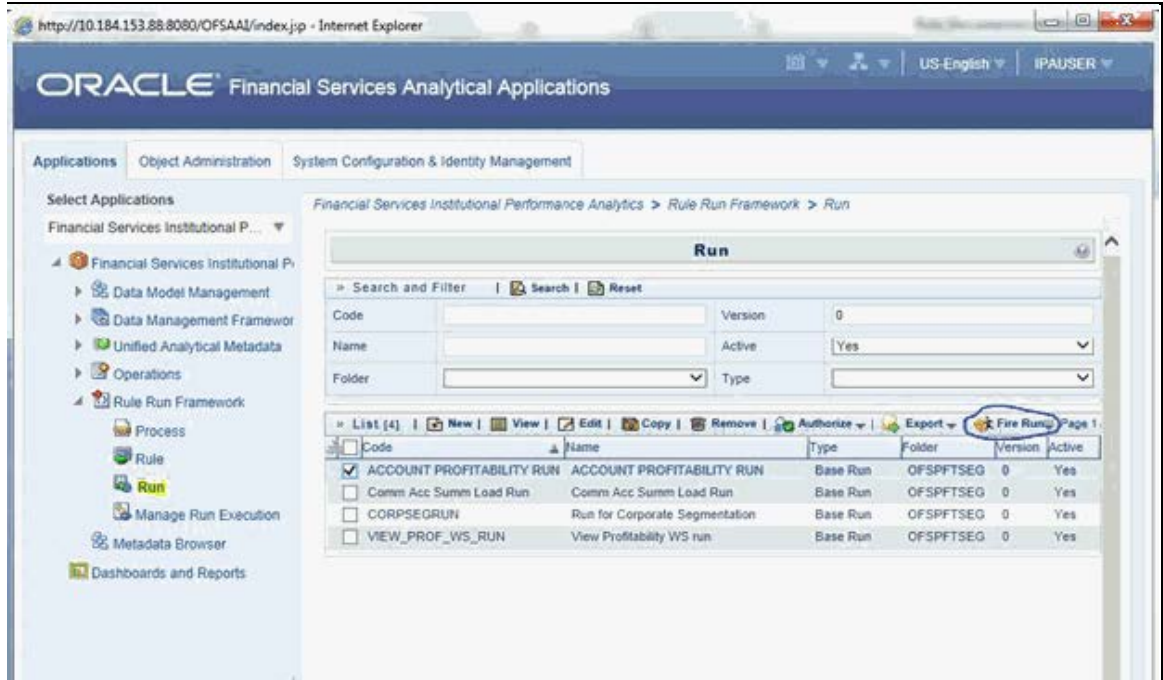
- [Executing a seeded run](#)
- [Runs Available](#)

### 9.1 Executing a seeded run

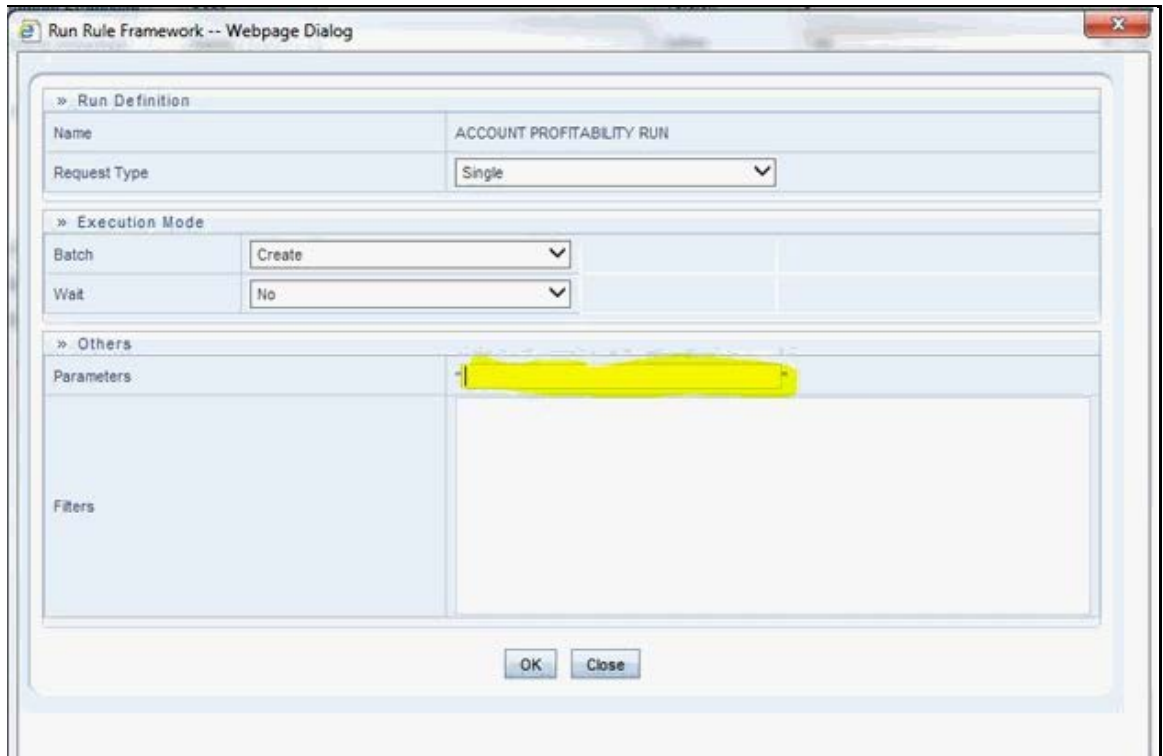
1. Navigate to **Rule Run Framework > Run**.



