Oracle Financial Services
Enterprise Modeling
User Guide

Release 8.0.1.0.0
July 2015

ORACLE®

FINANCIAL SERVICES

ORACLE®

Oracle Financial Services
Enterprise Modeling
User Guide

Release 8.0.1.0.0
July 2015

Part Number: E64216-01

Contents

APPENDIX A   *... **131***

APPENDIX B   *... **141***

APPENDIX C   *... **147***

List of Figures

List of Tables

Preface

The *Oracle Financial Services Enterprise Modeling User Guide* explains the concepts of Oracle Financial Services Enterprise Modeling and provides step-by-step instructions for navigating through the application.

This chapter discusses the following topics:

- Audience
- How this Guide is Organized
- Related Documents
- Conventions Used in This Guide

Audience

The *Oracle Financial Services Enterprise Modeling User Guide* is designed for various users of Oracle Financial Services Enterprise Modeling. This guide is intended for the following users:

- **Business Analysts** (BA) who are instrumental in solution designing and creating statistical models using historical data.
- **System Administrators** (SA) who are instrumental in maintaining and executing batches, making the application secure and operational, and configuring the users and security of the application.

How this Guide is Organized

The *Oracle Financial Services Enterprise Modeling User Guide* includes the following topics:

- Chapter 1: About Oracle Financial Services Enterprise Modeling provides an overview of the Oracle Financial Services Enterprise Modeling, its features, the workflow, user roles, privileges, and actions.
- Chapter 2: General Features, explains the general features of Oracle Financial Services Enterprise Modeling. These are the common User Interface (UI) features that are found across all modules.
- Chapter 3: Getting Started explains how to access and login to the application.
- Chapter 4: Managing Sandbox gives detailed information on Sandbox module, user roles, creating Sandbox definitions, maintaining the Sandbox, and so on.
- Chapter 5: Managing Techniques describes techniques registration, creating, authorizing, editing, and deleting a technique.
- Chapter 6: Managing Variables explains about creating, editing, viewing, and deleting variables.
- Chapter 7: Managing Modeling gives detailed information about creating, editing, viewing, and deleting models. It also explains how to execute and deploy models.
- Chapter 8: Managing Stress Testing describes about creating, editing, viewing, and deleting stress definitions. It also provides information about scenario management, the variable shock library, and so on.

Related Documents

This section identifies additional documents related to OFS AAAI Application 8.0.1.0.0. You can access Oracle documentation online from the OHC Documentation Library for OFSAAI 8.0.1.0.0.

- *Oracle Financial Services Advanced Analytical Applications Infrastructure (OFS AAAI) Application Pack 8.0.1.0.0 Installation and Configuration Guide*
- *Oracle Financial Services Analytical Applications Infrastructure Environment Check Utility Guide*
- *Oracle Financial Services Analytical Applications Infrastructure User Guide*
- *Oracle Financial Services Enterprise Modeling User Guide*
- *Oracle Financial Services Analytical Applications Infrastructure Administration Guide*
- *Oracle Financial Services Forms Manager User Guide*

To find additional information about how Oracle Financial Services solves real business problems, see our Web site at www.oracle.com/financialservices.

Conventions Used

Table 1 lists the conventions used in this guide.

Table 1. Conventions Used in this Guide

Convention	Meaning
Italics	- Names of books, chapters, and sections as references - Emphasis
Bold	- Object of an action (menu names, field names, options, button names) in a step-by-step procedure - Commands typed at a prompt - User input
Monospace	- Directories and subdirectories - File names and extensions - Process names - Code sample, including keywords and variables within text and as separate paragraphs, and user-defined program elements within text
<Variable>	Substitute input value

CHAPTER 1 — *About Oracle Financial Services Enterprise Modeling*

This chapter provides an insight to Oracle Financial Services Enterprise Modeling. This chapter includes the following topics:

- Overview
- Concepts of Oracle Financial Services Enterprise Modeling
- Components of Oracle Financial Services Enterprise Modeling
- User Groups and Entitlements

Overview

The core of Financial Institutions Models is Risk, Marketing, Financial Crime and Enterprise Performance Analytical Applications. These models include traditional statistical techniques, modern machine learning methods, computational and simulation models. Oracle Financial Services Enterprise Modeling leverages popular statistical platforms such as the R platform and presents a framework for developing, deploying and managing models at the enterprise level, for financial institutions.

As an enterprise modeling toolkit, Oracle Financial Services Enterprise Modeling enables an institution's IT policies to be enforced while providing flexibility and freedom that Data Scientists and Statistical Modelers desire. Administrative users grant analysts and modelers access to sandboxes - particular analytical subject areas of interest along with a subset of production data - for model building. Validated and approved models may then be promoted from sandboxes to the enterprise model repository. Models in the repository may then be woven into in analytical application flows crafted by mixing data management tasks, model execution and deterministic business logic.

As the use of models proliferate and as modeling becomes a self-service idea within financial institutions, authorized modelers may publish techniques -- parameterized templates of models that serve as building blocks or standardized blueprints for models - so that the best ideas from experienced modelers are captured and reused within the firm. Oracle Financial Services Enterprise Modeling supports techniques developed using R, C++ or Java or Python script languages.

Unique to the needs of large and medium sized financial institutions is the need to project capital levels under a variety of macroeconomic conditions, in order to assess the institution's financial strength under different stress scenarios. The Stress Testing Framework within Oracle Financial Services Enterprise Modeling enables risk and finance officers to define various shocks and scenarios and to apply these conditions uniformly across different model execution runs.

Data lineage and traceability are central to a Financial Intuition's governance process. Oracle Financial Services Enterprise Modeling Application together with the pre-requisite Oracle Financial Services Analytical Applications Infrastructure Application provides a toolkit for developing complete end-to-end analytical applications with data lineage and traceability enabled at every step along the analytical workflow.

Key Features and Advantages

The key features and advantages of Oracle Financial Services Enterprise Modeling are:

Table 2. Features

Model Registry	Centrally Manage and Control all in the enterprise
Statistical Technique Library	Catalog and publish best modeling practices via reusable templates, which are to be used as building blocks for models
Track model usage across applications	Real-time view of dependencies such as which applications use which models; which variables are used a model, etc. Perform what-if impact analysis for changes to data sets, variables and models.
Modeling Sandboxes	Provision sandboxes with subsets of production data for modelers giving modelers complete freedom to build, test and calibrate models using production data, while keeping production environment locked-down for security and compliance needs. Readily deploy built models to locked down production via a built-in workflow.
Integrated Platform (With Oracle Financial Services Analytical Applications Infrastructure)	Statistical modeling, Data management and application deployment integrated in a unified platform so that models may be deployed for use within analytical applications quickly
Execute Models local to data	In-database (Oracle) execution of statistical models for high performance and scalability.
Foundation for Enterprise Stress testing	Centrally define variables and shocks. Catalog Stress scenarios for use in stress testing. Enable baseline and stress executions of models defined within the platform

Enterprise-R support with Oracle Financial Services Enterprise Modeling

Oracle Financial Services Enterprise modeling application leverages the R statistical platform. R is an open source statistical programming language and environment for computing and graphics. For more information about R, see the R Project for Statistical Computing at http://www.r-project.org. Oracle R Enterprise (ORE) is an in-database implementation of the R platform. ORE is a component of the Oracle Advanced Analytics Option of Oracle Database Enterprise Edition. ORE enables running R models within the Oracle Database Environment. For more information on ORE, see the Oracle R Enterprise User Guide.

Oracle Financial Services Enterprise Modeling Application supports ORE based models and techniques, that is, models defined and registered within the platform may be executed on an ORE instance. R models may be scripted within the platform or may be imported into the platform from previously existing R scripts.

Concepts of Oracle Financial Services Enterprise Modeling

Oracle Financial Services Enterprise Modeling is built specifically to meet many of the needs of large Financial Institutions where external regulatory and internal governance policies.

Models may only be built and tested in a sandbox environment. A sandbox has to be provisioned and authorized for use (usually by an administrator) before it is visible to modelers. Any number of sandboxes may be provisioned, but generally, an enterprise may provision a sandbox for each department or analytical team. For e.g, there may be an LGD sandbox consisting of data needed to build and validate LGD models, and a separate one for PD modeling. Such segregation of modeling teamwork areas is desirable in practice, but it is not a requirement: here may be as few as a single sandbox for the entire organization

Sandboxes are provisioned along with data required for modeling. Tools are provided to aid administrators in provisioning sandboxes with subsets of production data. Datasets and variables abstract physical data sources from the modeler, and data in the sandbox is exposed to modelers using via datasets and variables. Models are built against datasets and variables not physical data tables and columns. i.e, the underlying data is exposed as a logical dataset within the application, and modelers need not write any database specific queries to obtain data for modeling.. It is generally an administrative task to define datasets, and have a menu of datasets available for the modelers. When a sandbox is provisioned, one or more datasets can be associated with the sandbox.

Models in a sandbox can be changed (created/ edited) by anyone with access to the sandbox. Model versions are preserved in the sandbox along with execution and output histories. Once a model has been validated in the sandbox environment, the modeler may request that model to be promoted to the locked-down "production" environment, and once promoted, the promoted model cannot be altered.

Modelers may create new models by using a registered technique from the technique library, as a template. A technique is simply a parameterized and reusable script. An enterprise may publish a menu of techniques and require that modelers use those techniques as the foundation for models. The act of model building is then reduced to selecting an appropriate technique, binding the technique to the appropriate dataset and variables and providing runtime parameters to the script. Generally, a central data sciences team within an enterprise or a department is responsible for publishing techniques.

Not all models can be built using published techniques, and so an alternate way to build a model is to write an R script 'from scratch' and execute the script as a model. Regardless of how the model is built, the model must be bound to dataset and variables before it can access data.

 A business analyst may decide to include a published model (that is a promoted model) in an application flow. For e.g., a capital computation application flow may include many steps, some of which may be steps to execute statistical models for computing PD and LGD. As the Oracle Financial Services Enterprise Modeling application is fully integrated with the Oracle Financial Services Analytical Applications Infrastructure (AAI), models promoted from Oracle Financial Services Enterprise Modeling application are available as tasks in AAI, and so can be included in any orchestrated execution of tasks (application run). Note that within a sandbox, executable models are also made available as tasks private to the sandbox, and so can be included in sandbox specific orchestration of tasks.

Oracle Financial Services Enterprise Modeling Workflow

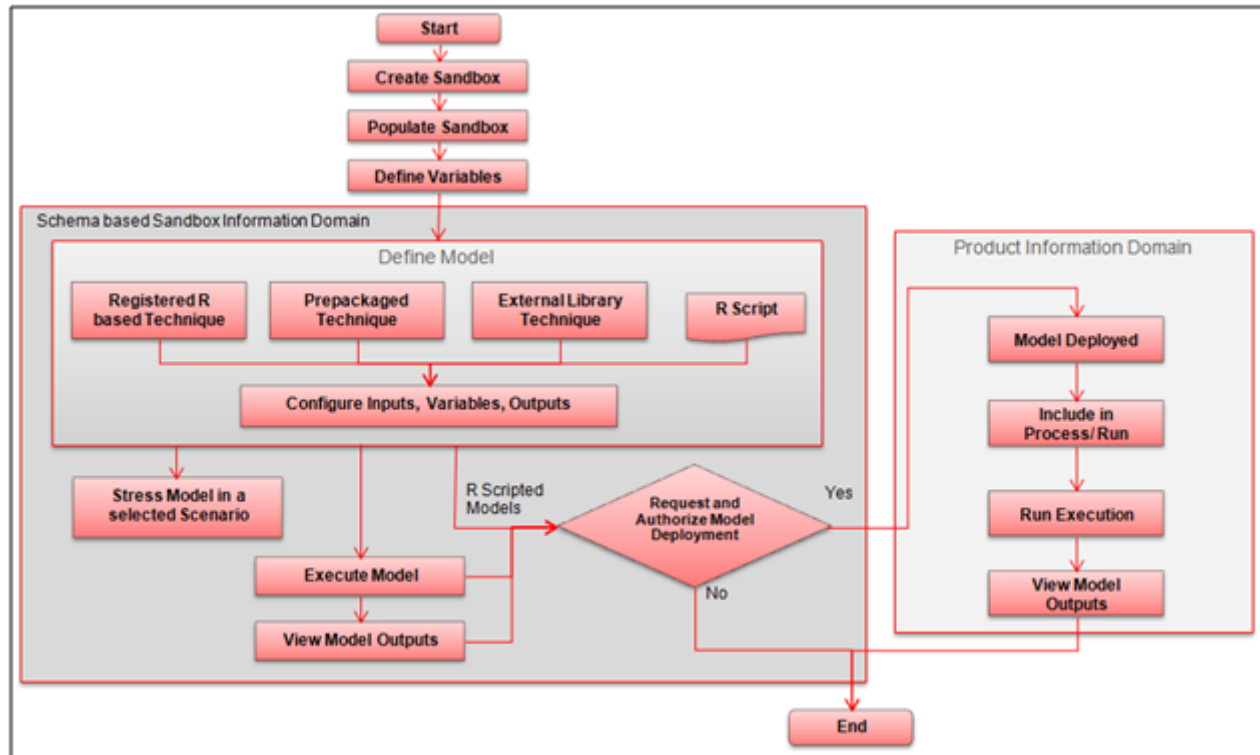**Figure 1. Oracle Financial Services Enterprise Modeling Workflow**

Components of Oracle Financial Services Enterprise Modeling

This section describes the components of Oracle Financial Services Enterprise Modeling.

The following are the components of Oracle Financial Services Enterprise Modeling application:

- Sandbox
- Sandbox Maintenance
- Techniques Registration
- Variable Definition
- Modeling
- Stress Testing

Sandbox

Sandbox in Oracle Financial Services Enterprise Modeling refers to a restricted modeling environment, where the Data model is uploaded. It is implemented as an information domain.

The following information domains are required to perform operations in Oracle Financial Services Enterprise Modeling.

- **Sandbox Information Domain**: This information domain is used to create and execute business models. Based on the execution status and the generated output, you can verify the results and deploy the model into production.
  You can create the following two types of Sandbox in the Oracle Financial Services Enterprise Modeling application.

  - **Logical Sandbox**: A Logical Sandbox is a restricted environment within the Production information domain, where the data is uploaded to the logical sandbox within the Production information domain. You can create a sandbox as a Logical Sandbox, where data model upload or data population is not required. The logical sandbox refers to the set of tables present in the datasets you have selected during sandbox definition. After creating the logical sandbox definition, map it to the required user groups from the Sandbox Maintenance window. After the Users are mapped, you can create model definitions in the defined logical sandbox.

  - **Physical Sandbox**: A Physical Sandbox is a restricted environment outside the Production information domain where there is actual movement of data from the production information domain to the physical sandbox. Creating a sandbox with multiple datasets eliminates the need for having a sandbox definition for each dataset. You can upload the data model while defining the sandbox, or upload it later using the Import Model option in Unified Metadata Manager. When you save the sandbox definition, the required tables are created in the sandbox information domain. However, data present in the tables are copied only after authorizing sandbox population from the Sandbox Maintenance.

- **Production Information Domain**: This information domain allows you to request for model execution, and generate model outputs.

Note: Ensure the data model of the sandbox information domain should be a sub-set of the data model of the production information domain.

Sandbox Maintenance

Sandbox Maintenance helps you to populate the data to the tables in the Sandbox information domain, based on the dataset and the filters in the Sandbox definition. You have the option to do complete or incremental sandbox data population. However, for logical sandbox definitions, data population is not required.

The Oracle Financial Services Enterprise Modeling application enables you to synchronize the different versions of a Data Model which exists in a Production and Sandbox Information Domain through Incremental Data Model Upload. You can refresh the details and fetch the incremental data model changes from Production to Sandbox Information Domain.

Techniques Registration

Technique is a set of generalized statistical algorithms which can be used to build analytical models. Previously, the Oracle Financial Services Enterprise Modeling application was completely based on techniques developed by NAG and deployed. But now, it helps develop techniques using R script and ORE functions, prepackaged techniques, and external library techniques. An external library technique is based on third party algorithms, which can be a library or executable. At present, only C/C++ and Java programming languages are supported as Third Party Algorithms.

Variable Definition

Variable refers to a logical set of attributes that are likely to change based on the selected parameters. In a modeling environment variable plays a vital role in filtering the model parameters and to derive an estimate based on historical data. Variables are defined in production information domain.

Modeling

Modeling refers to the process of designing a prototype based on a structured data model, considering all the variables for statistical analysis and to simulate real events and processes.

You can use the Modeling utility to measure and quantify risk. You can use the pre-defined models to predict business trends and validate the existing models. You can use R scripting (using R functions as well as ORE functions) or Open R to create business models. Refer *Enterprise-R support with Oracle Financial Services Enterprise Modeling* section for more information.

Oracle Financial Services Enterprise Modeling enables you to run and execute R functions as well as ORE functions in the database, thereby greatly increasing scalability and performance.

You can create models in a Schema Based sandbox or Logical sandbox.

Note: For models created in logical sandboxes, the production and sandbox information domains are the same.

Stress Testing

Stress testing is an integral part of a bank's risk measurement system and plays an important role in estimating the effects of potential financial crises on a bank's operations. Stress Testing or risk estimation technique, refers to the process of examining the stability of a system or entity in adverse conditions. It involves testing beyond normal operational capacity, often to a breaking point, in order to observe the results. It also helps banks conduct analysis to estimate the impact of movements in the variables on specific measures such as profitability and capital adequacy.

The Stress Testing utility supports the stress testing requirements across the entire suite of OFSAA products. It allows banks to define shocks and assess the impact of such shocks across multiple business areas.

The two commonly accepted forms of Stress Testing are:

- **Sensitivity Analysis**: Shocks are applied on a single variable.

- **Scenario Analysis**: A scenario is defined as a shock to a single variable or a collection of shocks on multiple variables. Scenario analysis involves applying simultaneous shocks on multiple variables to assess the impact of scenario on a measure or a set of measures. Scenarios are further classified into the following categories:

- **Historical Scenarios:** These scenarios replicate the past events docket. They are defined by specifying shocks to variables such that they replicate the movement seen during historical events.
  For example, the user may define a scenario that replicates the movement in stock market indices as observed during the catastrophic event. This scenario can then be applied to the current trading book portfolio of the bank to estimate the loss that might be incurred if a catastrophic event occurs. However, the historical scenarios may not cover the entire range of potential adverse conditions

- **Hypothetical Scenarios**: These scenarios are based on user judgment and addresses the other possible adverse movements in the variables.

User Groups and Entitlements

The following table gives the details about the User Groups in the Oracle Financial Services Enterprise Modeling Application.

Table 1. User Groups

User Group Name	User Group Description
Modeling Administrator	User mapped to this group will have access to all the menu items for Enterprise modeling and will be have authorization rights for sandbox population, model deployment and modeling technique authorization.
Modeler Group	User mapped to this group will have access to all the menu items for Enterprise modeling but will not have authorization rights for sandbox population, model deployment and modeling technique authorization.

CHAPTER 2   *General Features*

This chapter explains the general features of Oracle Financial Services Enterprise Modeling application UI. These are the common features that are found across the modules of Oracle Financial Services Enterprise Modeling. It describes the organization of the user interface and provides step-by-step instructions for navigating through the application.

This chapter includes the following topics:

- Home Page Components
- Oracle Financial Services Enterprise Modeling Common Elements
- Acronyms

Home Page Components

The Home Page contains the following sections.

- Top Menu
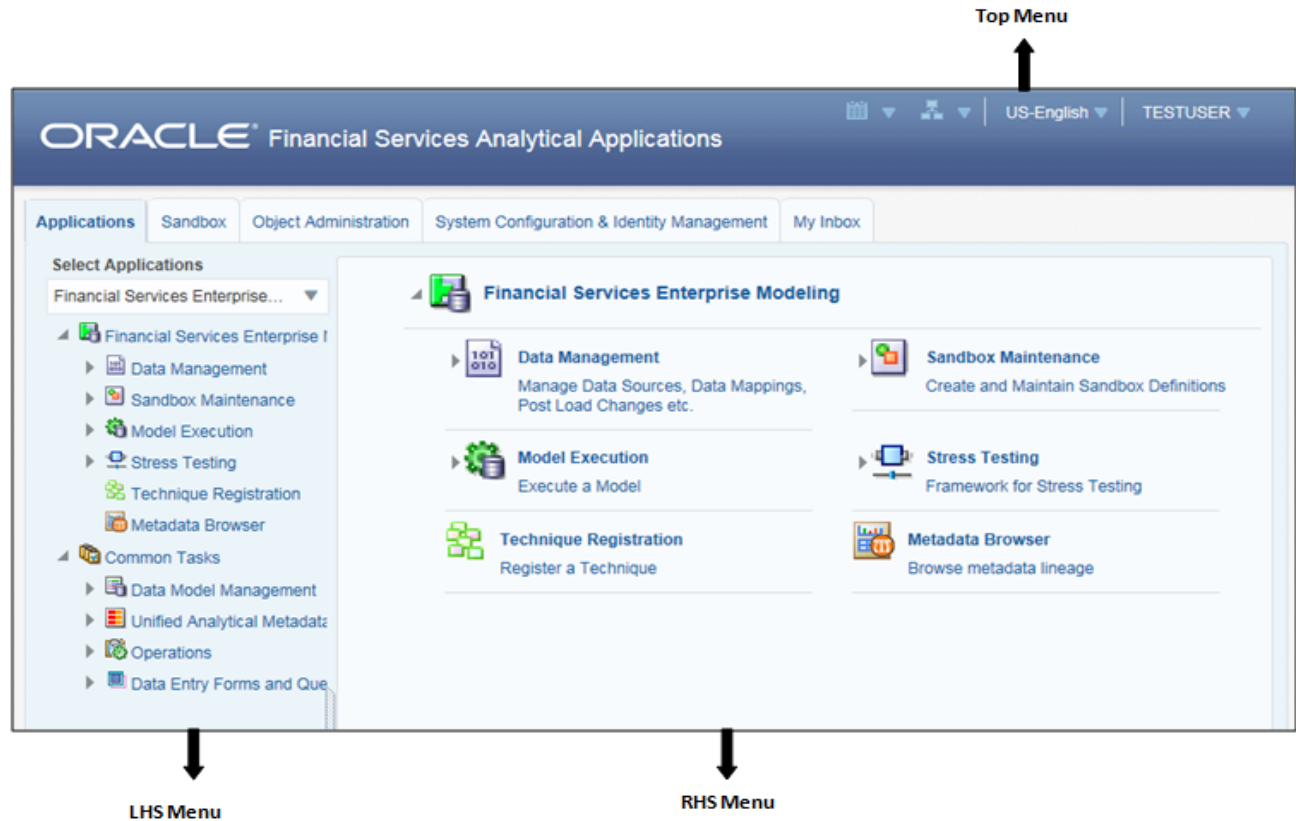- Left Hand Side (LHS)
- Right Hand Side (RHS)

Figure 2. Home Page

Top Menu

This section displays the logged in User information and Information Domain. The Top menu contains the following buttons:

- **Last Login Date**: This option allows you to view the last login date and time. It also displays the last failed login date and time.

- **Connected To**: This option allows you to view the Information Domain you are connected to and the Environment Details set in the Configuration page.

- **Language**: This option allows you to view the language selected.

- **User Name**: Displays the user name of the person logged in. click the user name to view the following options:

  - **Preferences**: This option allows you to set the Home Page.

  - **About**: This option displays the copyright information.

  - **Change Password**: This option opens the Change Password window.

  - **Logout**: This option allows you to log out of the application.

Left Hand Side (LHS)

This section displays the menu which allows you to navigate to the required module. The items displayed in the Menu depend upon the access rights of the logged in user.

You can select an Information Domain to which the Application is connected to, from the Connected To drop-down list. The Menu is refreshed based on the selected Information Domain.

The LHS menu changes depending on the application selected from the **Select Application** drop-down list. For more details, refer to the *Oracle Financial Services Analytical Application Infrastructure User Guide*.

When you select **Oracle Financial Services Enterprise Modeling**, the following menu items are displayed in the Home Page:

- Application Tab

- Sandbox Tab

- Object Administration Tab

- System Configuration & Identity Management Tab

- Inbox Tab

Application Tab

The following menu items are available in the Application tab:

- **Financial Services Enterprise Modeling**

    - Data Management
      For more information, refer to the *Oracle Financial Services Analytical Applications Infrastructure User Guide 8.0 Releas*e.

    - Sandbox Maintenance

    - Model Execution

    - Technique Registration

    - Stress Testing

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

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

    - Data Model Management

    - Unified Analytical Metadata

    - Operations

    - Data Entry Forms and Queries

Sandbox Tab

Select the required Sandbox from the **Select Sandbox** drop-down list. The following menu items are available in the Sandbox tab:

- Modeling
- Model Creation
- Model Execution Status
- Model Output
- Model Deployment

Note: The Sandbox options in the Select Sandbox drop-down list are available only for the enabled applications.

Object Administration Tab

Select the required Information Domain from the **Select Information Domain** drop-down list. For more information, refer to the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.
The following menu items are available in the Object Administration Tab:

- Financial Services Analytical Applications Infrastructure: Object Administration

    - Object Security
    - Object Migration
    - Translation Tools
    - Utilities
    - Metadata Browser

System Configuration & Identity Management Tab

Select **Administration and Configuration** from the **Financial Services Analytical Applications Infrastructure** drop-down list. For more information, refer to the *Oracle Financial Services Analytical Applications Infrastructure User Guide 8.0 Release*. The following menu items are available in the S ystem Configuration & Identity Management tab.

- System Configuration

    - Database Server
    - Application Server
    - Web Server
    - Database Details
    - OLAP Details
    - Information Domain
    - Configuration
    - Manage OFSAA Product License(s)

- Identity Management

Inbox Tab

The following menu items are displayed in the Inbox Tab.

- My Tasks

Right Hand Side (RHS)

This section displays user options available for the application selected.

Oracle Financial Services Enterprise Modeling Common Elements

This section describes the common screen elements of Oracle Financial Services Enterprise Modeling.

The following table lists the various icons and buttons in the Oracle Financial Services Enterprise Modeling user interface:

Table 2. Common Screen Elements
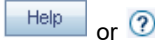
Button	Description
	To create a function.
	To view the details of a function.
	To edit the details of a function.
	To clear the fields and reset to default values.
	To select a new member.
	To select a filter / run condition/ define sub process.
	To select a source / component / job.
	To select a hierarchies / measures / job condition.
	To set precedence for members.
	To execute a Run definition.
	To select hierarchical members
	To delete a function.
	To select the entities.
	To validate grid data.
	To save the details.
/ /	To view the properties.

Table 2. Common Screen Elements

Button	Description
	To refresh the grid details.
	To erase a specific value.
	To define an expression.
	To generate source model.
	To add attributes / add Source Configuration / Authorize a function.
	To generate Source Models.
	To generate a logic and view the SQL query / check syntax of the stored procedure.
	To add the source database configuration details.
	To view the dependencies of the selected Object.
/	To export data.
	To trace a definition details.
	To receive instant on-window help.
	To view the log.
	To view the
	To specify a date using calendar.
	To view Dependencies.
	To run the object migration rule.
	To interrupt the object migration rule.
/	To add and view the source database configuration details
/ /	To authorize or reject a function / definition.
/	To map / un-map source tables to columns.
	To view the time dependencies.
	To view the pagination option.

Table 2. Common Screen Elements

Button	Description
	To view SQL statement.
	To view and enter the details in the Expression window.
	To create a Rule function.
	To open and view the rule details.
	To save a Rule function.
	To search / find a member.
	To save with customized details.
	To view the rule properties.
	To map between hierarchies.
	To select a member.
	To deselect a member.
	To sort in ascending order.
	To sort in descending order.
Help or	To access the documentation resources.

Acronyms

The following table describes the acronyms commonly used in this application.

Table 3. Acronyms

Acronyms	Description
BA	Business Analysts
Infodom	Information Domain
LHS menu	Left hand side menu
OFSAAI	Oracle Financial Services Analytical Application Infrastructure
OFSAA	Oracle Financial Services Analytical Applications
Production Infodom	Production Information Domain
RHS menu	Right hand side menu

Table 3. Acronyms

Acronyms	Description
Sandbox Infodom	Sandbox Information Domain
SA	System Administrator
URL	Uniform Resource Locator

Getting Started

This chapter explains different steps to login to the Oracle Financial Services Enterprise Modeling application.

The following sections are included in this chapter:

- Logging into the Oracle Financial Services Enterprise Modeling Application
- Changing Password

Logging into the Oracle Financial Services Enterprise Modeling Application

Once the application is installed and configured, you can access Oracle Financial Services Enterprise Modeling Application.

To access Oracle Financial Services Enterprise Modeling, follow these steps:

1. Enter the OFSAAI URL in your browser. The Login page displays.

Figure 3. Oracle Financial Services Enterprise Modeling Application Login Page

2. Select the **Language**.

3. Enter your **User ID** and **Password**.

4. Click **Login**. The landing page is displayed.

5. Select **Oracle Financial Services Enterprise Modeling** from the **Select Applications** drop-down list. The Enterprise Modeling Application Home Page is displayed.
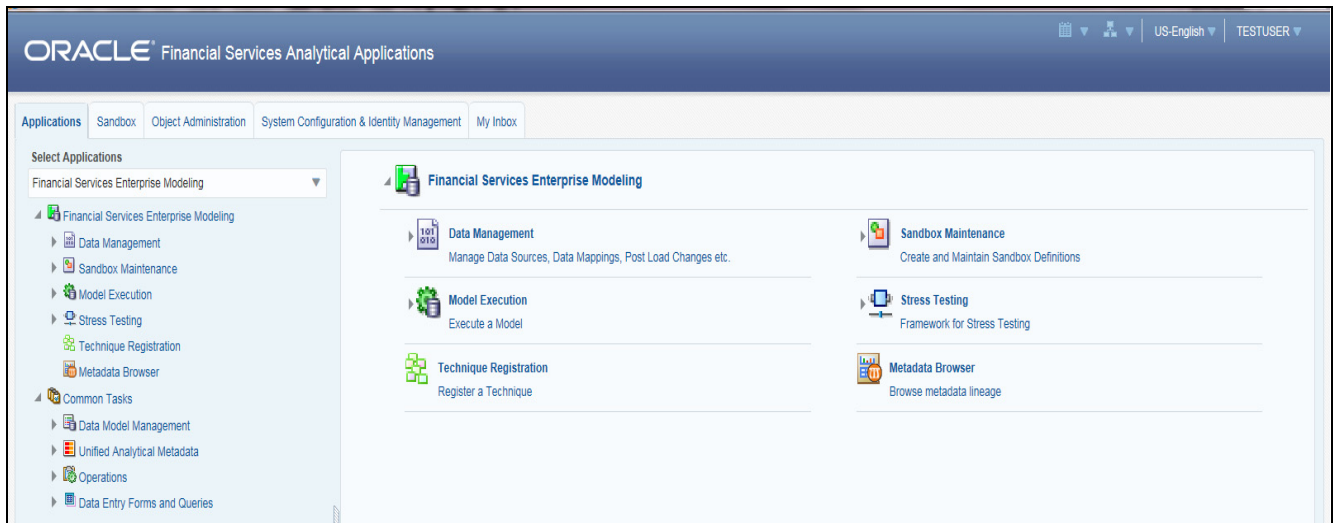
Figure 4. Home Page

Changing Password

The first time you login to the application, the Change Password window is displayed. Alternatively, you can also choose to change the password any time.

To change password, follow these steps:

1. Click **Change Password** in the Top Menu and the Change Password window is displayed.

Figure 5. Change Password Window

2. Enter the **User ID**.

3. Enter the **Old Password**.

4. Enter **New Password** and re-enter in the **Confirm Password** field.

5. Click **OK**.

Note:

- Passwords are displayed as asterisks while you enter. This is to ensure that the password is not revealed to other users.

- Ensure that the entered password is at least six characters long.

- The password must be alphanumeric with a combination of numbers and characters.

- The password should not contain spaces.

- Passwords are case sensitive and ensure that the Caps Lock is not turned ON.

- By default, the currently used password is checked for validity if password history is not set.

- The new password should be different from previously used passwords based on the password history, which can be configured. If you encounter any of the following problems, contact the System Administrator:

- Your user ID and password are not recognized.

- Your user ID is locked after three consecutive unsuccessful attempts.

- Your user ID has been disabled.

- Guest user cannot change the password.

CHAPTER 4 *Managing Sandbox*

This chapter provides information about Sandbox in the Oracle Financial Services Enterprise Modeling application and step-by-step instructions to use this section.

This chapter includes the following topics:

- About Sandbox
- Workflow
- User Roles
- Managing Sandbox

About Sandbox

Sandbox is a restricted environment in which you can analyze the data on trial and errors basis and come up with actual analysis that helps predict the risks and business opportunities for banking institutions. You can create a sandbox by selecting the required datasets and the information domain which you want to implement as the sandbox.

Workflow

This section explains the workflow of Sandbox.

There are two types of Sandbox.

- Logical Sandbox
- Physical Sandbox

Logical Sandbox Workflow

The following image gives a detailed workflow of Logical Sandbox.
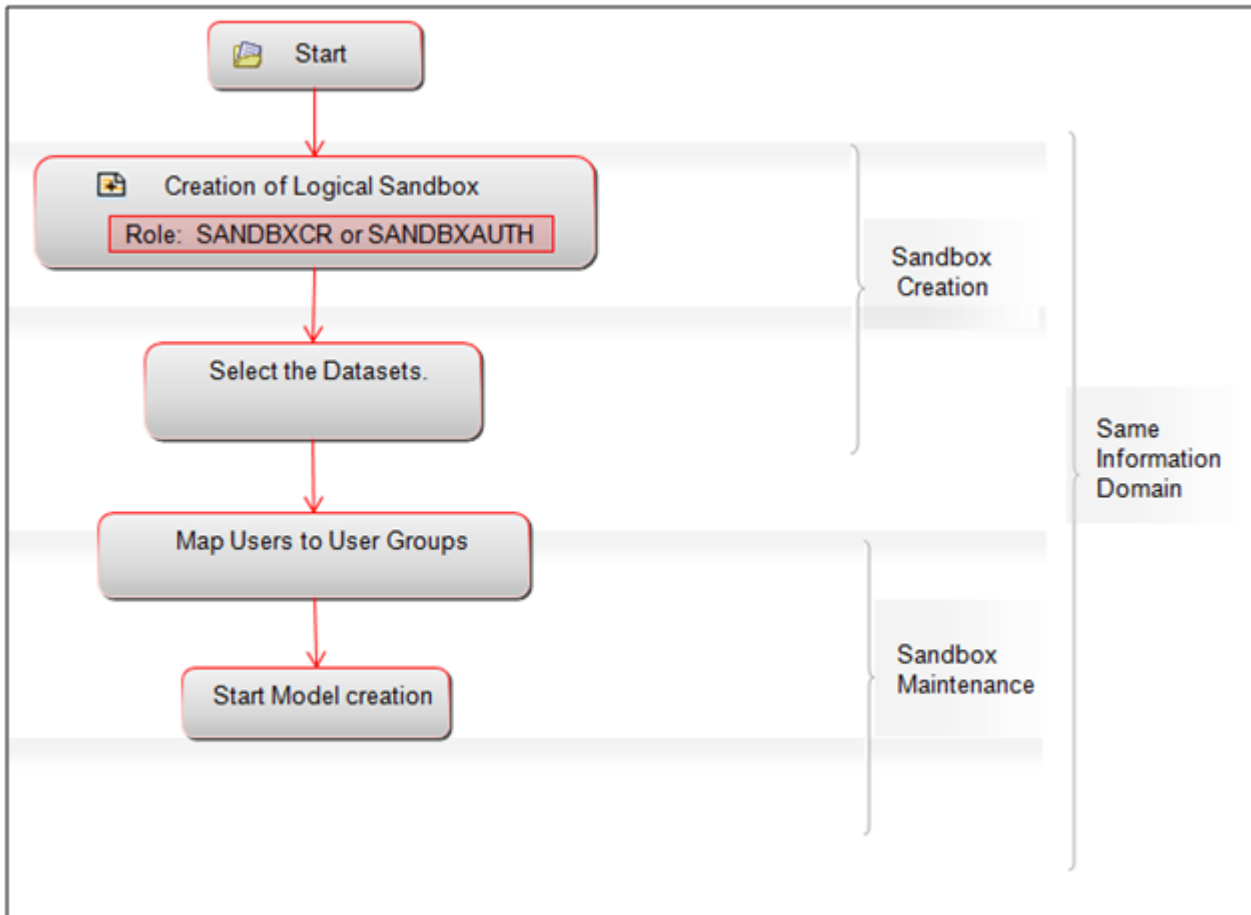
Figure 6. Logical Sandbox Workflow

Physical Sandbox Workflow

The following image gives a detailed workflow of Physical Sandbox.
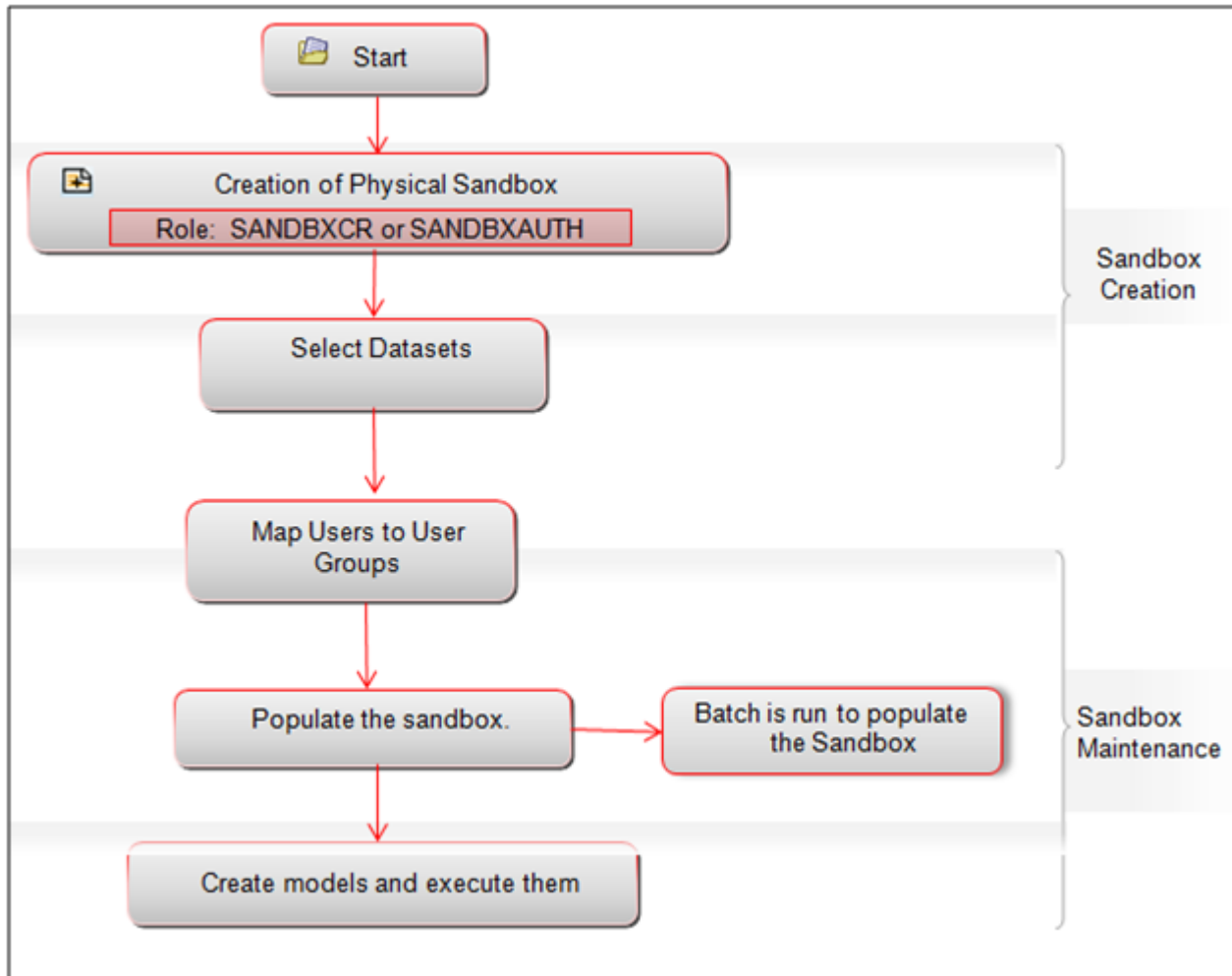
Figure 7. Physical Sandbox Workflow

User Roles

This section explains about the user roles required to define a Sandbox.

To create a sandbox, you must have the following use roles:

- SANDBXCR
- SANDBXAUTH

An authorizer must authorize the sandbox population and deletion.

To populate and delete a Sandbox, you must have the following user roles:

- SANDBXMOD
- SANDBXAUTH

Note: For more information on User Groups, refer to Appendix B.

Managing Sandbox

This section explains how to manage Sandbox feature in Oracle Financial Services Enterprise Modeling application. The section includes the following topics:

- Accessing Sandbox Definition
- Creating Sandbox
- Viewing Sandbox Definition
- Modifying Sandbox Definition
- Searching Sandbox Definition
- Sandbox Maintenance

Accessing Sandbox Definition

You can access the Sandbox Definition page from the Oracle Financial Services Enterprise Modeling Home page.

To access the Sandbox Definition page, follow these steps:

1. Login to Oracle Financial Services Enterprise Modeling. The Oracle Financial Services Enterprise Modeling Home page is displayed.

2. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

3. Click ▶ to expand the Sandbox Maintenance.

4. Click **Sandbox Creation**. The Sandbox Definition page is displayed.
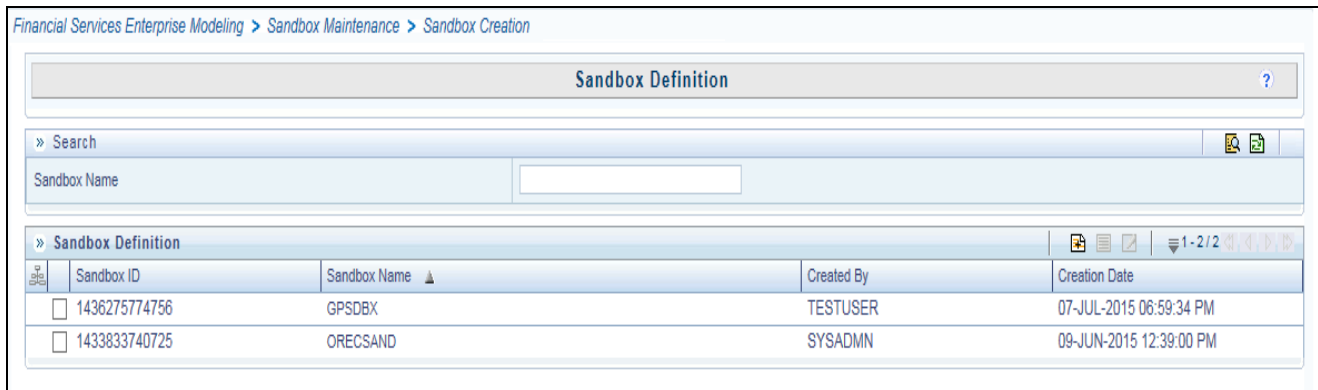
Figure 8. Sandbox Definition Page

Creating Sandbox

This feature allows you to create a sandbox where you can create and execute models.

Note: After Creating a Sandbox you should refresh the screen, so that the newly created sandbox is displayed in the Sandbox tab.

This section includes the following topics:

- Creating Schema-based Sandbox Definition
- Creating Logical Sandbox Definition

Creating Schema-based Sandbox Definition

A Physical Sandbox is a restricted environment outside the Production information domain, where there is actual movement of data from the production information domain to the physical sandbox. Creating a sandbox with multiple datasets eliminates the need for having a sandbox definition for each dataset. A schema-based sandbox is also called a physical sandbox.

To create a schema-based Sandbox, follow these steps:

1. Select [image] from the Sandbox Definition tool bar. The Sandbox Definition window is displayed.

   Note: The Add button is disabled if you have selected any checkbox in the grid.

Figure 9. Schema-based Sandbox Definition page

2. Enter the following details:

Table 3. Schema based Sandbox - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Sandbox Name	Enter a name for the sandbox. **Note:** Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces. **Note:** Enter the alphabetical characters for the Sandbox Name in either uppercase or lowercase. If you enter the Sandbox Name in mixed case, then the EMF-R Execution fails with errors. For example, Sandbox Name can be **ofssdbox180** or **OFSSDBOX180**. If you have existing Sandbox Names in mixed case, then you must edit and update them from the **Database Details** window. For more information, see the **Database Details** section in the Oracle Financial Services Advanced Analytical Applications Infrastructure User Guide.
Sandbox Description	Enter a brief description for the sandbox.
Is a logical definition?	Select **No**.

Table 3. Schema based Sandbox - Field Descriptions

Field	Description
Source Dataset	Select the required datasets. The datasets you want to select should match the Data Model of the sandbox. 1. Click ⬜ and open the Source Dataset Browser. All the Authorized Datasets in the selected production Infodom are listed in the Hierarchical Browser. 2. Select the required dataset from the Datasets pane and click ⮕ . To deselect a dataset select it from the Selected Datasets pane and click ⬅ . 3. Click ▦ to view the details of the selected datasets. 4. Click 🗔 to create a dataset. For more details on creating a dataset, refer *Creating Data Set* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*. **Note**: The Datasets based on Derived Entities are not supported.
Is data model upload required?	By default, the Yes option is selected. ● Select **Yes** to upload the data model while saving the sandbox definition. ● Select **No** if the data model is already uploaded.
Sandbox Location	This is the information domain you want to implement as sandbox. 1. Click ⬚ and open the Sandbox Location Browser. ● If Yes is selected for Is data model upload required? option, the Sandbox Location pane displays the list of the Information Domains where data model is not uploaded and sandbox has not been defined. ● If No is selected for Is data model upload required? option, the Sandbox Location pane displays the list of the Information Domains where data model is uploaded and sandbox has not been defined. 2. Select the required infodom and click ⮕ . To deselect the infodom select it from the Selected Information Domain pane and click ⬅ .
Non Time Hierarchy	1. Click 🖉 in the Non Time Hierarchy toolbar. The Hierarchy Browser displays all the Hierarchies/Filters from each selected Datasets. You can click "+" to expand the node to view the sub levels. 2. Select the required hierarchies and click ⮕ . To deselect hierarchy by select it from the Selected Members pane and click ⬅ . 3. Click **OK**.
Hierarchy member	Click 🖉 in the Hierarchy Member toolbar and specify the members or nodes in the Hierarchy browser. You can click "+" to expand the node to view the sub levels. When you select a hierarchy member, the data belonging to that hierarchy will be copied to the sandbox.

Table 3. Schema based Sandbox - Field Descriptions

Field	Description
Time Hierarchy	1. Click in the Time Hierarchy toolbar. The Hierarchy Browser displays the available Time Hierarchies which are common across all the selected Datasets. You can click "+" to expand the node to view the sub levels. 2. Select the required time hierarchy and click to move it to the Selected Hierarchies pane. To deselect time hierarchies select from the Selected Hierarchies pane and click . 3. Click **OK**. **Note:** Time hierarchy selection is optional. However, if you do not select a time hierarchy, the following will occur: ● Time referencing will not be applicable for model definitions. ● For the model definitions, the variable browser will not show any variable definitions that has got filters. ● The shocking of transition matrix multi factor models will not work in sandboxes ● For models defined using techniques transition matrix - EWMA, MCEM, MC simulations, the execution may fail
Hierarchy Member	Click in the Hierarchy Member toolbar and specify the members or nodes in the Hierarchy browser. You can click "+" to expand the node to view the sub levels. Time hierarchy is used to group data based on time (year or month). When you select time hierarchy member, the data belonging to that time hierarchy will be copied to the sandbox.