2. Choose a Run by checking the box before it and click **Fire Run**.



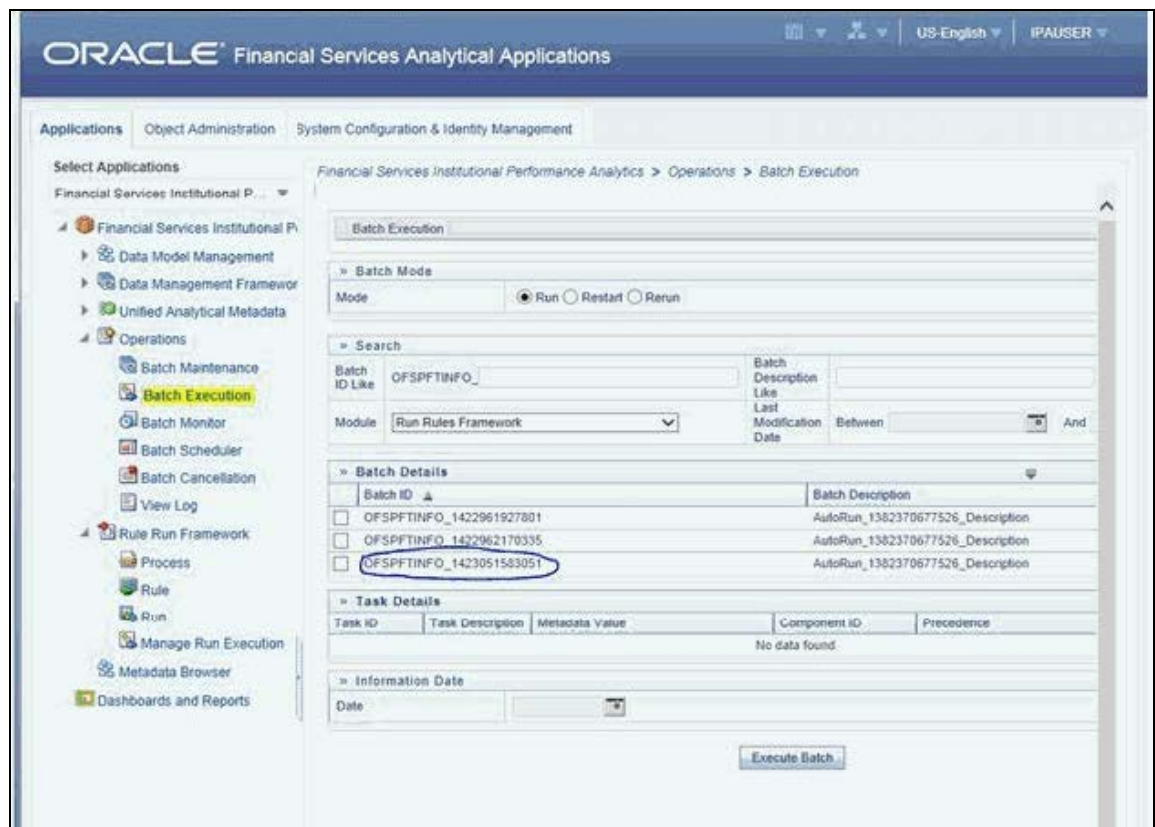
3. Enter the parameters required to execute the run (refer to details of individual runs) and click OK.