3. Click **Save**. The tables of the selected dataset get populated in the sandbox information domain.

Creating Logical Sandbox Definition

A Logical Sandbox is a restricted environment within the Production information domain, where the data is uploaded to the logical sandbox within the Production information domain. This feature allows you to create a sandbox as logical sandbox, where data model upload or data population is not required. The create sandbox definition allows you to create either a schema-based sandbox or a logical sandbox.

To create a logical Sandbox, follow these steps:

1. Select from the Sandbox Definition tool bar. The Sandbox Definition window is displayed.
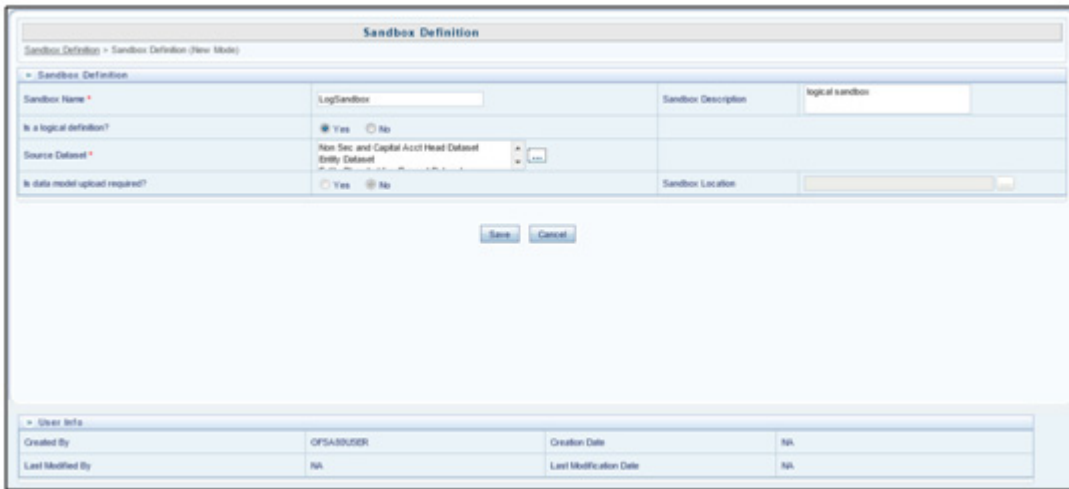
Figure 10. Logical Sandbox Definition page

2. Enter the following details:

Table 4. Logical Sandbox - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Sandbox Name	Enter a name for the sandbox. **Note:** Ensure that there are no special characters like \`, {,},", ', ~, <,>, /, \, and multiple spaces.
Sandbox Description	Enter a brief description for the sandbox.
Is a logical definition?	Select **Yes**.
Source Dataset	Select the required datasets. The datasets you want to select should match the Data Model of the sandbox. 1. Click [...] and open the Source Dataset Browser. All the Authorized Datasets in the selected production Infodom are listed in the Hierarchical Browser. 2. Select the required dataset from the Datasets pane and click [→] . To deselect a dataset select it from the Selected Datasets pane and click [←] . 3. Click [icon] to view the details of the selected datasets. 4. Click [icon] to create a dataset. For more details on creating a dataset, refer *Creating Data Set* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*. 5. Click **Ok.** **Note**: The Datasets based on Derived Entities are not supported.

3. Click **Save**.

Viewing Sandbox Definition

You can view individual Sandbox Definition details at any given point. To view the existing Sandbox Definition details in the Sandbox Definition window, follow these steps:

1. Select the checkbox adjacent to the Sandbox ID.

2. Click ▢ in the Sandbox Definition tool bar.
   The Sandbox Definition View window is displayed with the details such as Sandbox Name, Sandbox Description, Source Dataset, Sandbox Locations, Time, and Non Time Hierarchy and Hierarchical members.

Modifying Sandbox Definition

You can update only the description and Time Hierarchy Member (if it is selected in the definition) of an existing schema based Sandbox Definition. For a logical sandbox, you can add or remove datasets. However, it is not validated whether a model is already defined using the removed dataset.

To modify Sandbox Definition, follow these steps:

1. Select the checkbox adjacent to the Sandbox ID whose details are to be updated.

2. Click ▢ in the Sandbox Definition tool bar. The Edit button is disabled if you have selected multiple Sandbox IDs.

3. Modify the Sandbox Definition details as required. For more information, see the *Creating Sandbox* section.

   Note: You can edit Time and Non-Time Filters in Sandbox Definition Edit page.

Searching Sandbox Definition

The Sandbox Definition page allows you to search the sandboxes that you want to view.

To search for Sandbox, follow these steps:

1. Navigate to the Sandbox Definition page.

Note: By default, the Sandbox Definition page displays all Sandboxes.

2. Enter the name of the sandbox you want to search.

3. Click ▢ . The filtered list is displayed.

Sandbox Maintenance

The Sandbox Maintenance window is used to map user groups to the sandbox definition which enables the users to create models in the logical sandbox.
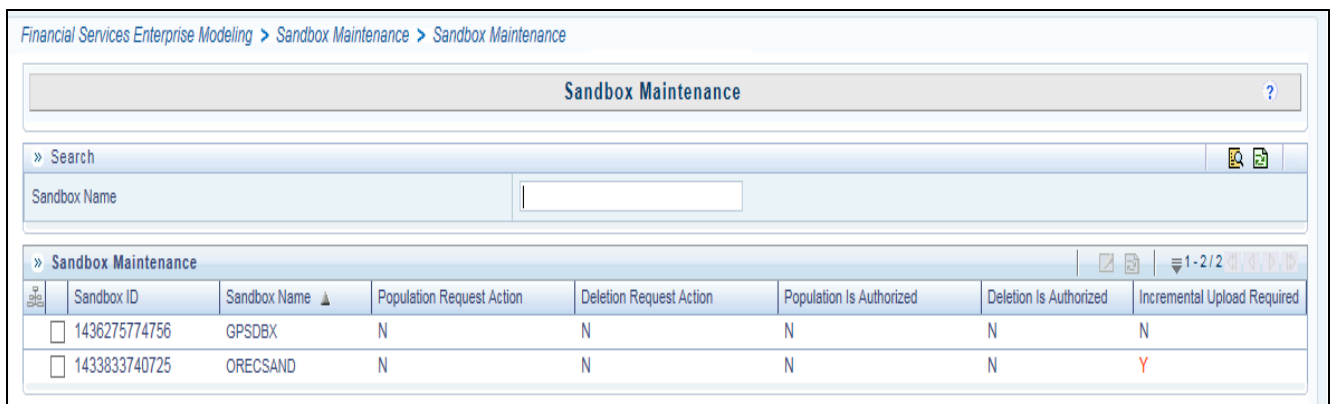
This section includes the following topics:

- Accessing Sandbox Maintenance
- Requesting and Authorizing to Populate Sandbox
- Requesting and Authorizing to Delete Sandbox
- Mapping Sandbox Definition User Group
- Incremental Data Model Upload
- Searching Sandbox Maintenance

Accessing Sandbox Maintenance

You can access the Sandbox Maintenance page from the Oracle Financial Services Enterprise Modeling Home page.

To access the Sandbox Maintenance, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click ▶ to expand the Sandbox Maintenance.

3. Click **Sandbox Maintenance**. The Sandbox Maintenance page is displayed.

Figure 11. Sandbox Maintenance Page

Requesting and Authorizing to Populate Sandbox

Note: This option is not available for 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 ![icon] in the Sandbox Maintenance toolbar. The Edit button is disabled if you have selected multiple checkboxes. The Sandbox Maintenance Edit window is displayed.

   Note: While populating a Sandbox both Time and Non-Time Filters can be edited in Sandbox Maintenance page.

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 is enabled only if your user role is mapped to the function SANDBXAUTH.

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

Requesting and Authorizing to Delete Sandbox

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

1. Select the sandbox which you want to delete and click ![icon] in the Sandbox Maintenance tool bar. 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 the **Delete Sandbox** checkbox.

3. Click the **Authorize** tab, and select the **Delete Sandbox** checkbox to authorize the deletion. This tab will be enabled only if your user role is mapped to the function SANDBXAUTH.

4. Click **Save** to remove the Sandbox Definition.

Mapping Sandbox Definition User Group

Note: This option is available only for Logical Sandbox definitions. Only the users belonging to the mapped user group can use the Logical Sandbox definition.

To map a user group to sandbox definition in the Sandbox Maintenance window, follow these steps:

1. Select the logical sandbox definition and click ![icon] in the Sandbox Maintenance tool bar. The Sandbox Maintenance (Edit Mode) window is displayed.

2. In the Request Action tab, select the **Sandbox Definition- User Group Mapping** checkbox. The Sandbox Definition- User Group Mapping pane is displayed.

3. Click ![icon] in the tool bar. The User Groups window is displayed. All the user groups mapped to the information domain in which the logical sandbox is defined are displayed.

4. Select the user group and click ![icon] . To un-map an information domain select it and click ![icon] .

5. Click **Save**.

Incremental Data Model Upload

The Increment Upload Required column in the Sandbox Maintenance window displays the status as "Y" or "N" depending on the need for Incremental Data Model Upload for the Sandbox definition. The Incremental Data Model Upload button is enabled only when the model versions between Production and Sandbox information Domain are different.

Note: The Incremental Data Model Upload button will not be enabled for the sandboxes in which the *No* option is selected for the Is data model upload required? field, even though there is a difference between the model versions in Production and Sandbox information domains.
Additionally, it will not be enabled for logical sandboxes.

To perform an incremental data model upload, follow these steps:

1. Select the checkbox adjacent to the Sandbox ID which has the Increment Upload Required status as "Y". The status will be Y only when there is a difference in the model versions between Production and Sandbox information Domain.

2. Click ![icon] from the Sandbox Maintenance tool bar. The Incremental Data Model Upload window is displayed with the selected Sandbox definition details inView mode.
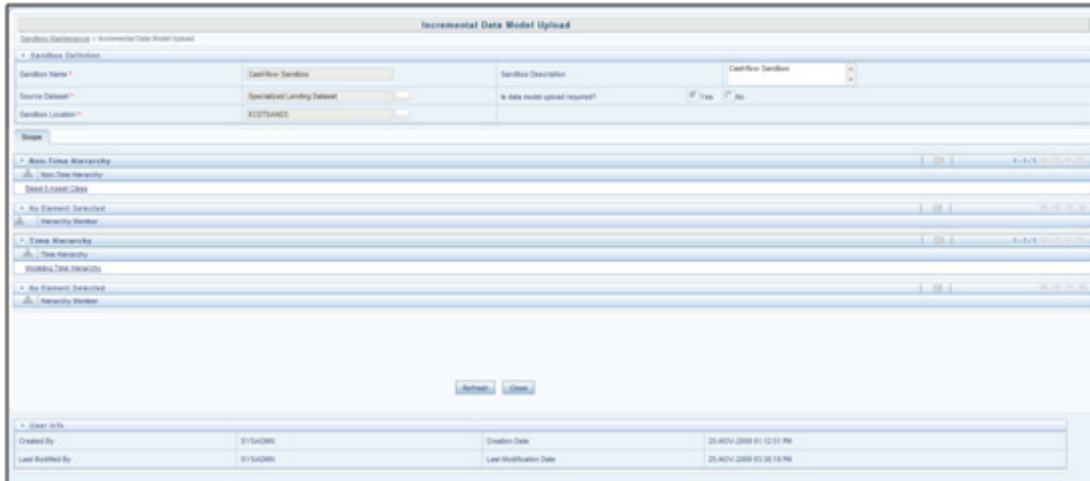
Figure 12. Incremental Data Model Upload

3. Click **Refresh**. The incremental data model changes are fetched and updated.

4. An information dialog is displayed confirming successful incremental data model upload. Click **OK**.

5. Once the Incremental changes are updated to the Data Model in the Sandbox Information Domain, you need to re-save the details in the Sandbox definition window. For more information, see the *Modifying Sandbox Definition* section.

Searching Sandbox Maintenance

The Sandbox Maintenance page allows you to search the sandboxes that you want to view.

To search for Sandbox Maintenance, follow these steps:

1. Navigate to the Sandbox Maintenance page.

2. Enter the name of the sandbox you want to search.

3. Click . The filtered list is displayed.

CHAPTER 5 — *Managing Techniques*

This chapter provides information about Techniques in the Oracle Financial Services Enterprise Modeling application and step-by-step instructions to use this section.

This section includes the following topics:

- About Techniques
- Managing Techniques

About Techniques

The Technique Registration module in Enterprise Modeling displays the techniques developed using R script and ORE functions, prepackaged techniques, and external library techniques. An external library technique is based on third party algorithms, which can be a library or executable.

Technique Registration Workflow
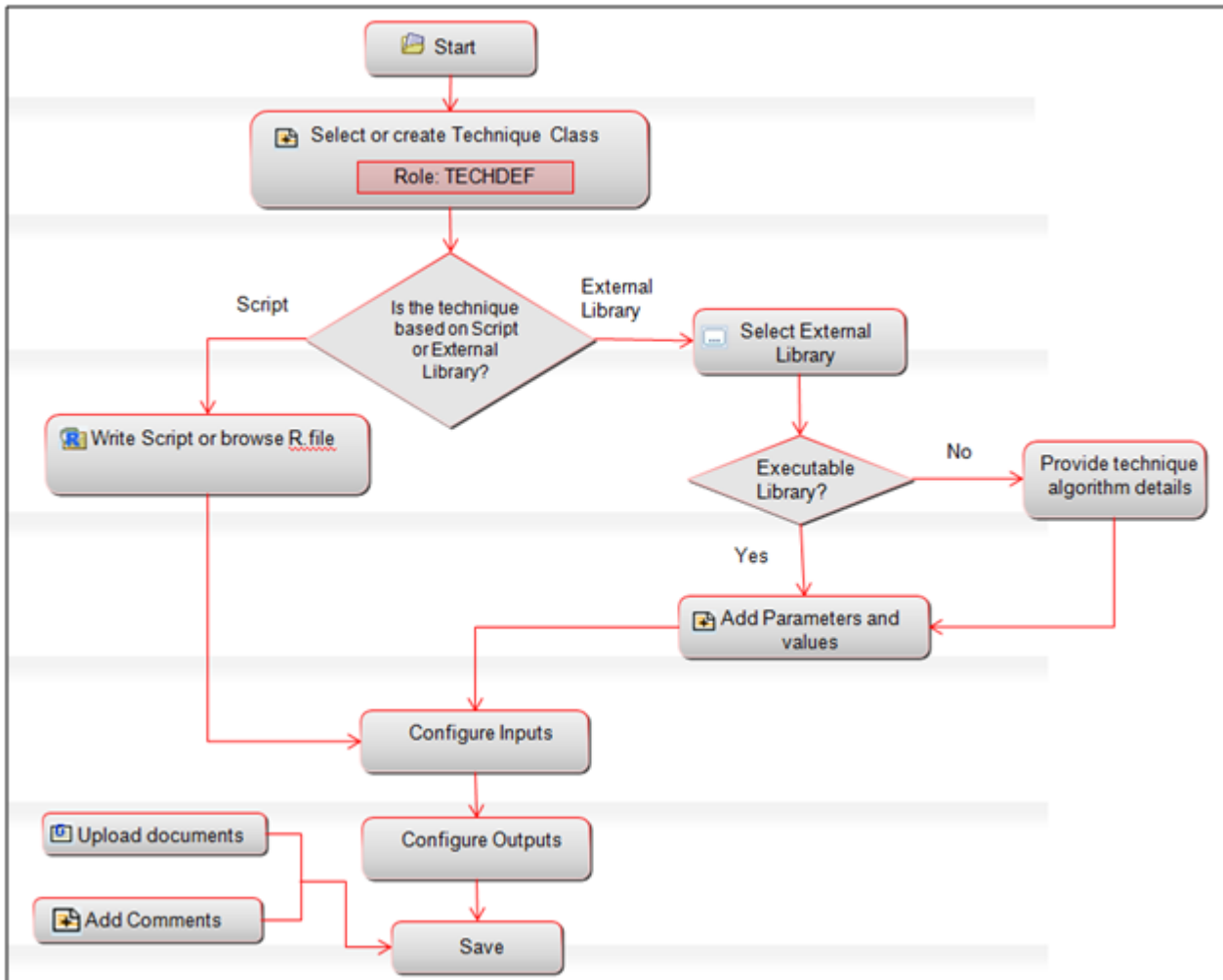
Figure 13. Technique Registration Workflow

User Roles, Privileges, and Actions

The following user roles are required to manage techniques:

- TECHDEF
- TECHAUTH

Managing Techniques

This section explains how to manage technique feature in Oracle Financial Services Enterprise Modeling application.

The following topics are included in this chapter:

- Accessing Techniques
- Creating a Technique Based on R Script
- Creating a Technique using External Library
- Executing a Technique
- Authorizing a Technique
- Viewing a Technique
- Modifying a Technique
- Copying a Technique
- Deleting a Technique
- Searching Techniques

Accessing Techniques

You can access the Technique Registration page from the Oracle Financial Services Enterprise Modeling Home page.

To access the Technique Registration page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click **Technique Registration**. The Technique Registration page is displayed.

	Technique ID	Technique Name ▲	Status	Technique Class	Created By	Creation Date	Based On
	AAIRP_CL_KMEA	R_k-means	Authorized	ⓘ Multivariate Methods	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_VCCM_KRC	R_Kendall Rank Correlation	Authorized	ⓘ Variance Covariance and Correlation Matrix	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_RT_LR	R_Linear Regression	Authorized	ⓘ Regression Techniques	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_RE_LORE	R_Logistic Regression	Authorized	ⓘ Regression Techniques	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_VCCM_PCC	R_Pearson Correlation Coefficient	Authorized	ⓘ Variance Covariance and Correlation Matrix	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_VCCM_SRC	R_Spearman Rank Correlation	Authorized	ⓘ Variance Covariance and Correlation Matrix	SYSADMN	19-May-2015 13:08:35	Script Based
	AAIRP_RT_SR	R_Stepwise Regression	Authorized	ⓘ Regression Techniques	SYSADMN	19-May-2015 13:08:35	Script Based
	1433750489364	R_testR	Authorized	ⓘ Regression Techniques	null	08-Jun-2015 13:31:29	Script Based
	1434362732758	testTpt	Authorized	ⓘ T1	null	15-Jun-2015 15:35:32	External Library Based

Figure 14. Technique Registration

Note: The View and Copy buttons are enabled only for pre-packaged R techniques.
To see the list of prepackaged ORE techniques, refer to the List of Prepackaged ORE Techniques section.

Creating a Technique Based on R Script

Oracle Financial Services Enterprise Modeling application enables you to create techniques.

To create a technique, follow these steps:

1. Click **Technique Registration**. The Technique Registration window is displayed.

2. Click [icon] from the Technique Registration toolbar. The Technique Definition window is displayed.

Figure 15. Technique Definition

3. Enter the details described in the following table:

Table 5. R script based Technique Definition - Field Description

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Technique Name	Enter a name for the technique. **Note:** Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces. **Note:** The prefix R_ is added for Pre-Packaged Techniques based on R/ORE script.

Table 5. R script based Technique Definition - Field Description

Field	Description
Technique Class	Technique Class is used for logically grouping the techniques. 1. Click [⋯] and open the Technique Class Selection window. 2. Right-click **Technique Classes** or the technique class name under which you want to create a technique class, and click **Add Technique Class**. Enter the technique class name in the Add Node window. 3. Expand the Technique Classes tree, select the required technique class and click [⤷]. 4. Remove a selected technique class by selecting the technique class and clicking [⤶].
Technique Description	Enter a brief description about the technique.
Based On	Select the **Script** option.
Is ORE implementation used?	Select the checkbox if you want to use ORE objects within the R script. It is recommended to use Standard-R entirely or ORE variant entirely and not to mix the two.
Language	Select the language which is used for scripting. The options include: ● R ● ORE

4. In the Script Console pane, enter the R script for the technique. For more information on how to use fitting and prediction, refer to the section.


```
» Script Console
## ------ DECLARATION------
## Variable -->
## Single Value Parameter -->
## Model Input -->
## Output -->
## Model Output -->
## -----------------------------
```
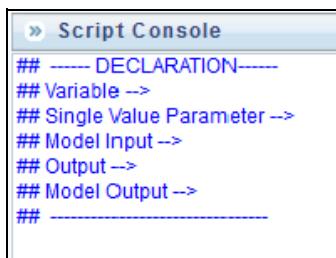
Figure 16. Script Console

5. By default, the Script Console is displayed as shown above. Enter the following details:
   ● **## Variable -->** Enter the variables you want to use in the R script separated by commas. You should map these variables to the corresponding variables which are available in the Variable Management window, during defining the model.
   ● **## Single Value Parameter -->** Enter the input parameters you want to use in the R script.
   ● ## **Model Input-->** Enter the object names you want to map the fitted output.
   ● **## Output -->** Enter the output parameters you want to use in the R script.
   ● ## **Model Output-->** Enter the object names you want to persist.

Note: The variables and the input parameters are case sensitive. The same name cannot be used for both variable and input parameters.

6. Click ▼ from the Script Console toolbar. The Inputs window is displayed.

Inputs			
⌄ Inputs			🗑
Input Name	Type	Display Name	Value
☐ x	Variable ▾	Mean	
☐ y	Variable ▾	Median	
☐ a	Single Value Parameter ▾	Pie	3.141
☐ b	Single Value Parameter ▾	e	2.71828

OK   Cancel
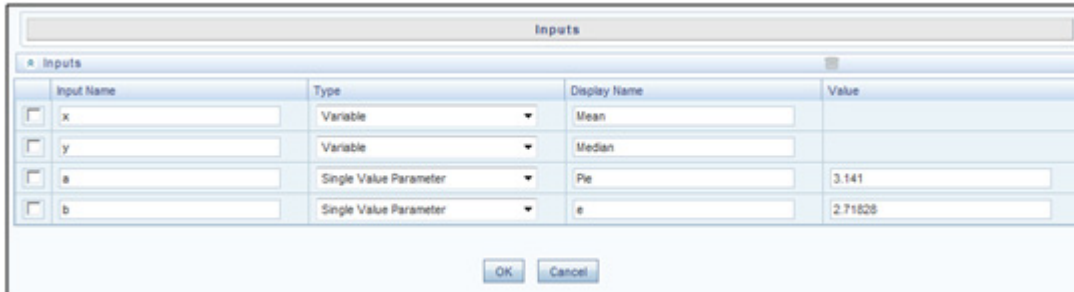
Figure 17. Inputs window

- Enter **Display Name** for the variables you defined. The mapping of variables is done during model definition.

- Enter **Display Name** and **Value** for single value parameters you defined.

- Select the parameter and click 🗑 to delete it.

7. Click ⬆ from the Script Console toolbar. The Outputs window is displayed.

Outputs	
⌄ Outputs	🗑
Output Name	Output Display Name
☐ z	

OK   Cancel
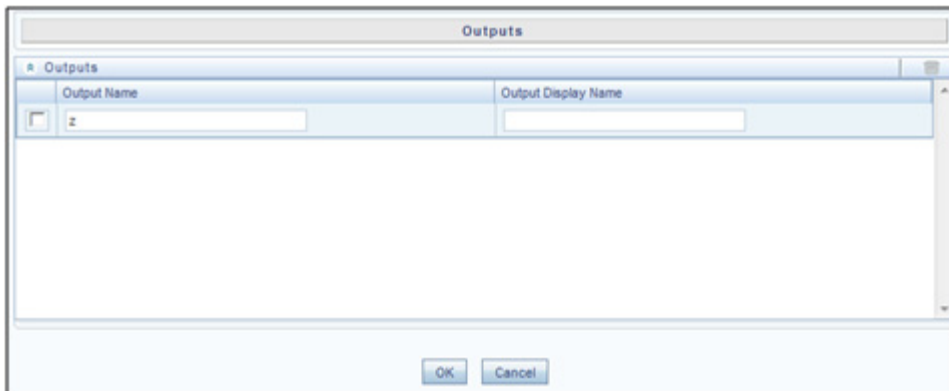
Figure 18. Outputs window

- Enter **Display Name** for the output parameter you defined.