The following message will be displayed: Fire run successful.



4. Execute the batch by navigating to Operations>Batch Execution and select the latest batch created in the Run Rules Framework module.



## 9.2 Runs Available

Following are the runs available for PA applications:

1. ACCOUNT PROFITABILITY RUN: Run for loading FCT\_CRM\_ACCOUNT\_SUMMARY and FCT\_ACCOUNT\_PROFITABILITY

Parameters to be entered as follows:

##RCY##", "<INFODOM>", "FCT\_ACCOUNT\_PROFITABILITY

---

**NOTE**

There are no quotes at the beginning and the end of the string. Consider as if the quotes are enclosing the delimiter (comma).

2. Comm Acc Summ Load Run: Run for loading FCT\_COMMON\_ACCOUNT\_SUMMARY
3. VIEW\_PROF\_WS\_RUN: Run for executing Web Service
4. RETSEGRUN: Run for executing Retail Segmentation Rule

## 10 Loading Multiple Load Runs in OFSAA

Multiple load run enables data to be loaded multiple times during the day for staggered processing of data by analytical applications. The degree of complexity of data required by analytical applications vary from one to the other, the load run ensures that the customer can process the data as soon as it is ready for an application to uptake. This reduces the turnaround time to reporting, by avoiding the 'end of day' type of processing of information as part of the original design.

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

FSDf staging model provides customers a flexibility to load multiple snapshots of the data in the staging tables (Product Processor's). A column named `n_load_run_id` was introduced as part of the primary key of the product processor tables to enable this. But the full fledged functionality to load and manage these snapshots will be part of the platform release at a later stage. Customers who would like to leverage this design in 8.0.3 release, the following mentioned changes will need to be done as a workaround to load multiple snapshot of data from staging to results tables such as Fact Common Account Summary.

For Loading Multiple Snapshots of Data for the same `FIC_MIS_DATE`, the existing T2T's need to be executed via Run Rule Framework and load needs to be filtered accordingly for each load run via the run filter. To enable execution of this run, navigate to `$FIC_HOME/ficweb/webroot/conf` and dit the file `excludeURLList.cfg` and add the following entry at the end of the file `[SQLIA].pr2`

There should not be any blank line in the file.

This chapter discusses the following topics:

- [Features](#)
- [Design Details](#)
- [Data Transformations](#)

### 10.1 Features

Following are the features:

- To optimize the end-to-end data flow and the need for intra-day reporting, institutions could load intra-day records into OFSAA. Current application can only handle one set of records per date (incremental loads are not possible).
- Users need to adjust and reload data (either full or partial) for the current date.
- Users need to adjust and reload data (either full or partial) for any of past dates.

### 10.2 Design Details

Loading of data into OFSAA can be done in any of the following ways:

- ETL Tool
- OFSAA F2T
- OFSAA T2T

- OFSAA Excel upload
- OFSAA DIH

OFSAA data model includes load run identifier as part of the primary key for a set of staging tables. This enables data to be stored for multiple load runs for any date. OFSAA data model also has a table to maintain master information about load run and can be used for identifying/filtering load run during run execution within OFSAA.

OFSAA data model also another entity that tracks the load run mapped to the functional key of each of the staging table. Since OFSAA processing is on snapshot of data, this entity helps users to identify set of records that are latest to be used in processing. If there is a need to load multiple sets of data within a day, customers can use the below components to manage the same.

If set of data is snapshot:

- Register table that got loaded in the load run through a DT (Register\_load\_run\_details)
- Register table that got loaded in the load run through a DT (Register\_load\_run\_details)
- Use the load run identifier to load data into OFSAA staging. You can use the same load run identifier for all the entities loaded in the same batch/group.
- Specify run-filter during OFSAA execution to filter records for the maximum run identifier within the day

If set of data is incremental:

- Register load run information in master table through a DT (Register\_Load\_Run\_Master)
- Register table that got loaded in the load run through a DT (Register\_load\_run\_details)
- Register table that got loaded in the load run through a DT (Register\_load\_run\_details)
- Execute another DT (Populate\_Load\_Run\_Map) with different parameters for each entity to maintain the latest load run for each record.
- Modify the T2T's to join with the load run map tables for identifying and filtering on the latest set of records to be used in processing.