8. Click 🄡 to browse and upload **.R** file. You can use this option if you have already defined a technique and saved it in an **R** file. Once uploaded successfully, the script is displayed in the script console pane.

9. Click 🅖 from the Upload Supporting Documents toolbar.

- Click **Browse** and select the file you want to upload.

- Click ➕ to add additional row to upload more files.

- Click ✕ to remove selected/ uploaded file.
- Click **Upload Files**.

10. Click the **Comments** tab and click 🖺 to add additional information, if required.

11. Click **Save**. If you have authorization privilege, then the technique is auto authorized and it can be used in creating models.

R Script Techniques

You can create a technique using R functions and ORE functions and register it and use in model definitions. If you already have defined a technique using R script and saved it in a .R file, you can upload it and register it as a technique.

The Declaration Block in the script is used to declare the input and output parameters. The format of declaration block and the content should not be altered. The scripting should start only from the line after the declaration block.

There are two types of input parameters which can be used in the R script, namely variable, and single value parameter. The input parameters can be defined from the Declaration block in the Script Console pane, or by right-clicking the parameter and selecting the type as variable or single value parameter. The value of single value parameter is defined from the Inputs window. However, the value of variable is mapped during the model definition in which the technique is used.

You can configure the numeric output by declaring it in the Declaration block (## Output -->), or by right-clicking the parameter and selecting the type as **Output**. It is not required to declare output parameters for plots or graphical outputs.

The script should be a valid R script. You can use ORE functions instead of R functions for better performance. All the object names or functions used in the script should be syntactically valid names. Validation of the script is not done while registering the technique.

Creating a Technique using External Library

This option facilitates you to create a technique based on Third Party algorithms. The algorithms can be a library or executables.

To create a technique based on external library, follow these steps:

1. From the Technique Registration window, click 🖺 . The Technique Definition window is displayed.
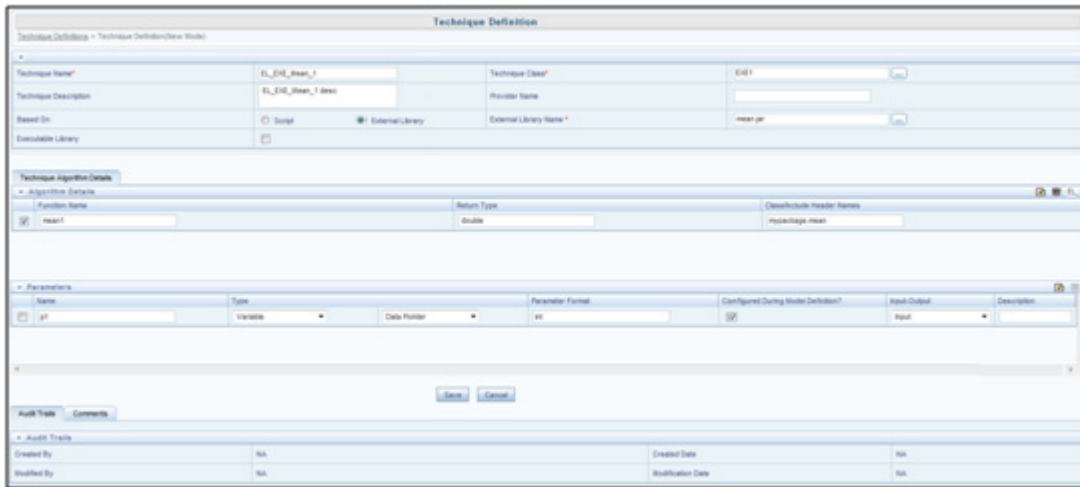
Figure 19. Technique Definition

2. Enter the details described in the following table:

Table 6. External Library based Technique Definition - Field Description

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Technique Name	Enter a name for the technique. Ensure that there are no special characters like `, {,},",', ~, <,>, /, \, and multiple spaces.
Technique Class	Technique Class is used for logically grouping the techniques. 1. Click [...] and open the Technique Class Selection window. 2. Right-click **Technique Classes** or the technique class name under which you want to create a technique class, and click **Add Technique Class**. Enter the technique class name in the Add Node window. 3. Expand the Technique Classes tree, select the required technique class and click . 4. Remove a selected technique class by selecting the technique class and clicking .
Technique Description	Enter a brief description about the technique.
Provider Name	Enter the name of the provider of the external library.
Based On	Select the **External Library** option.
External Library Name	1. Select the external library dependent file for the third party technique by clicking [...] . The External Library Selection window is displayed. 2. Select the dependent file from the External Library pane and click . 3. Add a new dependent file name. 4. Right-click **External Library** and click **Add External Library**. 5. Enter a Label for the new node and click **OK**. Ensure that there are no special characters like `, {,},",', ~, <,>, /, \, and multiple spaces.

Table 6. External Library based Technique Definition - Field Description

Field	Description
Executable Library	Select the checkbox if it is a self executable library. Self executable libraries are executables(C/C++ or JAVA or Python script) which have a main function defined and need to be invoked directly by passing the run time parameters. Currently only those executables which have interactive or new line arguments are supported. For techniques based on executable library, only parameters can be provided.
Algorithm Details This grid will not be displayed if you have selected the Executable Library checkbox. Click ![] to add more functions. Select a function and click ![] to delete it or click ![] to view the function declaration.	
Function Name	Enter the name of the function.
Return Type	Enter the return type of the function.
Class/ Include Header Names	Enter the class name or header names to be included for the function.
Parameters (It is not mandatory to pass parameters) Click ![] to add more parameters. Select a parameter and click ![] to delete it.	
Name	Enter the name of the parameter.
Type	Select the parameter type from the drop-down list. The options are: • **Variable:** Select this option if the input data to a function call has to be passed as a Variable. If the function call accepts array, select Data Pointer. If the call accepts input data to be stored in a file, select Data File. Based on the selection the framework passes the data as an array or stores it in a file and then pass the path of that file. • **Single Value Parameter:** Select this option to define single value parameters. You can enter multiple single value parameters separated by commas. • **Class Object:** Select this option to pass another function or class object to be passed as an argument/ parameter. The class objects need to be initialized before passing as an argument. Enter the initialization value that needs to be passed in the adjacent text box provided. Currently Class Object type is only supported for C++ libraries. • **Multi Value Parameter:** Select this option to pass an array of values or matrix input to the function. If the function call accepts array, select Data Pointer. If the call accepts input data to be stored in a file, select Data File. Only .csv files are supported. You need to upload the file while defining the model. • **Number of Observations:** Select this option for function arguments which need input data size. The number of records used in processing will be passed to this argument. • **Number of Variables in:** Select this option to pass the number of input rows used for computing coefficient for functions like linear regression, which have multiple independent variables. Enter the variable parameter name whose size is assigned to this argument in the adjacent text box provided. • **Plot path**: Select this option and enter the pathof the plot.
Parameter Format	Enter the format of the selected parameter type.

Table 6. External Library based Technique Definition - Field Description

Field	Description
Configured During Model Definition?	Select the checkbox if you want to configure the parameters during model definition. In case of Multi Value Parameter whose data is stored in a Data File, the checkbox should be selected.
Input- Output	Select the parameter type from the following options: ● Input ● Output ● Input- Output
Description	Enter a brief description of the parameter.

3. Click **Save**.

Note:

● In the Model Definition window, the parameters for which the **Configured During Model Definition?** checkbox is selected will only be displayed.

● Currently only static functions are supported for Java based external algorithms.

● The function return type should be primitive data types and be single value, that is, pointers or arrays are not supported.

Executing a Technique

After creating techniques based on non-self executable libraries, you must ensure that the techniques are compiled by executing a command line utility called TechniqueBuilder.sh, prior to executing the models. This utility is available under $FIC_DB_HOME/bin folder. For more information, refer to the *Command Line Utility for Compiling Non-Self Executable Libraries* section in the *Oracle Financial Services Analytical Application Infrastructure User Guide*.

The self executable libraries should be placed under $FIC_DB_HOME/bin/MF_EXTERNAL_EXECUTABLES folder. For non-self executable libraries, the libraries should be placed under $FIC_DB_HOME/lib/MF_EXTERNAL_LIBS/lib folder and the header files should be under $FIC_DB_HOME/lib/MF_EXTERNAL_LIBS/include folder.

Authorizing a Technique

An authorizer has to authorize a technique before it can be used for model definition. Users who have the function TECHAUTH mapped to their user role can authorize the technique. If the user who created the technique has authorization privilege, then the technique is auto authorized. An authorizer can reject a technique if required.

To authorize or reject a technique, follow these steps:

1. Click **Technique Registration** from the LHS menu. The Technique Registration window is displayed.

2. Select the technique and click . The Authorize Technique window is displayed.

3. Enter a brief description in the **Comments** field. This is mandatory for rejecting the technique.

4. Click **Approve** or **Reject**.

Viewing a Technique

This option allows you to view the details of user defined as well as prepackaged techniques like technique name, class, description and the script.

To view the technique details, follow these steps:

1. Click **Technique Registration** from the LHS menu. The Technique Registration window is displayed.

2. Select the checkbox adjacent to the Technique ID and click in the Technique Registration tool bar. The Technique View window is displayed with all the technique details.

Modifying a Technique

You can update the details of a user defined technique. If the technique is already mapped to a model or if it is a prepackaged technique, you cannot modify it. An alert message is displayed. You cannot modify the Technique Name and Technique Class. For techniques based on external library, you can modify External Library Name, function, and parameter details.

After modification, the technique should be authorized. If you have authorization rights, it will be auto authorized; else an authorizer has to approve it.

To modify a technique, follow these steps:

1. Click **Technique Registration** from the LHS menu. The Technique Registration window is displayed. The Technique Registration window is displayed.

2. Select the technique you want to modify and click . This button is disabled if you have selected multiple techniques.

3. Modify the technique details. For more information, refer to section *Creating a Technique using External Library*.

Copying a Technique

You can create a copy of a user defined as well as prepackaged technique with the required variants. The copied technique should be authorized. An authorizer has to approve or reject the copying of the technique. If you have authorization rights, it will be auto authorized.

To copy a technique, follow these steps:

1. Click **Technique Registration** from LHS menu. The Technique Registration window is displayed. The Technique Registration window is displayed.

2. Select the technique you want to copy and click . This button is disabled if you have selected multiple techniques.

3. In the Technique Definition Copy window, you can:

- Create a new technique with the existing details. Specify a new Technique Name and click **Save**.

- Create a new technique by updating only the required details. Specify a new Technique Name and update the required details.

For more information, refer to section *Creating a Technique using External Library*.

Deleting a Technique

You can delete a user defined technique which is no longer required. However, if the technique is already mapped to a model or if it is a pre-packaged technique, you cannot delete it.

An authorizer has to approve or reject the deletion of the technique. If you have authorization rights, it will be auto authorized.

To delete a technique, follow these steps:

1. From the LHS menu, click **Oracle Financial Services Enterprise Modeling** and **Technique Registration**. The Technique Registration window is displayed.

2. Select the technique you want to delete and click 🗑 .

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

Searching Techniques

The Techniques Registration page allows you to search for the techniques that you want to view.

To search for Techniques, follow these steps:

1. Navigate to the Technique Registration page.

2. Enter the Technique Name and Technique Class you want to search.

3. Click 🔍 .The filtered list is displayed.

Managing Variables

This chapter provides information about variables in the Oracle Financial Services Enterprise Modeling application and step-by-step instructions to use this section.

This chapter includes the following topics:

- About Variable Definition
- Managing Variables

About Variable Definition

Variable refers to a logical set of attributes that are likely to change based on the selected parameters. In a modeling environment, variables play a vital role in filtering the model parameters and deriving an estimate based on historical data. Variables are defined in the production information domain.

Managing Variables

This section explains how to manage variables in the Oracle Financial Services Enterprise Modeling application. The following topics are included in this section:
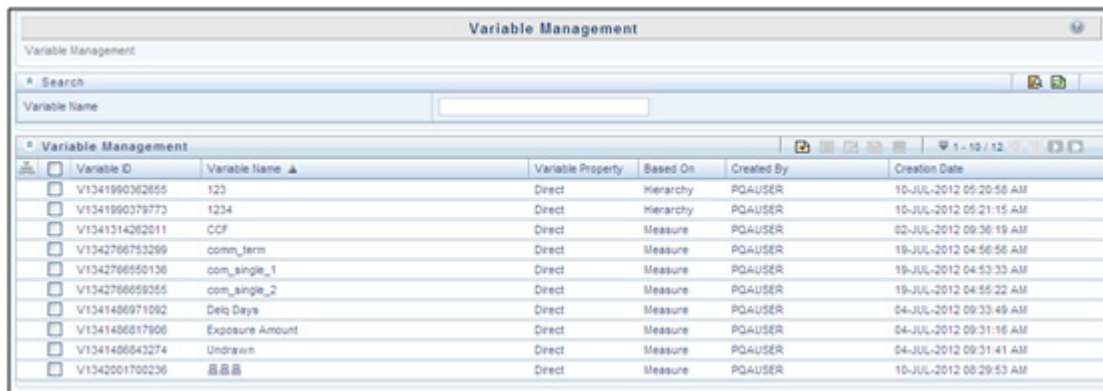
- Accessing Variable Management
- Adding Variable Definition
- Viewing Variable Definition
- Modifying Variable Definition
- Copying Variable Definition
- Deleting Variable Definition
- Searching Variables

Accessing Variable Management

You can access the Variable Management page from the Oracle Financial Services Enterprise Modeling Home page. To access the Variable Management page, follow these steps:

1. Login to Oracle Financial Services Enterprise Modeling as a Business User. The Oracle Financial Services Enterprise Modeling Home page is displayed.

2. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

3. Click ▶ to expand the Data Management section.

4. Click ▶ to expand the Datasets and Variables.

5. Click **Variable Definition**. The Variable Management page is displayed.

Figure 20. Variable Management

Adding Variable Definition

The Add Variable Definition page enables you to add a variable by filtering through the required parameters. The options available to filter the data vary depending on the variable type selected.

Note: The variables created in the Variable Definition page are IDIOSYNCRATIC NUMERIC variables.

To add variable definition, follow these steps:

1. Select 🔲 from the Variable Management tool bar. The Add button is disabled if you have selected any checkbox in the grid. The Variable Definition Add window is displayed.

Figure 21. Variable Definition

2. Enter the Variable details as described in the following table:

Table 7. Variable Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Based On	Select any of the following options from the drop-down list, based on which the variable definitions are to be created. ● Hierarchy ● Measure ● Business Processor ● Entity
Folder	Select the folder in which you want to create the variable.
Variable Name	Enter a name for the variable. Ensure that there are no special characters like \`, {,},", ', ~, <,>, /, \\, and multiple spaces.

Table 7. Variable Definition - Field Descriptions

Field	Description
Variable Description	Enter the required description for the variable.
Variable Transformation	This feature is available for variables based on Measures, Business Processor, and Entity. 1. Click to define a transformation for the Variable. The Technique Browser for Transformation window is displayed. 2. Select the transformation you want to apply to the variable and click . The options are Arithmetic, Exponential, Logarithm, Power, and Trigonometric. 3. Click or to change the order in which the transformation has to be applied. Click **OK**. - If Arithmetic is selected, select the Operator (Addition, Subtraction, Multiplication, and Division) from the drop-down list and enter the Operator Value. - If Logarithm is selected, select the Base from the drop-down list. The options are 10, 2, and e. - If Power is selected, enter the value of Power. - If Trigonometric is selected, select the Function from the drop-down list. The options are Sine, Cosine, and Tangent. - If Exponential is selected, the exponential value of the variable will considered for transformation of the variable.

4. If the variable is based on hierarchy, follow these steps:

- Select **Variable Property** as Direct (variable in Binary form and used mainly in regression analysis).

- Select the required Hierarchy from the drop-down.

   Note: Hierarchy-based variables cannot be created on multi-level Business Intelligence hierarchies.

5. If the variable is based on Measures, follow these steps:

- Select **Yes** adjacent to Apply Filters.

- Select the Datasets from the drop-down.

   You can click to view the details of the selected datasets.

- Select the Measures from the drop-down.

   You can click to add a new Business Measure. For more details on creating a Business Measure, refer to *Creating Business Measure* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

- Click in the Filters grid and the Hierarchy browser is opened.

- Select the hierarchy member(s) and click **OK**. The selected members are displayed in the Filters grid under Hierarchies.

- Double-click a hierarchy member to invoke a grid with the member name.

- Click in the hierarchy member grid and the Hierarchy Browser is displayed.

- Select the node(s) and click **OK**. The selected nodes are displayed in the Hierarchy member grid under Nodes.

- Click and specify the **Selected Measures** by selecting from the Measure Browser.

6. If the variable is based on Business Processor, follow these steps:

   - Select the Business Process from the drop-down.

     You can click to add a new Business Processor. For more details on careting a Business Processor, refer to *Adding a Business Processor* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

     Note: Variables cannot be created based on Business Processor having placeholders.

7. If the variable is based on Entity, follow these steps:

   - Click Entity to open the Entity and Attribute window. The available tables are displayed under the Available Entities section.

   - Select the checkbox corresponding to the table you want to select. The available columns in the selected table are displayed under the Available Attributes section.

   - Select the required attribute for which you want to create a varaible. You can select multiple attributes.

   - Click **Save**. The selected attributes are displayed in the Selected Attributes section.

8. Click **Save** to upload the new Variable definition details. The User Info grid at the bottom of Variable Definition Add window displays metadata information about the Variable definition created.

Viewing Variable Definition

You can view individual Variable Definition details at any given point.

To view the existing Variable Definition details, follow these steps:

1. Select the checkbox adjacent to the Variable ID.

2. Click in the Variable Management tool bar. The Variable Definition View window is displayed with all the variable details.

Modifying Variable Definition

You can modify only the Variable Description details if the variable is mapped to a definition. If the variable is not mapped to any definition, then all the filtering parameters can be modified.

To modify an existing Variable Definition, follow these steps:

1. Select the checkbox adjacent to the Variable ID whose details are to be updated.

2. Click in the Variable Management tool bar. The Edit button is disabled if you have selected multiple Variable IDs. The Variable Definition Edit window is displayed.

3. Edit the Variable Definition details as required. For more information, refer to section *Adding Variable Definition*.

4. Click **Save** to upload changes.

Copying Variable Definition

The Copy Variable Definition page enables you to quickly create a new Variable based on the existing variable parameters or by updating the required parameters.

To copy an existing Variable Definition, follow these steps:

1. Select the checkbox adjacent to the Variable ID whose details are to be duplicated.

2. Click in the Variable Management tool bar. The Copy button is disabled if you have selected multiple Variable ID(s).

3. In the Variable Definition Copy window, you can:

- Create new variable with the existing details. Specify a new Variable Name and click **Save**.

- Create new variable by updating only the required details. Specify a new Variable Name and update the required details. For more information, refer to section *Adding Variable Definition*. Click **Save**. The new variable definition details are displayed in the Variable Management window.

Deleting Variable Definition

You can remove the variable definition(s) which are no longer required in the system, by deleting from the Variable Management window. You cannot delete a variable which is mapped to any definition.

To delete a Variable Definition, follow these steps:

1. Select the checkbox adjacent to a Variable ID.

2. Click from the Variable Management tool bar. A confirmation dialog is displayed.

3. Click **OK** to confirm deletion.

Searching Variables

The Variables Management page allows you to search for the techniques that you want to view.

To search for Variables, follow these steps:

1. Navigate to the Variables Management page.

2. Enter the name of the Variable Name you want to search.

3. Click .The filtered list is displayed.

CHAPTER 7   *Managing Modeling*

This chapter provides information about Modeling in the Oracle Financial Services Enterprise Modeling application and step-by-step instructions to use this section.

This chapter includes the following topics:

- About Modeling
- User Roles and Actions
- Managing Modeling
- Model Execution Status
- Model Outputs
- Model Deployment
- Model Execution

About Modeling

Models are built based on various techniques and related parameters based on the business purpose. Models thus built use various metadata objects such as Measures, Hierarchies, Filters, Datasets, and so on. The techniques can be inbuilt techniques, user defined R/ ORE based techniques, or techniques based on External Library.

Modeling refers to the process of designing a prototype based on a structured data model, considering all the variables for statistical analysis and to simulate real events and processes. Modeling module enables you to measure and quantify risk. You can use of pre-defined models to predict the business trends and also to validate the existing models.

Note: You can create models only in Sandbox Information Domain.

Modeling Workflow
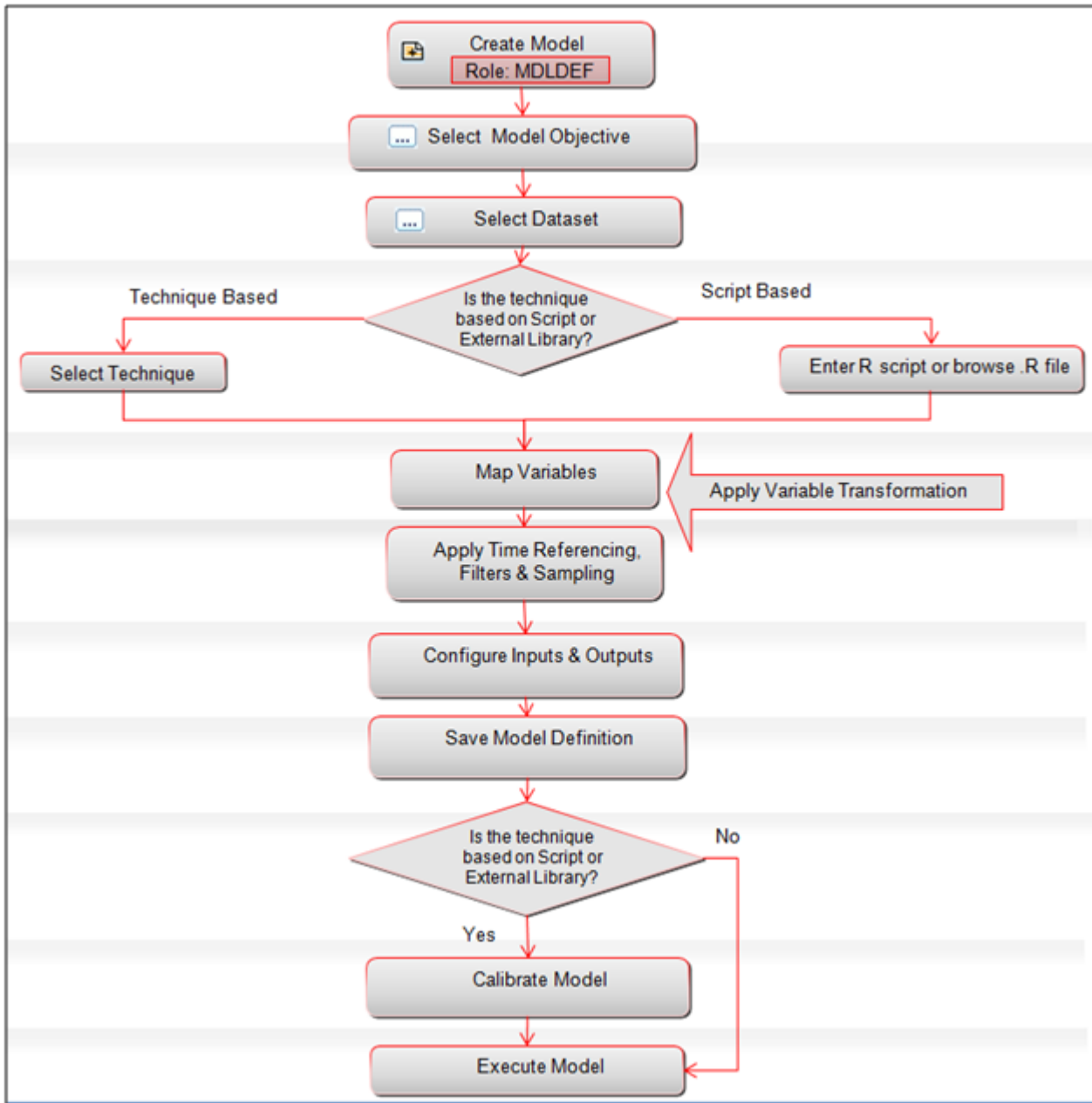
Figure 22. Model Creation Workflow

User Roles and Actions

User Roles required for creating models are as follows:

- MDLDEF
- MDLAUTH
- MDLEXEC
- MDLOUTPUT
- MDLDEPLOY
- MDLCALIB
- MDLCHAMP

Managing Modeling

This section explains how to manage modeling feature in *Oracle Financial Services Enterprise Modeling* application. The following topics are included in this section:
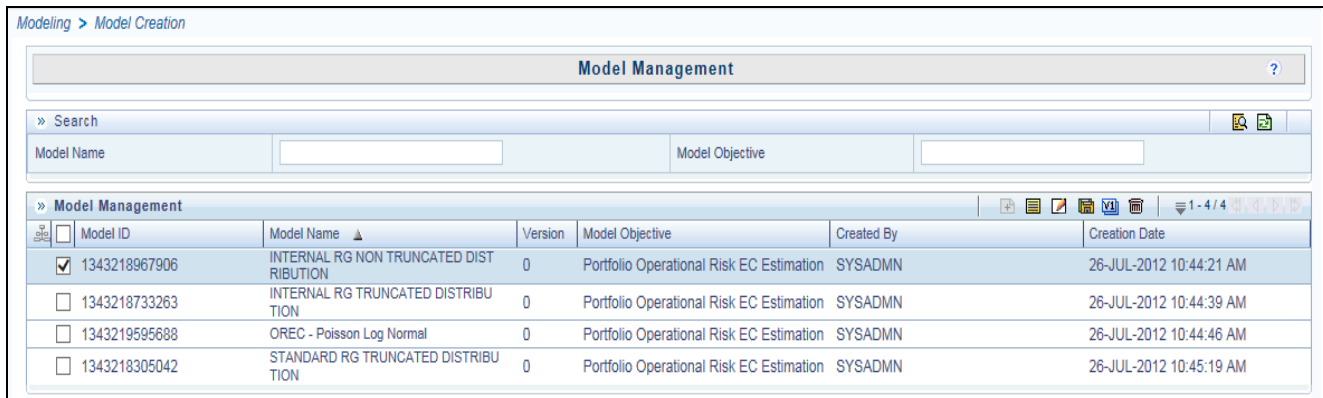
- Accessing Model Management
- Model Versioning
- Creating and Executing a Model
- Model Parameters
- Creating Models using Logical Sandbox
- Viewing Model Definition
- Modifying Model Definition
- Deleting Model Definition
- Searching Model Management

Accessing Model Management

You can access the Model Management page from the Oracle Financial Services Enterprise Modeling Home page. To access the Model Management page, follow these steps:

1. Click **Sandbox** tab.

2. Click **Model Creation**. The Model Management page is displayed.

Figure 23. Model Management

The following columns are displayed in the Model Management page.

- Model ID

- Model Name

- Version

- Model Objective

- Created By

- Creation Date

Note: The latest version of the model definition is displayed, when you select a model. You can click ☐ to view the Model Definitions and click ☐ to edit the Model Definition.

Model Versioning

The Model Definitons created in *Oracle Financial Services Enterprise Modeling* are versioned. The version of the Model Definiton is displayed in the Model Management summary screen. You can ☐ to view different versions of the selected Model Definition in the Model Management screen.

When a Model Definiton is created, it is saved as version 0. During subsequent modifications, the Model Definition is saved as different versions. For example: The version of a Model Definition is 0, after it is created. The version is updated to 1, after the Model Definition is edited. After editing a Model Definiton, you can choose save a Model Definition as a new version or overwrite the existing version.

Creating and Executing a Model

The Modeling module enables you to create and execute data models.

You can also create Variables from the Model Definition window. This option is available only for models created using External Library Based Technique, R scripted Technique, and R Script. This option is not available for NAG Techniques.

The following techniques can be used to create a data model.

- Creating a Model Using NAG Technique

- Creating a Model Using External Library Based Technique

- Creating a Model Using R Scripted Technique

- Creating a Model using R Script (R and ORE Functions)

Creating a Model Using NAG Technique

This section describes the procedure for creating a model using NAG Technique. For information on inbuilt NAG techniques, refer List of NAG Techniques. Oracle Financial Services Enterprise Modeling supports both time homogeneous and non-homogeneous estimates of transition probabilities. For more information on defining such a statistical technique, refer Transition Matrix.

To create a model using inbuilt technique, follow these steps:

1. Click **Sandbox** tab.

2. Click **Model Creation**. The Model Management page is displayed.

3. Click [icon] from the Model Management toolbar. The Model Definition window is displayed. The Add button is disabled if you have selected any Model ID in the grid.

Figure 24. Model Definition

4. Enter the model definition details as described in the following table:

Table 8. Model Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Model Name	Specify a model name for the model definition. Model Name is case sensitive and does not allow duplication. For example, model name "Linear Regression" is not allowed if a model with the name "linear regression" exists. Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces.
Model Description	Enter a description for the model.

Table 8. Model Definition - Field Descriptions

Field	Description
Do you like to script the model?	Select the checkbox to script the model in Model Script pane.
Model Objective	Select the Model Objective from the drop-down. You can also click [icon] to create a Model Objective.
Technique	This field is disabled if you have selected to script the model. 1. Click [icon] and open the Technique Selection window. "The pre-packaged techniques, user defined (registered and authorized) R techniques and External Library techniques are listed in the Techniques pane. 2. Click [icon] and expand the technique heading groups. 3. Select the required technique and click [icon]. For more information on inbuilt techniques, see the *List of NAG Techniques* section. 4. Click **OK**. The selected Technique details are displayed in the Model Definition New window. If you have selected **R** technique, click [icon] to view the script and upload documents.
Dataset	By default, the dataset of the Sandbox is displayed. You can change the dataset if necessary. However, the dataset you want to select should be a sub-set of the dataset of the sandbox. Dataset selection is mandatory: ● For models based on NAG techniques. ● For models based on R scripted techniques if variables are declared in the R script 1. Click [icon] to open the Dataset Selection window. The available datasets are listed in the Datasets pane. For models created in logical sandbox, the Source Datasets selected during Sandbox creation are displayed. 2. Click [icon] to view the details of the selected datasets. For more information, refer *Dataset Metadata* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*. 3. Click [icon] to create a dataset. For more details on creating a dataset, refer *Creating Data Set* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*. 4. Select the required Dataset based on which the model is to be created and click [icon]. 5. Click **OK**. **Note**: The Datasets based on Derived Entities are not supported.

Table 8. Model Definition - Field Descriptions

Field	Description
Language	Select the scripting language from the drop-down. The options include the following; ● R ● ORE **Note:** This filed is not applicable while creating models using NAG techniques and External Library.
Type	Select the type of engine from the drop-down. The options include the following: ● Standard R Engine ● ORE Engine **Note:** This filed is not applicable while creating models using NAG techniques and External Library.
Calibrate Model / Host Existing Model	These options to choose Calibrate or Host Existing Model are displayed if NAG technique is selected. The Calibrate Model option is selected by default. ● Select **Calibrate Model** to calibrate the variable output parameters and process for model execution. ● In **Host Existing** model, the sampling option is not available and model has to be processed through batch execution.

The grid below the Model Details section displays the various parameters available for the selected technique.

Some of the common parameter types for NAG techniques are explained below. To update the required information, click the following links.

● Variable

● Time Referencing (Not applicable for models defined in Logical Sandbox)

● Filter

● Sampling

● Inputs

● Outputs

5. Once you have updated all the necessary details in the available tabs, click **Save** to save the model definition details.

6. Click **Preview Data** to view the data of the selected dataset. It displays the primary keys and the attributes/columns of the tables in the selected dataset.

7. Click **Calibrate** and select the Calibration Date from the Calibration Parameters window.

8. Click **Execute** to execute the model. The Execution Parameters window is displayed.

Figure 25. Execution Parameters window

9. Select the Execution Date from the calendar. The data of the selected date will be used for model execution.

10. Select the Execution Type as **Baseline** or **Stress**.

   If **Stress** is selected, click **Stress Scenario** [icon] to open the Scenario Browser window. Select the scenario and

   click [icon] . The variables and their shocks of the selected scenario are displayed in the Model Variable Shock Mapping grid. For more information on stress testing, see the *Stress Testing* section.

   The Model Execution Status window provides the execution details.

Creating a Model Using External Library Based Technique

You can create models using Third Party algorithm based techniques which you have defined from the Technique Registration window. The algorithms can be a library or executables. Ensure techniques based on non-self executable libraries are compiled using the Technique Builder utility, before executing a model based on such techniques.

To create a model using external library based technique, follow these steps:

1. Select [icon] from the Model Management tool bar. The Model Definition window is displayed. The Add button is disabled if you have selected any Model ID in the grid.

Figure 26. Model Definition Page

2. Enter the Model Definition Details. The common fields are described in Table 8.

3. The grid below the Model Details section displays the various tabs available for the selected technique. To update the required information, refer to the following sections:

   ● Inputs

   ● Time Referencing (Not applicable for models defined in Logical Sandbox)

   ● Filter

4. Click **Save** to save the model definition details, after all the necessary details are updated.

5. Click **Preview Data** to view the data of the selected dataset. It displays the primary keys and the attributes/columns of the tables in the selected dataset. Only those columns which are mapped to the variables in the script are displayed.

6. Click **Execute**.

   ● Select **Baseline** to execute the model normally.

   ● Select **Stress** to execute the model based on a stress scenario. The Execution Parameters window is displayed.
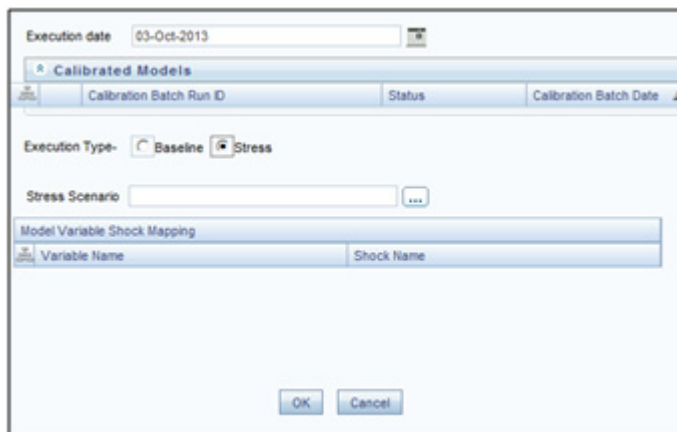

Figure 27. Execution Parameters

   ● Select the Execution Date from the calendar. The data of the selected date will be used for model execution.

   ● Click **Stress Scenario** to open the Scenario Browser window. Select the scenario and click ![icon]. The variables and their shocks of the selected scenario are displayed in the Model Variable Shock Mapping grid. For more information on stress testing, see the *Stress Testing* section.

7. The Execution Status grid displays the model execution log dynamically.

Creating a Model Using R Scripted Technique

You can create models using a prepackaged technique or a technique which you have defined from the Technique Registration window. To know about available prepackaged ORE techniques, refer to *Data usage when ORE native implementation is employed - Oracle Financial Services Enterprise Modeling application - Specifics to be considered.*

Note: Execution Status Log is not displayed in Model Definition window for Models created using Standard R Engine.

To create a model using R scripted technique, follow these steps:

1. Select ![icon] from the Model Management tool bar. The Model Definition page is displayed. The Add button is disabled if you have selected any Model ID in the grid.
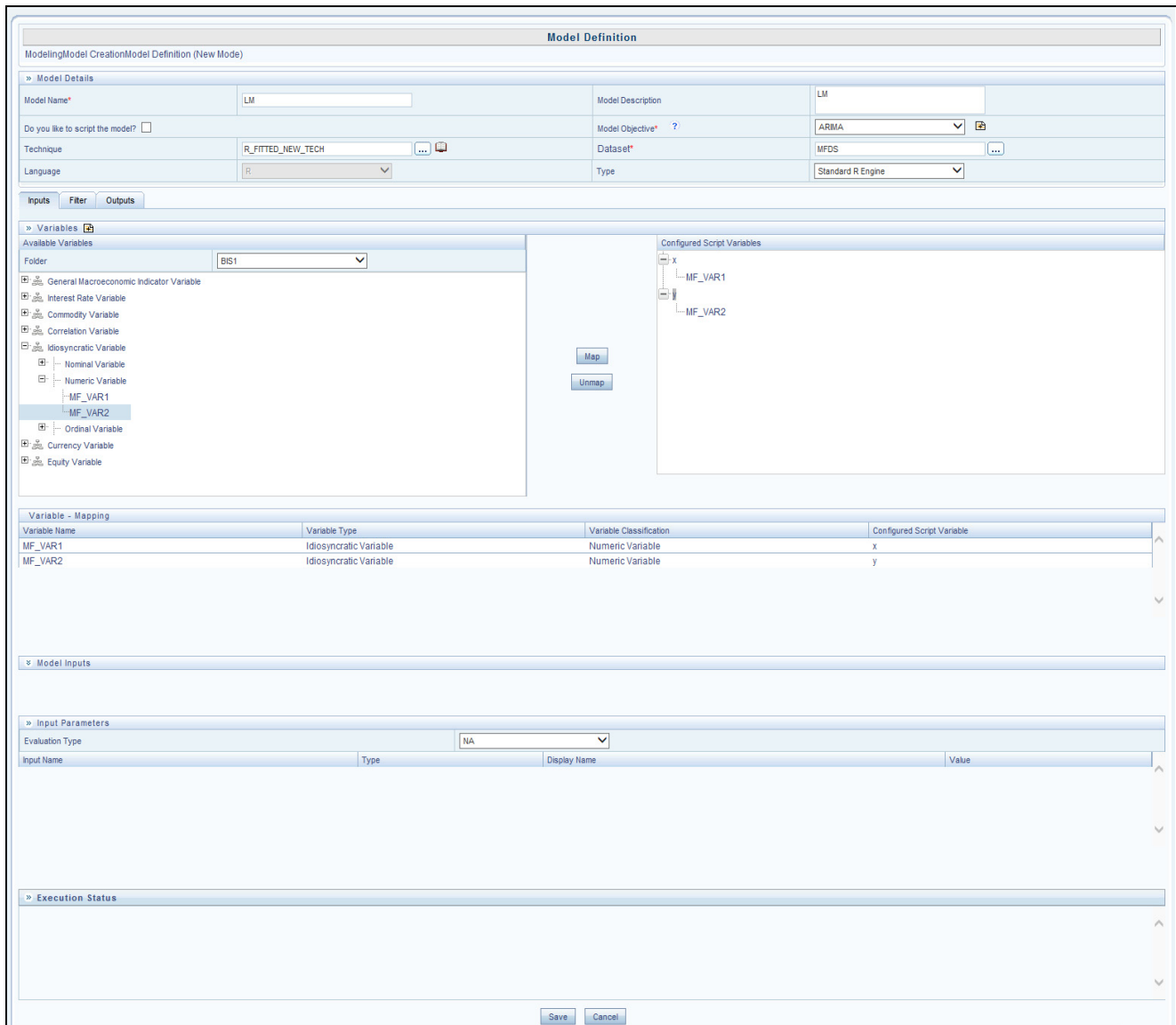
Figure 28. Model Definition page

2. Enter the Model Definition Details. The common fields are described in Table 8.

3. Select the scripting Language from the drop down.

4. Select the Type of engine from the drop-down.

The grid below the Model Details section displays the various tabs available for the selected technique.To update the required information, refer to the following sections:

- Inputs

- Time Referencing (Not applicable for models defined in Logical Sandbox)

- Filter

- Outputs

5. Click **Save** to save the model definition details, after all the necessary details are updated.

6. Click **Preview Data** to view the data of the selected dataset. It displays the primary keys and the attributes/columns of the tables in the selected dataset. Only those columns which are mapped to the variables in the script are displayed.

7. Click **Execute**.

- Select **Baseline** to execute the model normally.

- Select **Stress** to execute the model based on a stress scenario. The Execution Parameters window is displayed.
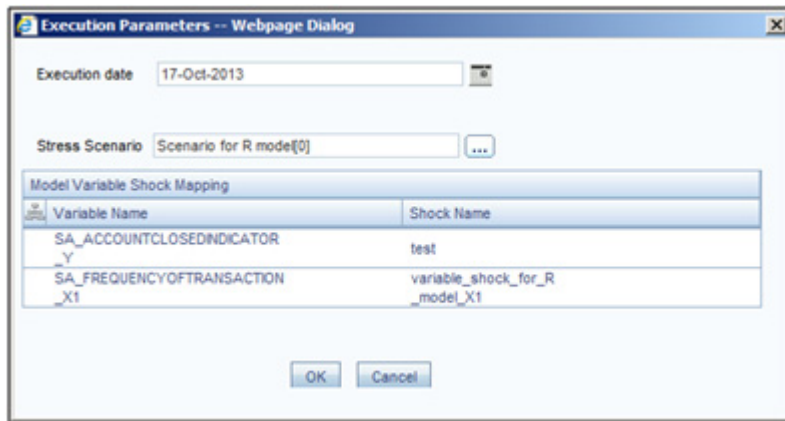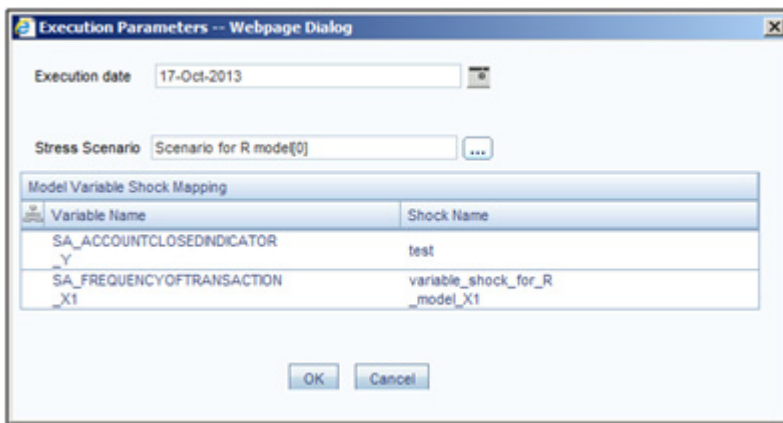

Figure 29. Execution Parameters

- Select the Execution Date from the calendar. The data of the selected date will be used for model execution.

- Click **Stress Scenario** to open the Scenario Browser window. Select the scenario and click [icon]. The variables and their shocks of the selected scenario are displayed in the Model Variable Shock Mapping grid. For more information on stress testing, see the *Stress Testing* section.

  The Execution Status grid displays the model execution log dynamically.

  Note: For R based models, the execution may fail if the dataset contains internal joins.

Creating a Model using R Script (R and ORE Functions)

You can script the model using R script with R functions as well as ORE functions. The script that is used should be a valid R/ ORE script. Moreover, all the object names used in the R-script should be syntactically valid names. You

should select the ORE Implementation checkbox if you are using ORE functions in the script. ORE enables transparent access to data stored in the Database for scalability and high performance. For more information, refer to *Enterprise Modeling support for R Scripts and Oracle R Enterprise (ORE) Statistical Functions.*

Note: The models defined based on R scripting can be directly deployed to the production infodom without execution in sandbox.

To create a model using script, follow these steps:

1. Select from the Model Management tool bar. The Model Definition window is displayed.
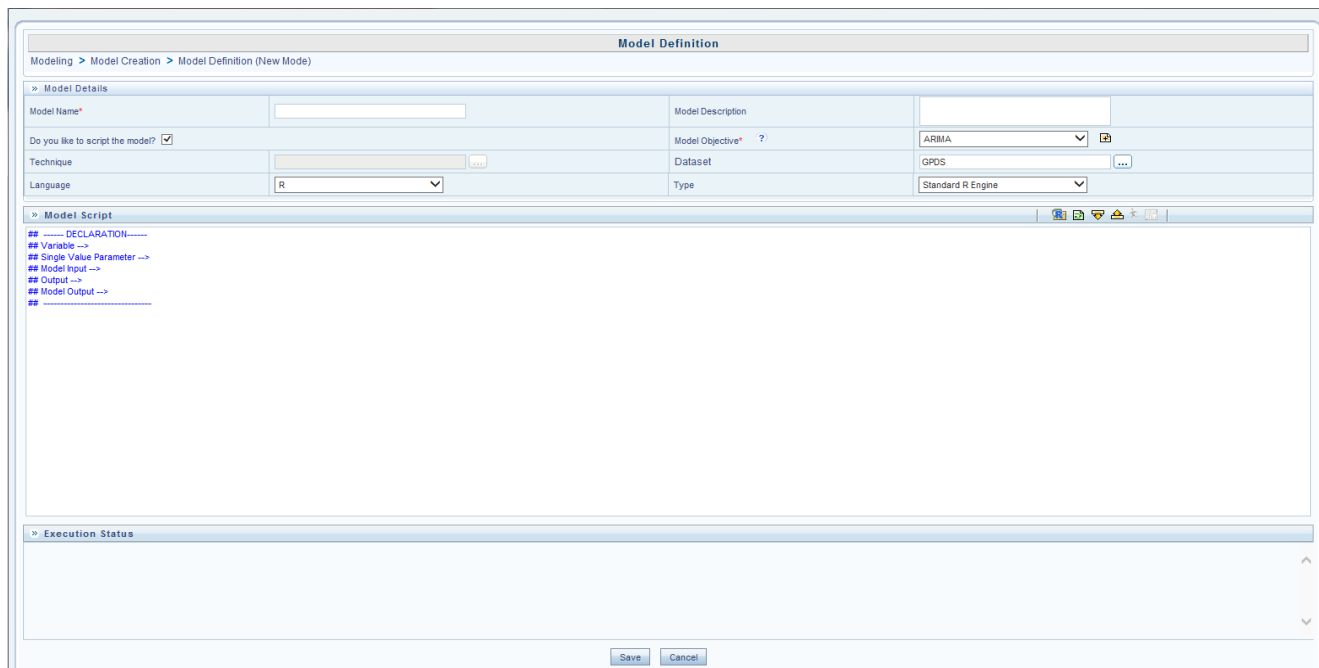
Figure 30. Model Definition

2. Enter the model definition details. The common fields are described in Table 8.

3. Click to browse and upload **.R** file. You can use this option if you have already defined a technique and saved it in an **R** file. Once uploaded successfully, the script is displayed in the Model Script pane.
   Or
   In the Model Script pane, enter the R script for the model. For information on how to use fitting and prediction, see *Fitting and Prediction Using R Script.*
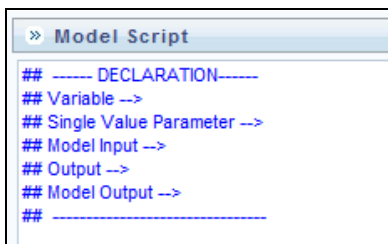
Figure 31. Scripting Console

Note: The format of declaration and the content should not be altered. The scripting should start only from the line after the declaration block.

By default, the Model Script is displayed as shown above.

- ## **Variable -->** Enter the variables you want to use in the R script separated by commas.
- ## **Single Value Parameter -->** Enter the input parameters you want to use in the R script.
- ## **Model Input -->** Enter the object names you want to map the fitted output.
- ## **Output -->** Enter the output parameters you want to use in the R script.
- ## **Model Output-->** Enter the object names you want to persist.