There could be some entities that can follow snapshot and some entities incremental load. Decision of snapshot vs incremental depends on the above use-case.

## 10.3 Data Transformations

### Function - Register Load Run

Parameters - Batch ID, MIS-Date, Load Run Name, Load Run Purpose, Load Run Type

#### Steps

1. Check if the parameters are valid. Load run type can be 'B – Base, A – Adjustments, P – Backdated adjustments'.
2. Check uniqueness of load run name. Load run name is a user-specified string for easier retrievals could be MIS-DATE <Sequence> or the starting timestamp of load run.
3. Increment the load run id for a given MISDATE and insert the rest of the details.

4. Return the load run identifier (if possible, else user will query this table to get the load run id given a name).
5. Log messages accordingly.
6. Return success/failure.

### 10.3.1 Execution

Execute this DT before loading any fact for intra-day load. Use the registered load run identifier as a value to map to load run identifier field in staging. When one load run is ongoing (loading data to OFSAA using the load run identifier), do not register any new load runs. Else, make sure load run name is used as a filter instead of max load run identifier when querying the load run master table.

#### Function – Register Load Run Details

Parameters – batch id, mis-date, load run name, load run id, stage table name, load type

##### Steps:

1. Check if the parameters are valid. Load type can be 'S – Snapshot, I - Incremental'.
2. Load run name or ID can be provided. If load run name is provided, we can lookup into load run master for retrieving the ID. Check if table name exist.
3. Register the information in load run details table.
4. Log messages accordingly.
5. Return success/failure.

### 10.3.2 Execution

Execute this DT after registering load run master and before loading any fact for intra-day load. When one load run is ongoing (loading data to OFSAA using the load run identifier), do not register any new load runs.

#### Function – Populate Load Run Map

Parameters – batch id, mis-date, and stage table name

##### Steps:

1. Check if the parameters are valid.
2. Pick the corresponding load run map table from a setup table. Pick the corresponding functional key columns and their mapping to load run map table from a setup table.
3. If record do not exist in load run map table for the functional key in staging, then insert a new record with the functional key and load run identifier.
4. If record exist in load run map table for the functional key in staging, then update latest record indicator for existing rows to 'N' and then insert a new record with the functional key and load run identifier.
5. This operation has to be done in bulk mode.
6. Log messages accordingly.
7. Return success/failure.



### 10.3.3 Execution

Execute this DT after loading any fact for intra-day load in case the table has incremental loads.

Multiple data runs can be done for the same date using one of the following two approach:

1. Using the Load Run ID, multiple data runs can be done for the same date. By default, the load run ID will be 0.
2. There will be multiple run\_keys generated for each extraction date. If there are four sources, then 4 distinct run keys are generated for the load date. Some of the out of box dashboards will show partial data as there is a filter for the Run. If all the sources need to be seen at a time, the processing should happen in a single run.
3. All the T2T's which are loading data into Fact CRM Account Summary has to be modified for ANSI Join conditions to include table DIM\_DATA\_ORIGIN.
4. The Existing process "ACCOUNT PROFITABILITY PROCESS" is modified to have all the T2T's which are loading data into Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN\_RUN\_EXE\_PARAM".
5. A new UMM regular BI Enabled hierarchy with only one level is created based on entity "DIM\_DATA\_ORIGIN".
6. New run definitions will be created for each data source which would be based on the same process "ACCOUNT PROFITABILITY PROCESS". Each run definitions would be set to have data source as the filter using the hierarchy defined earlier.

For example if there are 3 sources then,

- ACCOUNT PROFITABILITY Src1 with "Source1" as the filter
  - ACCOUNT PROFITABILITY Src2 with "Source2" as the filter
  - ACCOUNT PROFITABILITY Src3 with "Source3" as the filter
7. After defining the Run definitions, the run's are executed for the required MIS date. This approach requires as many run definitions for each of the sources which has to be processed at once. Each Run definition execution would be storing its Run Skey value in the target fact table.
  8. Modify the existing DT "FCT\_ACCT\_TRANSFORMATION" for passing additional parameters to PL/SQL function "FN\_FCT\_ACCT\_PFT\_DT".
  9. Grant Select privileges on few tables from Config schema user to atomic schema user.
  10. Create Database views in atomic schema.