Note: The variables and the input parameters are case sensitive. Same name cannot be used for both variables and input parameters.
You should explicitly define the outputs in the script that need to be updated back to tables. If the output is not defined, then the tables are not updated.
For example :
x = rbeta(100,1,2)
y = rbeta(100,1,2)
op = lm(y~x)

If op$residuals needs to be updated then declare the output with a name.
For Example:
Residuals = op$residuals

4. Click ![icon] from the Script Console toolbar. The Inputs window is displayed.
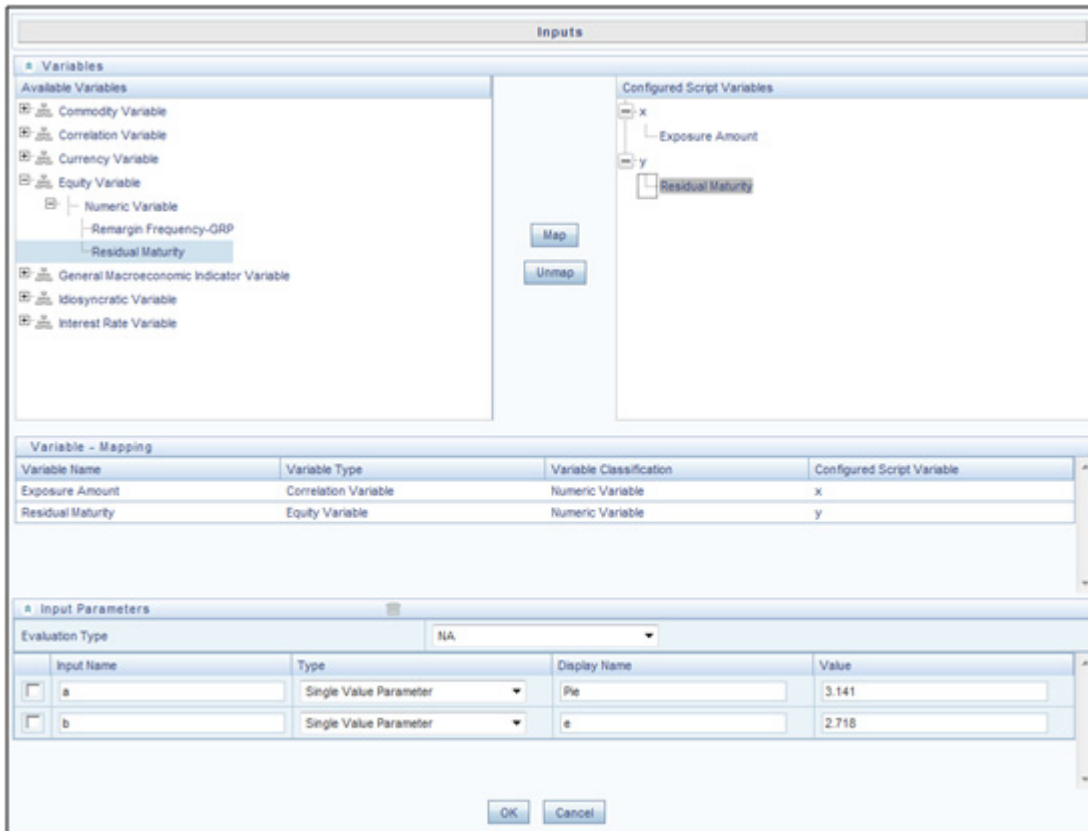
Figure 32. Inputs Window

> **Note:** You need to select the dataset before configuring input parameters.

5. Select the variable from the Available Variables pane, to which you want to map the variable declared in the script console, and click **Map**. The mapping details are displayed in the Variable-Mapping grid.
   To unmap, select the variable from the Configured Script Variables pane and click **Unmap**.

   > **Note:** You cannot select a variable in the Available Variables pane which is already mapped to another variable in the R script.

6. From the Input Parameters grid, select the **Evaluation Type** from the drop-down list. You can execute model on complete set of records in the dataset, or on a group of records, or on a set of rows.

   > **Note:** Standard R does not support group and row level executions.

   ● **Group**- Select this option if the model needs to be executed on a group of records. Select the Grouping Variable from the drop-down list.

   ● **Row**- Select this option if the model needs to be executed on a set of rows. Specify the Number of rows.

   > **Note:** For Evaluation Type as Row, graphical output (plot) is not supported.

   ● **NA**- Select this option if the model needs to be executed on complete set of records in the dataset.

   All single value parameters declared in the Model Script Console are displayed.

7. Enter the name to be displayed and the value in the Display Name and Value fields respectively.

Note: You can select an input parameter and click to delete it. The deleted input parameter is removed from the declaration block.

8. Click from the Script Console toolbar. The Outputs window is displayed. For more information, see the *Outputs* section.

9. Enter valid R script of the model using R and ORE functions as required. For more information on data usage and data handling when R script and ORE functions are used, see *R Scripts and Oracle R Enterprise (ORE) Statistical Functions*.

10. Click to execute the model.

   ● Select **Baseline** to execute the model normally.

   ● Select **Stress** to execute the model based on a stress scenario. The Execution Parameters window is displayed.
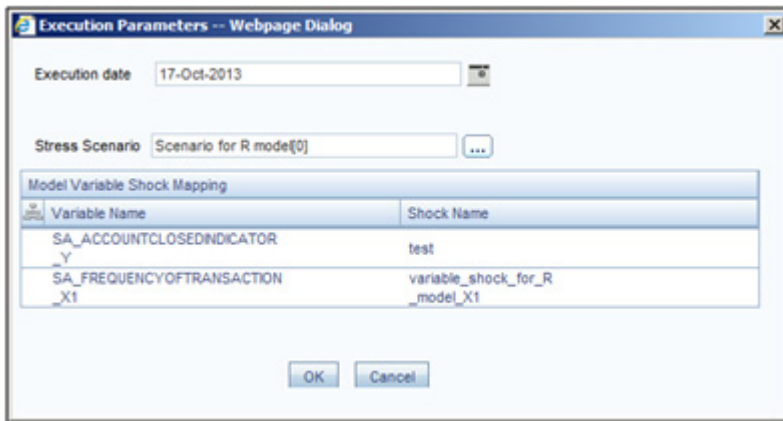
Figure 33. Execution Parameters

   ● Select the Execution Date from the calendar. The data of the selected date will be used for model execution.

   ● Click **Stress Scenario** to open the Scenario Browser window. Select the scenario and click . The variables and their shocks of the selected scenario are displayed in the Model Variable Shock Mapping grid. For more information on stress testing, see the *Stress Testing* section.

   You can view the status of the model execution dynamically in the Execution Status grid.

   Note: For R based models, the execution may fail if the dataset contains internal joins.

11. Click to view the output. For more information, see the View Model Outputs section.

Model Parameters

The grid in the Model Details section displays the various parameters applicable when a technique is selected. It is mandatory to update the required information in the tabs displayed and the parameters or fields in the available tabs vary depending on the technique selected. Few of the common input parameter types are explained below.

Variable

This tab is displayed for NAG techniques only.

In the Variable tab, you can add variables you want to use in the model definition.

1. Click from the Variable Selection toolbar. The Variable Browser window is displayed. It displays all the variables defined in the Variable Management window.
   or

   Click to create new variable

2. Select the required variable from the hierarchical members list and click .

3. Click **OK**. The selected Variable details are displayed in the grid. Multiple instances of a single variable can be selected and the variable summary grid displays all the variables with unique IDs.

4. Click in the variable grid to view the Variable Transformation details. You can add a technique for transformation.

5. Select a variable you want to delete and click to remove the variable.

Time Referencing

In the Time Referencing tab, you can apply a time filter to further slice the data based on time for specific models. Time referencing can be in terms of days, months, or years. You can specify a particular variable as the Anchor Variable and all the other selected variables are referenced relatively to the anchor variable.

1. In the Model Definition window, click adjacent to **Make Anchor Variable** in the Relative Referencing grid. The Anchor Variable Browser window displays the list of variables you have mapped in the Variable tab.

   Note: You need to map at least one variable from the Variable tab before adding Anchor variable.

2. Select the required variable from the hierarchical members list and click . The selected variable is added to the Selected Members pane.

3. Select the **Time Reference Period** as either **Days** or **Months** or **Years**.

4. Click **OK**. The Variables collapsible grid displays the list of variables in the model definition. The selected Time Referencing variable is displayed in the Model Definition window and is denoted with an anchor sign.

   Once you have defined a set of Time Referencing variables, you can click to view the details and to edit the details specifying the anchor position and Time Reference Values.

Filter

In the Filter tab, you can add multiple non-time hierarchy members as filters.

1. In the Model Definition window, click from the Filters toolbar. The Filter Browser window is displayed.

2. Select Data Filter or Hierarchy to display the corresponding filters.

3. Select the required filter or hierarchy from the list and click . The selected filter is added to the Selected Members pane.

4. Click **OK**. The selected Filters are displayed in the grid.

5. Filters can be applied to both Production and Sandbox Information Domains. Select the required option **Apply in Production** or **Apply in Sandbox** to apply the filter in the required information domain.

Note: Data Element Filters created in the Filters screen is displayed when you select **Data Filter**.

Sampling

Note: This is applicable for models based on NAG techniques only.

In the Sampling tab, you can specify the type of sample for the model definition. The available options are Stratified, Training, Test, and Control sample.

1. In the Sampling tab of the Model Definition window, select the **Basis of Sampling** as either in **Percentage (%)** or **Absolute** option.

2. Select **Replacements** option as either **Yes** to repeat the records with other samples, or **No** to restrict one record to one sample.

3. Select the **Stratification** (grouping) Basis based on distinct values. Click , the Stratification Basis browser is displayed.

4. Select the hierarchy in which you want to group the records and click . The selected filter is added to the Selected Hierarchies pane.

5. Click **OK**. The selected hierarchy is displayed in the grid. Usually there is a single level hierarchy available for Stratified sample selection.

6. Specify the **Training Sample** value, which is the default sample on which model fitting or predictions are carried out. It is mandatory to select a training sample, or the default 100% record set is assumed as Training Sample.

7. Add Test Sample by clicking , and specifying the Sample Name and Sample Size in the New Test Sample window. Click **OK**. You can also view, modify, and delete the test sample created.
    Ensure that the sum of Training Sample value and Test Sample values constitutes to 100%.

8. Specify the Control Sample by clicking [...] and selecting the Control Hierarchy and Control Hierarchy Node in the Control Hierarchy browser separately.

Inputs

The following sections describe the various inputs.

For R scripted Technique

The variables you have declared in the R script of the technique are displayed under the Configure Script Variables pane.

1. Select a variable from the Available Variables pane by expanding the required variable type and a variable in the Configured Script Variables pane, and click **Map**. The mapping details are displayed in the Variable-Mapping grid. To unmap, select the variable from the Configured Script Variables pane and click **Unmap**.

2. Click to [icon] add new variables. The Varaible Defintion page is displayed. You can create a new variable from the Model Definition page, as per the requirement. For more information, refer *Adding Variable Definition* section.

Note:

- You cannot select a variable in the Available Variables pane which is already mapped to another variable in the R script.

- Mapping of single variable declared in the R script to multiple variables is not supported if ORE implementation is used. However, for prepackaged ORE models, multiple variable mapping to single variable in the R script is supported.

3. From the Input Parameters grid, select the **Evaluation Type** from the drop-down list. You can execute model on complete set of records in the dataset, on a group of records or on a set of rows.

- **Group**- Select this option if the model needs to be executed on a group, which is defined using a grouping variable. Select the Grouping Variable from the drop-down list.

- **Row**- Select this option if the model needs to be executed on a set of rows. Specify the Number of rows.

Note: For Evaluation Type as Row, graphical output (plot) is not supported and the number of records processed is five times the value given in the Number of Rows field. These are limitations.

- **NA**- Select this option if the model needs to be executed on complete set of records in the dataset.

4. All single value parameters declared in the technique are displayed. Displays the Value if it was given while defining the technique. Else enter the appropriate value.

For External Library Based Technique

You can view the Parameters you have added during Technique Definition, for which you have selected the **Configured During Model Definition?** checkbox, under the Configured Script Variables pane.

1. Select a variable from the Available Variables pane (by expanding the required variable type) and a variable from the Configured Script Variables pane, and click **Map**. The mapping details are displayed in the Variable-Mapping grid.
   To unmap, select the variable from the Configured Script Variables pane and click **Unmap**.

2. Under the Input Parameters pane, the Single Value Parameters and Multi Value Parameters (for which the Configured During Model Definition? checkbox is selected) will be displayed. Enter the values for the displayed parameters.

For NAG Technique

Based on the selected NAG Technique, the Input tab varies. For some techniques, this tab will not appear.

In the Inputs tab, you can specify the link function associated to the selected technique and confirm if a Mean value is included during prediction for Transition Matrix techniques

1. Specify the **Link Function** by selecting from the drop-down list.

2. Specify if **Mean value** is included by selecting **Yes** or **No**.

3. In the Model Inputs tab, you can define the transition parameters for the historic data.

4. In the State Space grid, enter the details as described in the following table:

Table 9. NAG Technique - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Historical Transition Percentage	1. Click ⬚ . The Variable Browser window is displayed.
	2. Search for a particular Member by entering the keyword and clicking ⬚ in the Search grid. In the Members list, you can click + to expand each hierarchy member and view the subsequent members.
	3. Select the required member and click ⬚ or double click to move the member to the Selected Members pane. You can also remove a member by selecting from the Selected Members list and clicking ⬚
	4. Click **OK.** The selected member is displayed in the Historical Transition Percentage field.
Current State	Click ⬚ . The Hierarchy Browser window is displayed. Follow the above procedure (as explained to define Historical Transition Percentage) to define the Current State parameter.

Table 9. NAG Technique - Field Descriptions

Field	Description
Future State	Click [...]. The Hierarchy Browser window is displayed. Follow the above procedure (as explained to define Historical Transition Percentage) to define the Future State parameter.
State Space Sequence	Click [...]. The Hierarchy Browser window is displayed. Follow the above procedure (as explained to define Historical Transition Percentage) to define the State Space Sequence parameter. **Note:** You can select multiple states by holding Ctrl button while selecting. You can also sort the selected list by clicking [▲] and [▼] buttons.
Unity Adjustment State	Click [...]. The HierarchyMember Browser window is displayed. Follow the above procedure (as explained to define Historical Transition Percentage) to define the Unity Adjustment State parameter. **Note:** Ensure that you select a state which is present in the selected State Space Sequence list.
Absorbing State	Click [...]. The HierarchyMember Browser window is displayed. Follow the above procedure (as explained to define Historical Transition Percentage) to define the Current State parameter. **Note:** Ensure that you select a state which is present in the selected State Space Sequence list.

4. Select the Time Interval as Day, Week, Month, Quarter, Half-Year, or Year from the drop-down list.

5. Enter a numeric value in the **Number of Historic Intervals** field.

Note: The Number of Historic Intervals field is present only if you have selected the Calibrate Model option.

6. The options to define the transition parameters vary depending on the technique selected.

● If you have selected **Transition Matrix - EWMA**, enter the Decay Factor value (lesser than 1) in the EWMA Parameter grid.

● If you have selected **Transition Matrix - Linear Regression**, enter the Confidence Level value (lesser than 1) in the Regression Parameters grid.

● If you have selected **Transition Matrix - Multi Factor**, select the Time Homogeneous Estimation option as either YES or NO. In case you have selected No, enter numeric value in Number of Intervals Predicted field.

● You also need to select **Standard Normal Assumption** as either YES or NO in the Asset Value Parameter grid.

● If you have selected **Transition Matrix - Time Series**, enter the numeric values in following fields:

■ Number of Autoregressive Terms

■ Number of Non-seasonal Difference

■ Number of Lagged Forecast Error in the Prediction Equation

- ■ Seasonal Period

- ■ Order of Seasonal Differencing

- ■ Seasonal Autoregressive

- ■ Seasonal Moving Average Arguments

- ■ Number of Future Values

Outputs

The following sections describe the various Output options.

For R scripted Technique

The Outputs window allows you to choose outputs from the list of R objects in the output tree view. The tree is formed out of either the R objects declared as outputs in the declaration block, or the output of the last line of the script. That is, when no output is declared in the declaration block, the framework will follow the same logic as it is done by any regular R interface.

If the output section in the declaration block is left unfilled and the last line of the script is a graphical output (plot () or so), then no output tree will be generated as there are no non-graphic output (only the plots will come up in the results).

Note: You can view the output tree structure only after entering the R/ ORE script in the Model Script pane and configuring the input parameters.
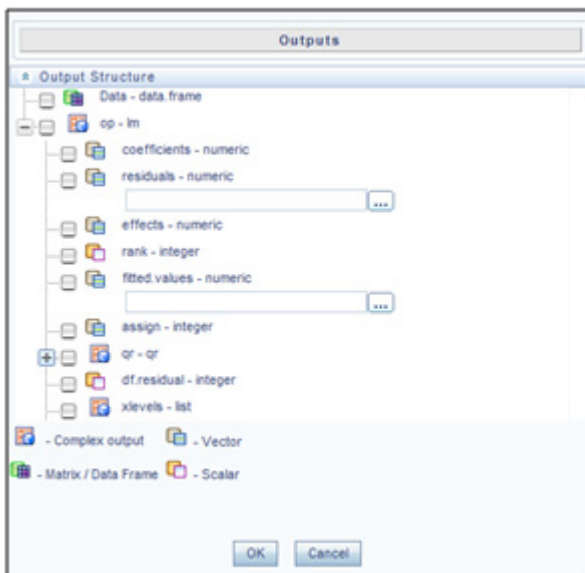
Figure 34. Outputs window

Select the outputs to be generated, by selecting the checkboxes corresponding to those outputs.

You can store the output values computed during the model execution to a table in your atomic schema by mapping the output value to a variable, which you have defined through the Variables Definition window. Click

corresponding to the output variables which you want to map to an attribute. From the Variable Browser window, select the variable.

Note: For models where ORE implementation is used, you cannot map the row level outputs to Variables.

Scenarios

Consider the following script where outputs are not configured. Then the framework output will be just the graphical output. The output tree will not be formed as there are no non-graphical results. That is, when the outputs are not chosen by the user, then the framework reports the last line's output as the only default output.

Note that the plots are always produced.

```
##  ---------DECLARATION------------
## Variable --> DepVar,IndepVar1,IndepVar2
## Single Value Parameter -->
## Output -->
##  ------------------------------
art.mod<-lm(DepVar ~ IndepVar1 + IndepVar2)

art.summ<-summary(art.mod)
coef( art.summ )
art.summ[[ "r.squared" ]]

#do some line plots
#plot()

#form new data.frame object
new.x.datafrme = data.frame(x=seq(from=range(IndependentVariable1)[ 1 ],to=range(
IndependentVariable1)[2],length=length(DependentVariable)))

#form new regression model
RegModel<-lm(DepVar ~ new.x.datafrme$x + IndepVar2)

#perform some operations on the new data
Plot(RegModel)
```

In this case, the output window will have a tree containing the components of 'RegModel' object with the same name as the root node.

Alternatively, you can also put the intended objects name at the last line of the script as:

```
##  ---------DECLARATION------------
## Variable -->
## Single Value Parameter -->
## Output --> RegModel
##  ------------------------------
art.mod<-lm(DepVar ~ IndepVar1 + IndepVar2)

art.summ<-summary(art.mod)
coef( art.summ )
art.summ[[ "r.squared" ]]
```

```
#do some line plots
#plot()

#form new data.frame object
new.x.datafrme = data.frame(x=seq(from=range(IndependentVariable1)[ 1 ],to=range(
IndependentVariable1)[2],length=length(DependentVariable)))

#form new regression model
RegModel<-lm(DepVar ~ new.x.datafrme$x +
IndepVar2)

#perform some operations on the new data
Plot(RegModel)
RegModel
```

Here the output window will show a tree containing the components of 'RegModel' object, but since the output name was not configured or specified, it will be held under a generic framework name 'OFSAAOutput'

One of the added advantages of configuring the outputs is that one can get more than one object as output as shown in the following script:

```
##  ---------DECLARATION------------
## Variable -->
## Single Value Parameter -->
## Output --> RegModel, art.mod
##  ------------------------------
art.mod<-lm(DepVar ~ IndepVar1 + IndepVar2)

art.summ<-summary(art.mod)
coef( art.summ )
art.summ[[ "r.squared" ]]

#do some line plots
#plot()

#form new data.frame object
new.x.datafrme = data.frame(x=seq(from=range(IndependentVariable1)[ 1 ],to=range(
IndependentVariable1)[2],length=length(DependentVariable)))

#form new regression model
RegModel<-lm(DepVar ~ new.x.datafrme$x +
IndepVar2)

#perform some operations on the new data
Plot(RegModel)
```

For the above script, the output window will contain trees for both 'RegModel' and 'art.mod'.

Here are the possible scenarios and the respective framework behavior:

- **Case A**: When no outputs are specified in the declaration block and outputs are not configured in the Outputs window:
  The value of the last evaluated expression is returned with the name 'OFSAAOutput' for the model execution.

- **Case B:** When no outputs are specified in the declaration block and the Configure Outputs button is clicked:
  All the objects from the script execution environment get listed as a tree with each object as separate nodes. Now you can choose the outputs to be reported. The selected ones will be placed in the declaration block. If you do not select any object as output, then the behavior will be the same as case A, for the model execution.

- **Case C:** Some objects are declared as outputs in the declaration block either by explicitly writing the correct names in the declaration block against the label Outputs, or by right-clicking a valid object name and selecting it as output:
  Now, only those which are listed in the declaration block will come in the Configure Outputs window (as checked ones, since you have already selected them as the outputs explicitly).

For NAG Techniques

In the Outputs tab, there are two grids namely, Process Output Selection and Data Output variables Selection. In the Process Output Selection grid, you can specify the type of output which can be derived when predictions are carried out.

1. In the Model Definition window, click from the Process Output Selection toolbar. The Process Output browser is displayed.

2. Select the required output type from the Members list and click . The selected process output is added to the Selected Members pane.

3. Click **OK**. The selected Process Outputs are displayed in the grid.

4. Select the **Process Output Required in Tables** checkbox to store process output in your Database with Name and Comments.

In the Data Output Variables Selection grid, you can populate the variable data into the tables as another filtering parameter.

5. Click from the Data Output Variable Selection toolbar. The Data Output browser is displayed.

6. Select the required Data Output variable from the Members list and click . The selected Data output filter is added to the Selected Members pane.

7. Click **OK**. The selected Data Outputs are displayed in the grid.

In the Data Output Variable Selection toolbar, you can click , to view the properties of Data output variable transformation and click to delete the Data output.

Creating Models using Logical Sandbox

This feature allows you to create models in logical sandbox. In this case, the production and the sandbox information domains are the same. You can define the model from the production infodom by selecting the required logical sandbox from the Sandbox drop-down list provided in the Modeling window.

Note:

- The model management window displays only the models which are defined in the selected Logical sandbox.

- While creating model definition in the sandbox, you can select only those datasets which are selected in the sandbox definition.

- There will be no deployment process for the models defined in the logical sandboxes.

To create a model using logical sandbox, follow these steps:

Note: If a logical sandbox is defined in a production information domain, then only you will have the Model Management LHS menu in the Modeling window.

1. Click **Model Management**. The models defined in the selected logical sandbox are displayed in the Model Management window.

2. Click ![icon] from the Model Management toolbar. The Model Definition window is displayed. For more information, see section *Creating and Executing a Model.*

Viewing Model Definition

You can view individual Model Definition details at any given point. To view the existing Model Definition details in the Model Definition window, follow these steps:

1. Select the checkbox adjacent to the Model ID.

2. Click ![icon] in the Model Management toolbar.
   An information dialog is displayed indicating that view mode is selected and any accidental changes will not be updated. Click **OK** to confirm and view the details.

3. In the Model Definition View window you can:

- Select **Preview Data** to view the variable parameters corresponding to the technique selected.