Following are the details of the approach:

1. All the T2T's which are loading data into Fact CRM Account Summary has to be modified for ANSI Join conditions to include table DIM\_DATA\_ORIGIN.
  - Navigate to Database Extracts and to the required Application & Data Source.
  - Select T2T T2T\_STG\_CRMAS\_ANNUIITY\_CONTRACTS and Edit the definition.
  - Modify the ANSI join condition by appending the following:
 

```
LEFT OUTER JOIN DIM_DATA_ORIGIN ON
```

```
DIM_DATA_ORIGIN.V_DATA_SOURCE_CODE =
STG_ANNUITY_CONTRACTS.V_DATA_ORIGIN
```

- Save the T2T definition.
- Similarly do the changes for the other T2T's

```
T2T_STG_CRMAS_BILLS_CONTRACTS
```

```
T2T_STG_CRMAS_BORROWINGS
```

```
T2T_STG_CRMAS_CARDS
```

```
T2T_STG_CRMAS_CASA
```

```
T2T_STG_CRMAS_INVESTMENTS
```

```
T2T_STG_CRMAS_LC_CONTRACTS
```

```
T2T_STG_CRMAS_LOAN_CONTRACTS
```

```
T2T_STG_CRMAS_MM_CONTRACTS
```

```
T2T_STG_CRMAS_OD_ACCOUNTS
```

```
T2T_STG_CRMAS_TD_CONTRACTS
```

```
T2T_STG_CRMAS_LEASES_CONTRACTS
```

```
T2T_STG_CRMAS_GUARANTEES
```

```
T2T_STG_CRMAS_TRUSTS
```

```
T2T_STG_CRMAS_COMMITMENTS
```

```
T2T_STG_CRMAS_MUTUAL_FUNDS
```

2. The Existing process "ACCOUNT PROFITABILITY PROCESS" is modified to have all the T2T's which are loading data into Fact Common Account Summary. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN\_RUN\_EXE\_PARAM".

- Edit the "ACCOUNT PROFITABILITY PROCESS".
- Select the "Component" button.
- Navigate to Component>Insertion Rules><Source> section from the LHS menu.
- Select the T2T's which load data into Fact Common Account Summary.

```
T2T_STG_ANNUITY_CONTRACTS_CAS
```

```
T2T_STG_BILLS_CAS
```

```
T2T_STG_BORROWINGS_CAS
```

```
T2T_STG_CARDS_CAS
```

```
T2T_STG_CASA_CAS
```

```
T2T_STG_GUARANTEES_CAS
```

```
T2T_STG_INVESTMENTS_CAS
```

```
T2T_STG_LC_CAS
```

```
T2T_STG_LEASES_CONTRACTS_CAS
```

```
T2T_STG_LOANS_CAS
```

T2T\_STG\_MM\_CAS

T2T\_STG\_OD\_CAS

T2T\_STG\_TD\_CONTRACTS\_CAS

T2T\_STG\_TRUSTS\_CAS

T2T\_STG\_COMMITMENT\_CONTRACTS\_CAS

T2T\_STG\_MUTUAL\_FUNDS\_CAS

- Select Precedence button and select FN\_RUN\_EXE\_PARAM from the drop-down. All the T2T's which are loading data into Fact Common Account Summary are set as Precedence for the existing task "FN\_RUN\_EXE\_PARAM".

T2T\_STG\_ANNUITY\_CONTRACTS\_CAS

T2T\_STG\_BILLS\_CAS

T2T\_STG\_BORROWINGS\_CAS

T2T\_STG\_CARDS\_CAS

T2T\_STG\_CASA\_CAS

T2T\_STG\_GUARANTEES\_CAS

T2T\_STG\_INVESTMENTS\_CAS

T2T\_STG\_LC\_CAS

T2T\_STG\_LEASES\_CONTRACTS\_CAS

T2T\_STG\_LOANS\_CAS

T2T\_STG\_MM\_CAS

T2T\_STG\_OD\_CAS

T2T\_STG\_TD\_CONTRACTS\_CAS

T2T\_STG\_TRUSTS\_CAS

T2T\_STG\_COMMITMENT\_CONTRACTS\_CAS

T2T\_STG\_MUTUAL\_FUNDS\_CAS

- Save the process definition as same version.
3. A new UMM regular BI Enabled hierarchy with only one level is created based on the entity "DIM\_DATA\_ORIGIN".
    - Create a new hierarchy as mentioned in the following:
    - After the hierarchy is defined, Authorize and save the metadata. Ensure data exists in DIM\_DATA\_ORIGIN before saving the metadata.
  4. New run definitions will be created for each data source which would be based on the same process "ACCOUNT PROFITABILITY PROCESS". Each run definitions would be set to have data source as the filter using the hierarchy defined earlier.