- Select **Execute** to process the model execution. The status of which can be verified in Model Execution Status option, accessing *Oracle Financial Services Enterprise Modeling > Modeling > Model > Model Execution Status.*

Modifying Model Definition

You can update the model definition details of an existing Model in the Model Definition window:

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

2. Click in the Model Management tool bar. The Edit 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 the variable parameters based on the technique selected. For more information, refer to *Creating and Executing a Model*.

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

   ● Click **Preview Data** to view the new Model Definition details before upload.

   ● Click **Save** to update the model definition details. You can choose save a Model Definition as a new version or overwrite the existing version. For more details, refer to *Model Versioning* section.

   ● Click **Save** and select **Execute** to process the model execution. The status of which can be verified in Model Execution Status option, accessing *Oracle Financial Services Enterprise Modeling > Modeling > Model > Model Execution Status*.

Deleting Model Definition

You can remove the model definitions which are no longer required by deleting from the Model Management window.

1. Select the checkbox adjacent to the Model ID(s) whose details are to be removed.

2. Click in the Model Management tool bar.

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

Searching Model Management

The Model Management page allows you to search the models that you want to view.

To search for models, follow these steps:

1. Navigate to the Model Management page.

2. Enter the Model Name and Model Objective.

3. Click . The filtered list is displayed.

Model Chaining

Model Chaining can be used for separation of calibration and prediction. You can script models to map the fitted output and persist the results as ORE object in datastore or in a file in case of standard R engine. You can specify which model's output you want to use for predicting the results.

The following tags are available in the Model Script Console:

- **##Model Input-->** Enter the object names you want to map the fitted output.

- **##Model Output-->** Enter the object names you want to persist.

Note: You cannot create a prediction model on ORE Engine using Fitting Model executed on Standard R Engine as input and vice-verse.

Figure 35. Model Script Console

Model Execution Status

Model Execution Status in the Sandbox Information Domain enables you to verify the status of the executed model at any given point. The status can be Complete, Ongoing, or Failed based on the current processed stage of the system.

This section includes the following topics:

- Accessing Model Execution Status
- Searching Model Execution Status

Accessing Model Execution Status

To access the Model Execution Status page, follow these steps:

1. Click **Sandbox** tab.

2. Click **Model Execution Status**. The Model Execution Status page is displayed.

Note: This menu is displayed only if you are in sandbox infodom.

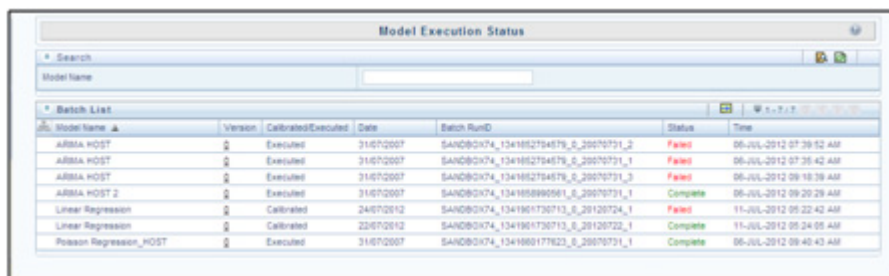**Figure 36. Model Execution Status**

Searching Model Execution Status

The Model Execution Status page allows you to search the models that you want to view.

To search for Model Execution, follow these steps:

1. Navigate to the Model Execution Status page.

2. Enter the Model Name.

3. Click [icon]. The filtered list is displayed

Model Outputs

The Model Outputs page in Sandbox Information Domain displays all the outputs of the model executed in the Sandbox Information Domain. The Model Outputs page in Production Information Domain displays the outputs for the models deployed in the production Information Domain and also those requested for execution. You can also view or download the derived output values as a .csv file and plots as .png files for reference.

This section includes the following topics:

- Accessing Model Outputs

- Searching Model Outputs

- View and Compare Model Outputs

Accessing Model Outputs
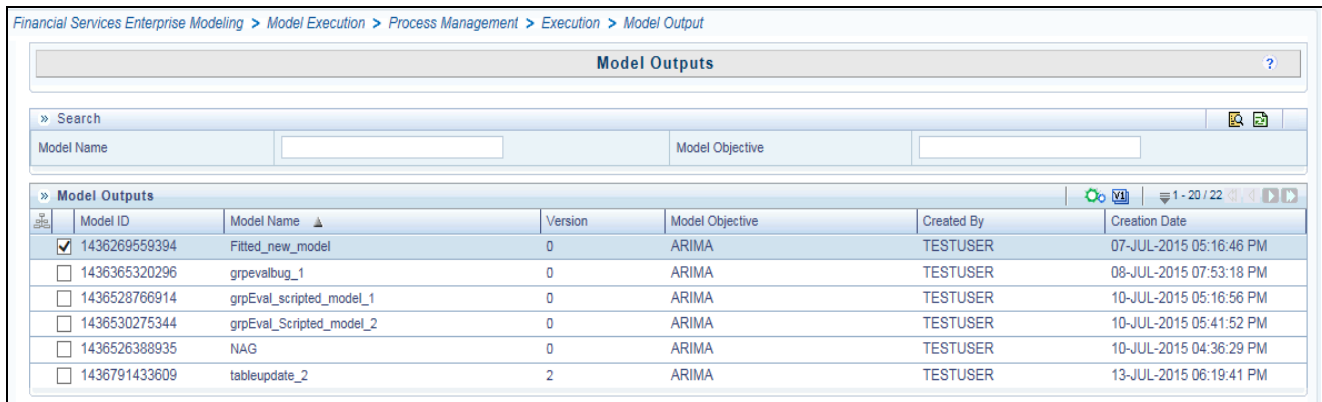
You can access the Model Outputs page from the Application tab and the Sandbox tab.

To access the Model Outputs page from Application tab, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click [icon] to expand the Model Execution.

3. Click [icon] to expand the Process Management.

4. Click [icon] to expand the Execution.

5. Click **Model Outputs**. The Model Outputs page is displayed.

To access the Model Outputs page from Sandbox tab, follow these steps:

1. Click **Sandbox** tab.

2. Click **Model Outputs**. The Model Outputs page is displayed.

Figure 37. Model Outputs

Searching Model Outputs

The Model Outputs page allows you to search the models that you want to view.

To search for Model Outputs, follow these steps:

1. Navigate to the Model Outputs page.

2. Enter the Model Name and Model Objective.

3. Click ![icon]. The filtered list is displayed.

View and Compare Model Outputs

To view and compare the model outputs, follow these steps:

1. Select the checkbox adjacent to the Model ID.

2. Click ![icon] from the Model Outputs toolbar. The Output Summary window is displayed.
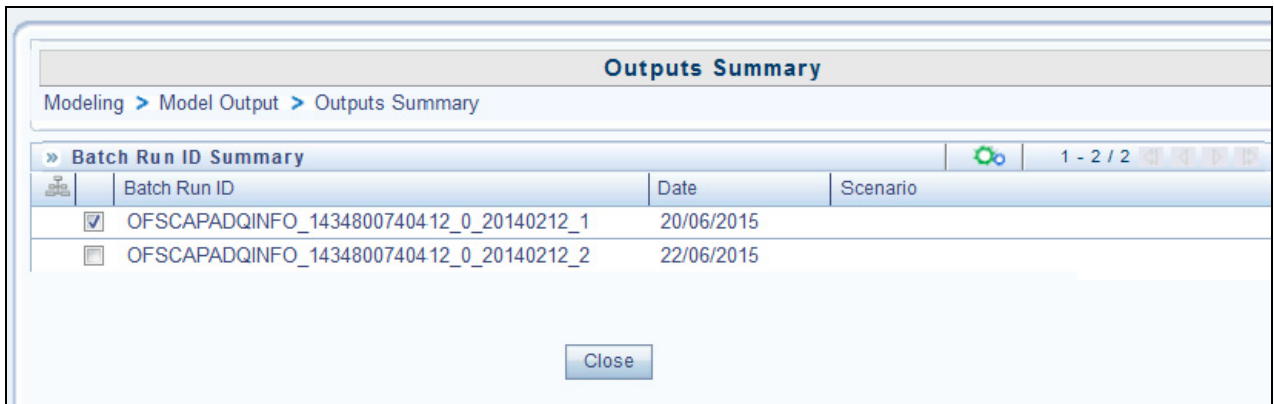
Figure 38. Outputs Summary

3. Select the checkbox adjacent to a Batch Run ID to view the details or select multiple checkboxes to compare Batch Run IDs.

4. Click ⚙ to view details or compare Batch Run ID(s). The View/Compare Sample window is displayed with the comparison details of the selected models.
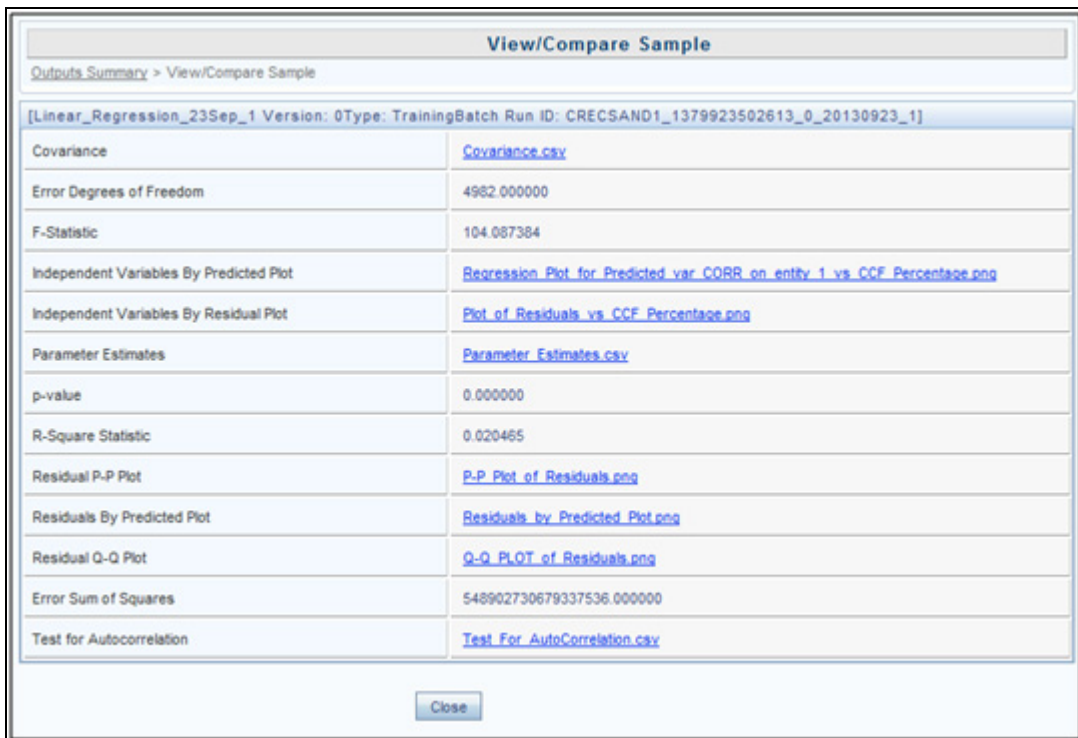
Figure 39. Output Window for NAG Technique Based Models

5. Click the links to view/download the derived values stored as .csv files or plots saved as image files.

Model Deployment
Chapter 7–Managing Modeling
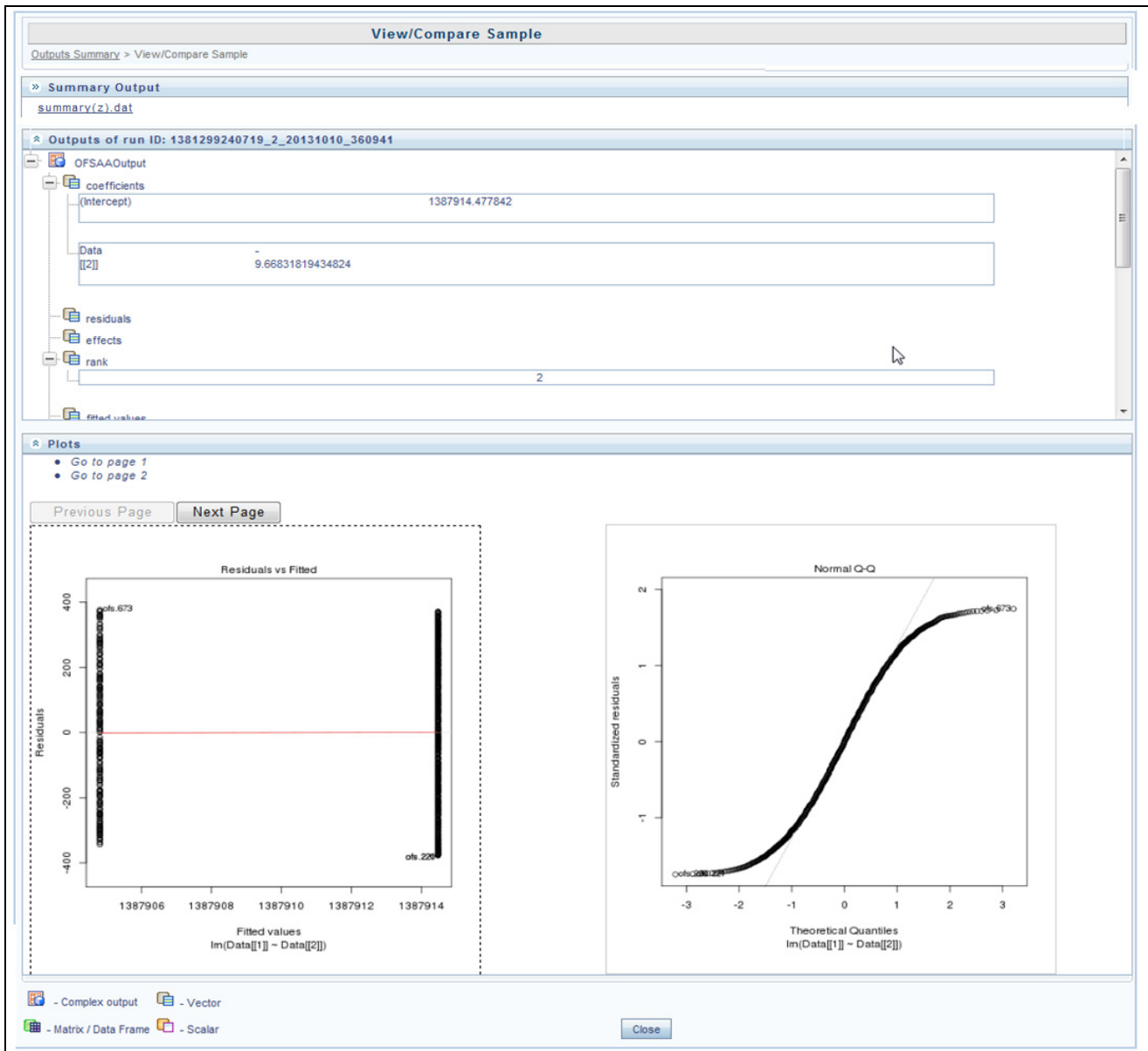
Figure 40. Output window for R scripted models

6. Click the file name (.dat file) under the Summary Output pane to view the summary of outputs.This is displayed only for models based on Standard R Engine.

7. Click the links in the Outputs pane to download or view the derived values in .csv files.

Model Deployment

Model Deployment in the Sandbox Information Domain enables you to request for deployment of a model to the production information domain. Since models based on R scripting can be directly deployed without execution,

those models will be displayed in the Model Deployment window. In case of models based on NAG techniques, the successfully executed models only will be displayed.

Note: Model deployment is not required for models created in logical sandboxes.

This section includes the following topics:

- Accessing Model Deployment
- Model Deployment Authorization
- Searching Model Deployment

Accessing Model Deployment

To access the Model Deployment page, follow these steps:

1. Click **Sandbox** tab.
2. Click **Model Deployment**. The Model Deployment page is displayed.

Note: This menu is displayed only if you are in sandbox infodom.

Figure 41. Model Deployment Page

Model Deployment Authorization

To request and authorize for model deployment, follow these steps:

1. Select the checkbox adjacent to Model ID and click   . The Model Deployment Authorization window is displayed.
2. Select the **Request for Deployment** checkbox in the Model Deployment grid.
3. If you have been mapped to MDLAUTH function role, select **Authorize and Deploy** checkbox to authorize the deployment request. Otherwise, an authorizer has to authorize for deployment.
4. Click **Save** to process for model deployment.

When there is a request for model authorization and deployment, the data in both sandbox and production Information Domains are validated for compatibility and then deployed into the production Information Domain. The status of processing is displayed in an information dialog.

After successfull deployment of the model it will be available in Run Rule Frmaework as component. Run filter and process filter are supported for models stitched in a Run for execution.

Note: For execution of Open R models, the value for `$HOST` should be passed as a Run Time parameter.

Searching Model Deployment

The Model Deployment page allows you to search the models that you want to view.

To search for Model Deployment, follow these steps:

1. Navigate to the Model Deployment page.

2. Enter the Model Name and Model Objective.

3. Click . The filtered list is displayed.

Model Execution

The Model Execution module allows to you request for model execution and work with champion challenger.

This section includes the following topics:

- Request for Model Execution
- Champion Challenger

Request for Model Execution

Request for Model Execution in the Production Information Domain enables you to request for a batch execution for the selected model. A batch is scheduled in the Batch Processing and the model is executed in the production environment.

This section includes the following topics:

- Accessing Request for Model Execution
- Requesting for Model Execution
- Searching for Request for Model Execution

Accessing Request for Model Execution

To access the Request for Model Execution page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click to expand Model Execution.

3. Click ▶ to expand Process Management.

4. Click ▶ to expand Model Management.

5. Click **Request for Model Execution**. The Request for Model Execution page is displayed.
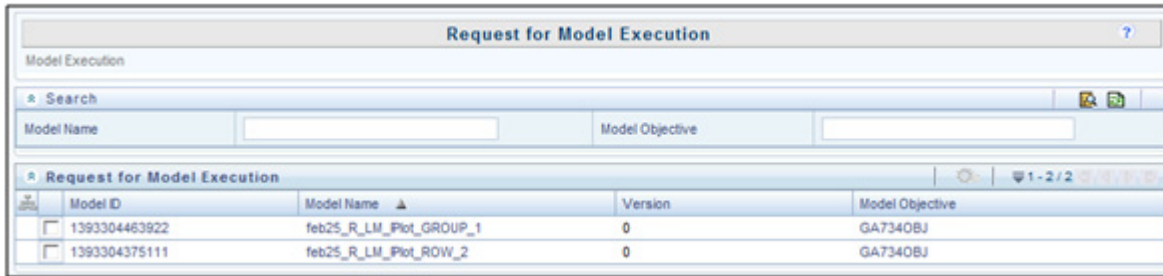
Figure 42. Request for Model Execution Page

Requesting for Model Execution

To request for model execution, follow these steps:

1. Select the required model and click 🔵 . The Request for Model Execution window is displayed.

2. Select the **Register Batch** checkbox.

3. Click **Save** to process for model execution.

When there is a request for model executin, a batch is registered in the Operations framework which has to be executed from the Batch Scheduling window in the application. For more information, refer *Operations* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide.*

Searching for Request for Model Execution

The Request for Model Execution page allows you to search specific model execution request.

To search for Request for Model Execution, follow these steps:

1. Navigate to the Request for Model Execution page.

2. Enter the Model Name and Model Objective.

3. Click 🔍 . The filtered list is displayed.

   Note: The Request for Model Execution page displays all deployed models in case of schema based sandboxes.

Champion Challenger

Champion refers to the first model deployed, after calibrating the parameters for a specific model objective. Challenger(s) refers to the subsequent models deployed for the same model objective but with different techniques used for predictions.

When several prediction models (challengers) are deployed and one of the challengers produces optimum output meeting all the prediction requirements, the same can be set as Champion.

This section includes the following topics:

- Accessing Champion Challenger
- Searching for Champion Challenger
- Requesting New Champion

Accessing Champion Challenger

To access the Champion Challenger page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click ▶ to expand Model Execution.

3. Click ▶ to expand Process Management.

4. Click ▶ to expand Model Management.

5. Click **Champion Challenger**. The Champion Challenger page is displayed.
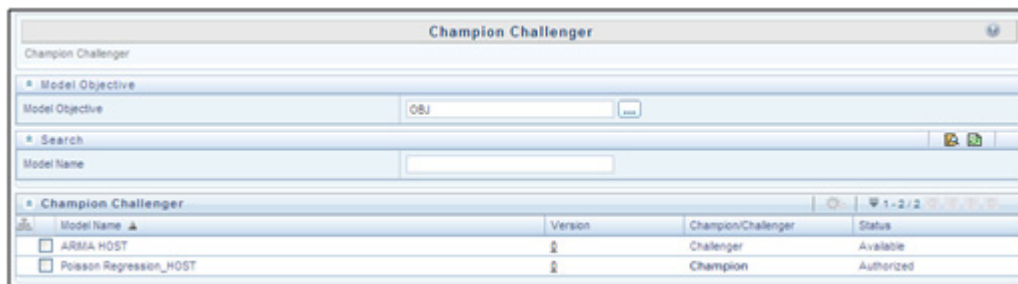
Figure 43. Champion Challenger

Requesting New Champion

To request a new champion, follow these steps:

1. Select the Model Objective, click [...] and open the Model Objective browser.

2. Select the required Model Objective in the Hierarchy list and click ▶ .

3. Click **OK**. The window is refreshed to display the available models.

4. Select the checkbox adjacent to the model whose status has to be set to champion.

5. Click ⚙ in the Champion Challenger toolbar. The Request New Champion window is displayed.

6. Select the **Request New Champion** checkbox.
   If you have been mapped to MDLCHAMP function role, you can select **Authorize** checkbox to authorize new champion.

7. Click **Save** to update the changes and return to Champion Challenger window.
   In the Champion Challenger window, the status of the selected model is updated as Champion and Authorized.

Searching for Champion Challenger

The Champion Challenger page allows you to search for a specific champion challenger.

To search for Champion Challenger, follow these steps:

1. Navigate to the Champion Challenger page.

2. Enter the Model Name.

3. Click . The filtered list is displayed.

CHAPTER 8 *Managing Stress Testing*

This chapter provides information about Modeling in the Oracle Financial Services Enterprise Stress Testing application and step-by-step instructions to use this section.

This chapter includes the following options:

- About Stress Testing
- Variable Definition
- Variable Shock Library
- Scenario Management
- Stress Definition

About Stress Testing

Stress Testing also facilitates banks to conduct analysis to estimate the impact of movements in the variables on specific measures such as profitability and capital adequacy. Stress testing supports the stress testing requirements across the entire suite of OFSAA products. It allows banks to define shocks and assess the impact of such shocks across multiple business areas.