For example if there are 3 sources then,

ACCOUNT PROFITABILITY Src1 with "Source1" as the filter

ACCOUNT PROFITABILITY Src2 with "Source2" as the filter

ACCOUNT PROFITABILITY Src3 with "Source3" as the filter

Create a new run definition say "ACCOUNT PROFITABILITY Source 1" as the run definition name.

Code: AccountProfitabilitySrc1

Name: Account Profitability Source 1

Type: Base Run

- Select Add>Job.
- Select the process modified earlier from the LHS menu.
- Select Add>Run Condition.
- Select Data Origin from the LHS menu.
- After selecting Run Condition and Job, select Next.
- Add the condition to Run Condition by selecting the "Launch Browser".
- Select the required source from the LHS menu.
- Select Save.
- Similarly, new run definitions have to be defined for each source that the user wants to process, by repeating the above mentioned process.

**5. Grant Select privileges on few tables from Config schema user to atomic schema user.**

pr2\_run\_object\_member

metadata\_master

metadata\_element\_master

metadata\_attribute\_master

metadata\_locale\_master

Execute the script "ConfigPrevsRunFilter.sql" by modifying the file. Replace the values ##ATOMIC\_USER## with actual Atomic Schema user.

**ConfigPrevsRunFilter.sql**

```
GRANT SELECT ON pr2_run_object_member to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_master to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_element_master to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_attribute_master to ##
/
ATOMIC_USER##/GRANT SELECT ON metadata_locale_master to
##ATOMIC_USER##
/
```

**6. Create Database views in atomic schema which are required.**

Execute the script "Create\_Run\_Filter\_Views.sql" by modifying it. Replace the values **##CONFIG\_USER##** with actual Config schema user and **##INFODOM##** with the Infodom name.

Create\_Run\_Filter\_Views.sql

```
CREATE OR REPLACE VIEW Vw_pr2_run_object_member AS SELECT * FROM
##CONFIG_USER##.pr2_run_object_member WHERE v_infodom_name =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_master AS SELECT * FROM
##CONFIG_USER##.metadata_master WHERE dsn_id = '##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_element_master AS SELECT * FROM
##CONFIG_USER##.metadata_element_master WHERE v_metadata_infodom =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_attribute_master AS SELECT * FROM
##CONFIG_USER##.metadata_attribute_master WHERE v_metadata_infodom =
'##INFODOM##'
/
CREATE OR REPLACE VIEW VW_metadata_locale_master AS SELECT * FROM
##CONFIG_USER##.metadata_locale_master WHERE metadata_infodom =
'##INFODOM##'
/
```

7. Modify the PL/SQL function "FN\_FCT\_ACCT\_PFT\_DT." Execute the script "FN\_FCT\_ACCT\_PFT\_DT.sql" in atomic schema.
8. Modify the existing DT "FN\_FCT\_ACCOUNT\_PFT" for passing additional parameters to PL/SQL function "FN\_FCT\_ACCT\_PFT\_DT".
  - Navigate to Post Load transformation screen.
  - Edit the definition "FCT\_ACCT\_TRANSFORMATION".
  - Navigate to Stored Procedure section.
  - Copy and paste the contents of the file "fn\_fct\_acct\_pft.sql" into Stored Procedure Editor Section and save the definition.
  - Execute the Run's as required.

# 11 Steps to Upload Files to SFTP Server Shared Mount

1. Place the encryption key used to encrypt the files in the below location:  
**/u02/data\_files/**  
**KeyName should be EncKey.**
2. Upload the data files from the customer (.csv encrypted files) to the below directory:  
**/u02/data\_files/StagingArea**
3. All the files will be scanned and infected files will be present in the following directory:  
**/u02/data\_files/InfectedFiles**

## NOTE

If files are not present in the **InfectedFiles** directory, then connector batch can be executed.

If some files are present in the **InfectedFiles** directory, then verify for the infection, perform the clean-up and re-upload in the staging area.

If connector batches are failing after uploading the files in staging area and there is no file in the **InfectedFiles** directory, then contact the Support Team.