Stress Testing Workflow
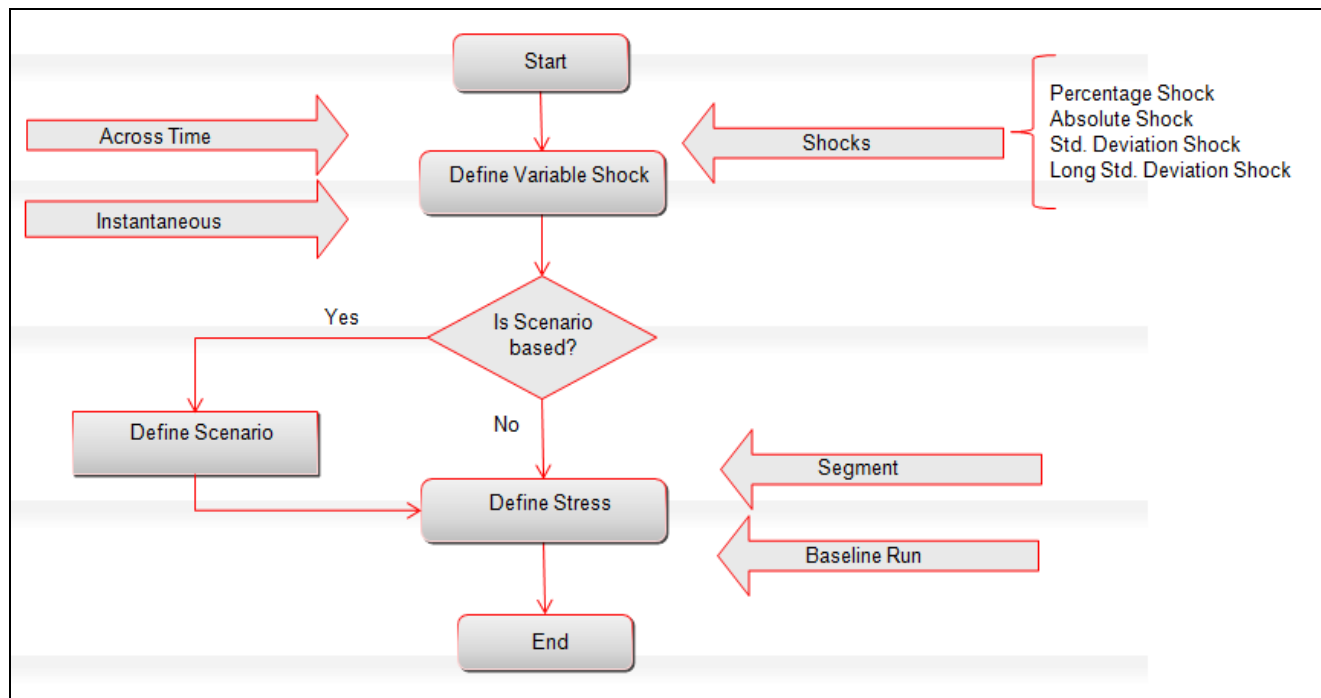
Figure 44. Stress Testing Workflow

Variable Definition

Variable refers to a logical set of attributes that are likely to change based on the selected parameters. In a modeling environment, variables play a vital role in filtering the model parameters and deriving an estimate based on historical data. Variables are defined in the production information domain.

This section includes the following topics;

- Accessing Variable Definition
- Adding Variable Definition

Accessing Variable Definition

1. Login to Oracle Financial Services Enterprise Modeling as a Business User. The Oracle Financial Services Enterprise Modeling Home page is displayed.

2. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

3. Click ▶ to expand the Stress Testing section.

4. Click **Variable Definition**. The Variable Management page is displayed.

Figure 45. Variable Management

Adding Variable Definition

The Add Variable Definition page enables you to add a variable by filtering through the required parameters. The options available to filter the data vary depending on the variable type selected.

To add variable definition, follow these steps:

1. Select ⊞ from the Variable Management tool bar. The Add button is disabled if you have selected any checkbox in the grid. The Variable Definition Add window is displayed.

Figure 46. Variable Definition

2. Enter the Variable details as described in the following table:

Table 10. Variable Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Folder	Select the folder in which you want to create the variable.
Variable Name	Enter a name for the variable. Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces.
Variable Description	Enter the required description for the variable.

Table 10. Variable Definition - Field Descriptions

Field	Description
Variable Type	Select the variable type from the drop down list. The available options are: • Commodity Variable • Correlation Variable • Currency Variable • Equity Variable • General Macroeconomic Indicator Variable • Idiosyncratic Variable • Interest Rate Variable You can add a custom variable type by clicking and entering a **Variable Type** name in the Variable Type window.
Variable Structure	Select the Variable Structure as either Single Value or Term Structure from the drop down list. The Term Structure option is available only if the Variable Type is selected as Commodity Variable or Interest Rate Variable. Click from the Term Structure toolbar. Select the required variables from the Variable Browser window.
Single Maturity	This field is displayed only if Commodity Variable or Interest Rate Variable is selected as Variable Type: • Select the Maturity Unit as Days, Months, Quarters, or Years from the drop-down list. • Select the Maturity Type as either Value or Node.
Value	This field is displayed only if Commodity Variable or Interest Rate Variable is selected as **Variable Type** and Maturity Type is selected as **Value**. Specify a numeric value in the Maturity Value field.
Node Selection	This field is displayed only if Commodity Variable or Interest Rate Variable is selected as Variable Type and Maturity Type is selected as Node. 1. Click and select the required hierarchy from the Hierarchy browser window. 2. Click and select the required node of the selected hierarchy from the Hierarchy browser window.
Based On	Select any of the following options from the drop-down list, based on which the variable definitions are to be created. • Hierarchy • Measure • Business Processor • Entity

Table 10. Variable Definition - Field Descriptions

Field	Description
Variable Classification	Select the required variable classification from the drop-down list. The options are Numeric Variable, Nominal Variable, and Ordinal Variable. The Nominal Variable and Ordinal Variable options are available only if you select General Macroeconomic Indicator or Idiosyncratic Variable as Variable Type and Hierarchy as Based On.

Variable Transformation	This feature is available for variables based on Measures, Business Processor, and Entity.
	
	1. Click to define a transformation for the Variable. The Technique Browser for Transformation window is displayed.
	
	2. Select the transformation you want to apply to the variable and click . The options are Arithmetic, Exponential, Logarithm, Power, and Trigonometric.
	
	3. Click or to change the order in which the transformation has to be applied. Click **OK**.
	  • If Arithmetic is selected, select the Operator (Addition, Subtraction, Multiplication, and Division) from the drop-down list and enter the Operator Value.
	  • If Logarithm is selected, select the Base from the drop-down list. The options are 10, 2, and e.
	  • If Power is selected, enter the value of Power.
	  • If Trigonometric is selected, select the Function from the drop-down list. The options are Sine, Cosine, and Tangent.
	  • If Exponential is selected, the exponential value of the variable will considered for transformation of the variable.

4. If the variable is based on hierarchy, follow these steps:

   • Select **Variable Property** as Direct or Dummy (variable in Binary form and used mainly in regression analysis). The Dummy option is available only if General Macroeconomic Indicator or Idiosyncratic Variable is selected as Variable Type.

   • Select the required Hierarchy from the drop-down.

   Note: Hierarchy-based variables cannot be created on multi-level Business Intelligence hierarchies.

5. If the variable is based on Measures, follow these steps:

   • Select **Yes** adjacent to Apply Filters.

   • Select the Datasets from the drop-down.

     You can click to view the details of the selected datasets.

   • Select the Measures from the drop-down.

     You can click to add a new Business Measure. For more details on creating a Business Measure, refer to *Creating Business Measure* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

- Click in the Filters grid and the Hierarchy browser is opened.

- Select the hierarchy member(s) and click **OK**. The selected members are displayed in the Filters grid under Hierarchies.

- Double-click a hierarchy member to invoke a grid with the member name.

- Click in the hierarchy member grid and the Hierarchy Browser is displayed.

- Select the node(s) and click **OK**. The selected nodes are displayed in the Hierarchy member grid under Nodes.

- Click and specify the **Selected Measures** by selecting from the Measure Browser.

6. If the variable is based on Business Processor, follow these steps:

   - Select the Business Process from the drop-down.

     You can click to add a new Business Processor. For more details on careting a Business Processor, refer to *Adding a Business Processor* section in *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

7. If the variable is based on Entity, follow these steps:

   - Click Entity to open the Entity and Attribute window. The available tables are displayed under the Available Entities section.

   - Select the checkbox corresponding to the table you want to select. The available columns in the selected table are displayed under the Available Attributes section.

   - Select the Attribute for which you want to create the variables and click **Save**. The selected attributes are displayed in the Selected Attributes section.

8. Click **Save** to upload the new Variable definition details. The User Info grid at the bottom of Variable Definition Add window displays metadata information about the Variable definition created.

Variable Shock Library

Variable Shock refers to modulating the data causing an adverse shift in the value of a variable. The Variable Shock Library within the Stress Testing framework facilitates you to define multiple shocks to variables and maintaining a library of such shocks in the Production Information Domain. You can also view, modify, copy, and delete variable shock definitions.

This section includes the following topics:

- Accessing Variable Shock Library

- Adding Variable Shock Definition

- Viewing Variable Shock Definition

- Modifying Variable Shock Definition

- Copying Variable Shock Definition

- Deleting Variable Shock Definition

- Searching Variable Shock Library

Accessing Variable Shock Library

To access the Variable Shock Library page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click ▶ to expand Stress Testing.

3. Click **Variable Shock Library**. The Variable Shock Library page is displayed.

Figure 47. Variable Shock Library

Adding Variable Shock Definition

The Add Variable Shock Definition facilitates you to define a variable shock by filtering through the specific required parameters. The options available to filter the data vary depending on the variable type selected.

To add variable shock definition, follow these steps:

1. Select ⬛ from the Variable Shock List tool bar. The Add button is disabled if you have selected any checkbox in the grid. The Variable Shock Definition Add window is displayed.

Figure 48. Variable Shock Definition

2. Enter the Variable shock details as described in the following table:

Table 11. Variable Shock Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Shock Name	Enter a name for the variable shock. Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces.
Shock Description	Enter the required description for the variable shock.
Dataset	1. Click [icon] and open the Dataset browser. The available datasets are listed in the Members pane. 2. Select the required Dataset based on which the variable shock is to be created and click [icon] . 3. Click **OK**. **Note:** The Datasets based on Derived Entities are not supported.

Table 11. Variable Shock Definition - Field Descriptions

Field	Description
Variable Name	1. Click [...] and open the Variable browser. The defined variables are listed in the Members pane. 2. Select the required variable based on which the variable shock is to be created and click ▶. 3. Click **Ok.** The fields Variable Type, Classification, Structure, and Property are auto populated based on the variable selected.
Time Specification: Specify the shifts in the value across multiple time points.	
Time Hierarchy	1. Click [...] and open the Hierarchy browser. The available Time Hierarchies are listed in the Members pane. 2. Select the required Time Hierarchy based on which the variable shock is to be created and click ▶. 3. Click **Ok.**
Shock Type	Select the shock type from the drop-down list, as either: • **Across Time** - to specify shocks at multiple time points and the occurrence of which might be in the past or in the future. • **Instantaneous** - if you don't have specific time points and want to shock the data based on the current values. Selecting Instantaneous will disable the other options in the Time Specification of Shock grid.
(Optional) If **Across Time** option is selected in the Shock Type	Select **Shock in Reference to** from the drop-down list as either **Current Value** or **Future Estimated Value**. If Current Value is selected: 1. Select **Time Point** as either **Standard** or **Custom**. 2. Specify the frequency from the drop-down list as either Days or Weeks or Months or Years. 3. Specify **Number of Time Points** and click [↪]. The Time Point values are auto populated. The Shock Current Value checkbox is selected by default. If Future Estimated Value is selected, you can specify the Time Point, Frequency, and Number of Time Points as indicated above. 4. Click [...] and open the Model browser. The models which are successfully executed in sandbox and deployed to production are listed in the Members pane. 5. Select the required model and click ▶. 6. Click **OK.**

Table 11. Variable Shock Definition - Field Descriptions

Field	Description
Filter Specification: Define the filtering criteria for the specified shock. The option is available only for Idiosyncratic Variables. Multiple shocks can to be applied across filters which are specified based on combinations of dimensions. **Note:** While defining a Variable Shock across time, a Dataset with Date Filter should be selected. For more information, refer Creating Data Set section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*. 1. Click [icon] and open the Filter browser. 2. Click [icon] in the Filter Browser and open the Hierarchy browser. 3. Select the required hierarchy(s) and click [icon]. 4. Click **OK** and return to the Filter Browser. 5. In the Hierarchical list click [icon] to expand a node and select the required checkbox. 6. Click [icon] and move the selection to the Selected Hierarchy pane. 7. Click **OK**.	
Shock Parameters: Specify quantum of shifts in variables, based on time specification and shock curve.	
Shock Unit	Select the shock unit from the drop-down list. The available options are; ● Percentage Shift ● Absolute Shift ● Standard Deviation Shift ● Long Standard Deviation Shift A shock value may be positive or negative.
Time Window Size	If you have selected Standard Deviation Shift or Long Standard Deviation Shift as Shock Unit, then the Time Window Size is enabled with a default value "2". You can change the same by entering the required value.

Table 11. Variable Shock Definition - Field Descriptions

Field	Description
Shock Curve	Shock Curve is enabled for term structure variables. Select any of the following option from the drop-down list. ● Parallel ● Twist ● Inversion
Shock Values	Shock Value is the quantum of shift in the variable. You can specify Shock Values across filters and time points. You can specify a shock that involves a transition from one category to another or map the selected to a shock category. 1. Click ☒ in the Shock Value toolbar. The Category browser is displayed. 2. Select the required category and click ▶. 3. Click **OK**. 4. Click ⮕ to generate the list of shock values that are selected. 5. Click ⋯ against the listed category and specify the values for each category in the Category browser. 6. Click **OK**.

3. Click **Save**. The new variable shock details are saved and displayed in the Variable Shock Library window.

Viewing Variable Shock Definition

You can view individual Variable Shock Definition details at any given point.

To view the existing Variable Shock Definition details, follow these steps:

1. Select the checkbox adjacent to the Shock ID.

2. Click ▤ from the Variable Shock List tool bar. The Variable Shock Definition View window is displayed with the variable shock details.

Modifying Variable Shock Definition

You can modify only the Shock Description if the selected shock is mapped to a Variable definition. If not, then all the filtering parameters can be modified except the Shock Name. When you modify any of the Variable Shock parameters other than the Shock Description, the details are uploaded as an incremented version without overwriting the existing definition details.

To modify an existing Variable Shock Definition follow these steps:

1. Select the checkbox adjacent to the Shock ID whose details are to be updated.

2. Click ▧ in the Variable Shock List tool bar. Edit button is disabled if you have selected multiple Shock IDs. The Variable Shock Definition Edit window is displayed.

3. Edit the Variable Shock Definition details as required. For more information, refer to *Add Variable Shock Definition*.

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

Copying Variable Shock Definition

The Copy Variable Shock Definition facilitates you to quickly create a new Shock Definition based on the existing shock variables or by updating the values of the required variables.

To copy an existing Variable Shock Definition follow these steps:

1. Select the checkbox adjacent to the Shock ID whose details are to be duplicated.

2. Click in the Variable Shock List tool bar. Copy button is disabled if you have selected multiple Shock ID(s).

3. In the Variable Shock Definition Copy window, you can:

   ● Create new variable shock definition with existing variables. Specify a new Variable Name and click **Save**.

   ● Create new variable shock definition by updating the required variables. Specify a new Variable Name and update the required details. For more information, refer *Add Variable Shock Definition*. Click **Save**.

The Variable Shock details are displayed in the Variable Shock Library window.

Deleting Variable Shock Definition

You can remove the variable shock definition(s) which are not used in Scenario Definition and which are no longer required in the system, by deleting from the Variable Shock Library window.

To delete for Variable Shock Definition, follow these steps:

1. Select the checkbox adjacent to the Shock ID whose details are to be removed.

2. Click in the Variable Shock List tool bar.

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

Searching Variable Shock Library

The Variable Shock Library page allows you to search specific model execution request.

To search for Variable Shock Library, follow these steps:

1. Navigate to the Variable Shock Library page.

2. Enter the Variable Shock Name.

3. Click . The filtered list is displayed.

Scenario Management

Scenario in Stress Testing refers to a set of unusual, hypothetical events structured within the variables. A scenario can be defined using variable based shocks and every single Variable Shock constitutes a scenario.

Scenario Management within the Stress Testing enables you to define and maintain multiple Scenarios on the same set of Variables in the Production Information Domain. You can also view, modify, copy, and delete scenario(s).

This section includes the following topics:

- Accessing Scenario Management
- Adding Stress Testing Scenario
- Viewing Stress Testing Scenario
- Modifying Stress Testing Scenario
- Copying Stress Testing Scenario
- Deleting Stress Testing Scenario
- Searching Scenario Management

Accessing Scenario Management

To access the Scenario Management page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click ▶ to expand Stress Testing.

3. Click **Scenario**. The Scenario Management page is displayed.
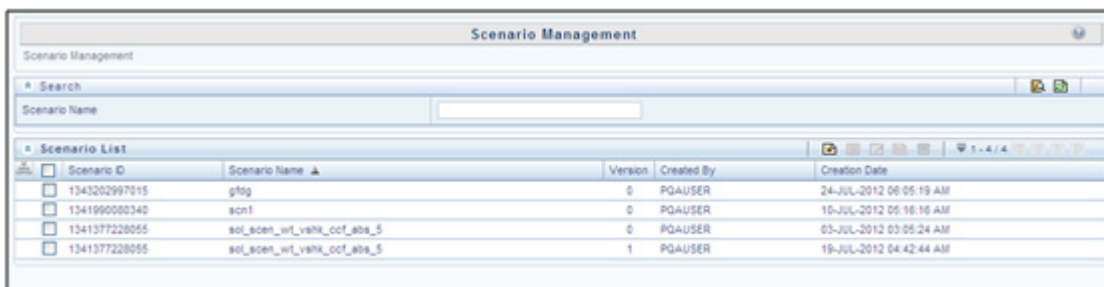
Figure 49. Scenario Management Page

Adding Stress Testing Scenario

You can create stress testing scenario by associating the appropriate Variable Shock Definitions. To add stress testing scenario definition, follow these steps:

1. Select ![icon] from the Scenario List tool bar. The Add button is disabled if you have selected any checkbox in the grid. The Scenario Definition Add window is displayed.
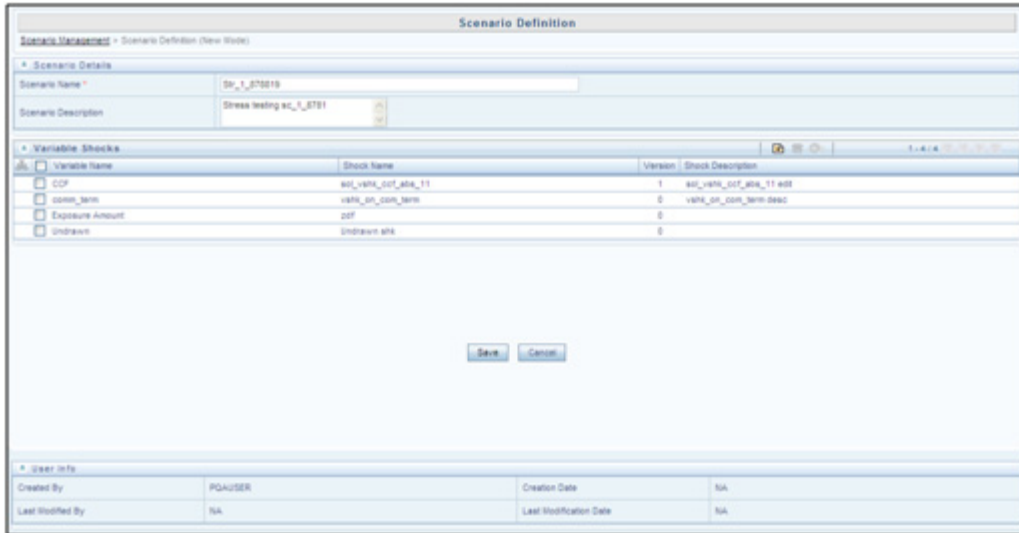
Figure 50. Scenario Definition Page

2. Enter the Scenario details as described in the following table:

Table 12. Scenario Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Scenario Name	Enter a name for the scenario. Ensure that there are no special characters like `, {,}," , ', ~, <,>, /, \, and multiple spaces.
Scenario Description	Enter the required description for the scenario.

To add Variable Shock to the scenario, follow these steps:

1. Select ![icon] from the Variable Shocks tool bar. The Add button is disabled if you have selected any checkbox in the grid. The Add Variable to Scenario window is displayed.

2. Select the required variable(s) by clicking on the adjacent checkbox in the Variable Shock List.
   You can also make use of Search option to search for a variable by specifying Variable Name or Shock Name or even by selecting the Variable Type from the drop-down list.

3. Click **OK**. The selected variables are listed in the Variable Shock grid in the Scenario Definition Add window.

   You can also remove a variable by selecting the adjacent checkbox and clicking ![icon] .

4. Click **Save**. The scenario details are saved and displayed in the Scenario Management window.

Viewing Stress Testing Scenario

You can view individual Scenario Definition details at any given point.

To view the existing Scenario Definition details, follow these steps:

1. Select the checkbox adjacent to the Scenario ID.

2. Click ▣ in the Scenario List tool bar.
   The Scenario Definition View window is displayed with the scenario details and the mapped variable shock.

Modifying Stress Testing Scenario

You can modify only the Scenario Description if the selected scenario is mapped to a Stress definition. If not, then all the filtering parameters can be modified except the Scenario Name. When you modify any of the Scenario parameters other than the Scenario Description, the details are uploaded as an incremented version without overwriting the existing scenario details.

To modify an existing Scenario Definition, follow these steps:

1. Select the checkbox adjacent to the Scenario ID whose details are to be updated.

2. Click 🖉 in the Scenario List tool bar. Edit button is disabled if you have selected multiple Scenario IDs. The Scenario Definition Edit window is displayed.

3. Edit the Scenario Definition details as required. For more information, refer *Add Stress Testing Scenario*.

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

Copying Stress Testing Scenario

The Copy Stress Testing Scenario facilitates you to quickly create a Scenario definition based on the existing details or by updating the values and remapping the required variables.

To copy an existing Stress Testing Scenario Definition, follow these steps:

1. Select the checkbox adjacent to the Scenario ID whose details are to be duplicated.

2. Click 🖫 in the Scenario List tool bar. The Copy button is disabled if you have selected multiple Scenario IDs.

3. In the Scenario Definition Copy window, you can:

- Create new scenario definition with existing variables. Specify a new Scenario Name and click **Save**.

- Create new scenario definition by updating the required variables. Specify a new Variable Name and remap the required variables. For more information, refer *Add Stress Testing Scenario*. Click **Save**.

The scenario definition details are displayed in the Scenario Management window.

Deleting Stress Testing Scenario

You can remove only those scenarios which are not mapped to any Stress Definitions and which are no longer required in the system, by deleting from the Scenario Management window.

To delete an existing Stress Testing Scenario Definition, follow these steps:

1. Select the checkbox adjacent to the Scenario ID whose details are to be removed.

2. Click [icon] in the Scenario List tool bar.

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

Searching Scenario Management

The Scenario Management page allows you to search specific model execution request.

To search for Scenario Management, follow these steps:

1. Navigate to the Scenario Management page.

2. Enter the Scenario Name.

3. Click [icon]. The filtered list is displayed.

Stress Definition

Stress refers to modeling a series of scenarios to analyze the impact of extreme market conditions and to measure the risk. This module enables you to create a stress definition. You can either specify a stand-alone execution of a Scenario, or map a Scenario(s) to a Baseline Run. The integrated process helps you to assess the impact of a Scenario across multiple areas. You can also view, modify, and delete stress definitions.

This section includes the following topics:

- Accessing Stress Definition
- Adding Stress Definition
- Executing Stress Definition
- Viewing Stress Definition
- Modifying Stress Definition
- Deleting Stress Definition
- Searching Stress Definition

Accessing Stress Definition

To access the Stress Definition page, follow these steps:

1. Select **Oracle Financial Services Enterprise Modeling** from the Application tab.

2. Click [icon] to expand Stress Testing.

3. Click **Stress Definition**. The Stress Definition page is displayed.

Figure 51. Stress Definition Page

Adding Stress Definition

Stress definition can be based on stress scenarios or business assumptions. A business assumption is essentially a process comprising of multiple rules which facilitate the computation of the impact of the assumption on base cash flows. Business assumptions are non-variable based.

Scenario based stress testing involves applying simultaneous shocks on multiple variables to assess the impact of scenario on a measure or a set of measures.

To add stress definition, follow these steps:

1. Select [+] from the Stress Definition List tool bar. The Stress Definition Add window is displayed. The Add button is disabled if you have selected any checkbox in the grid.
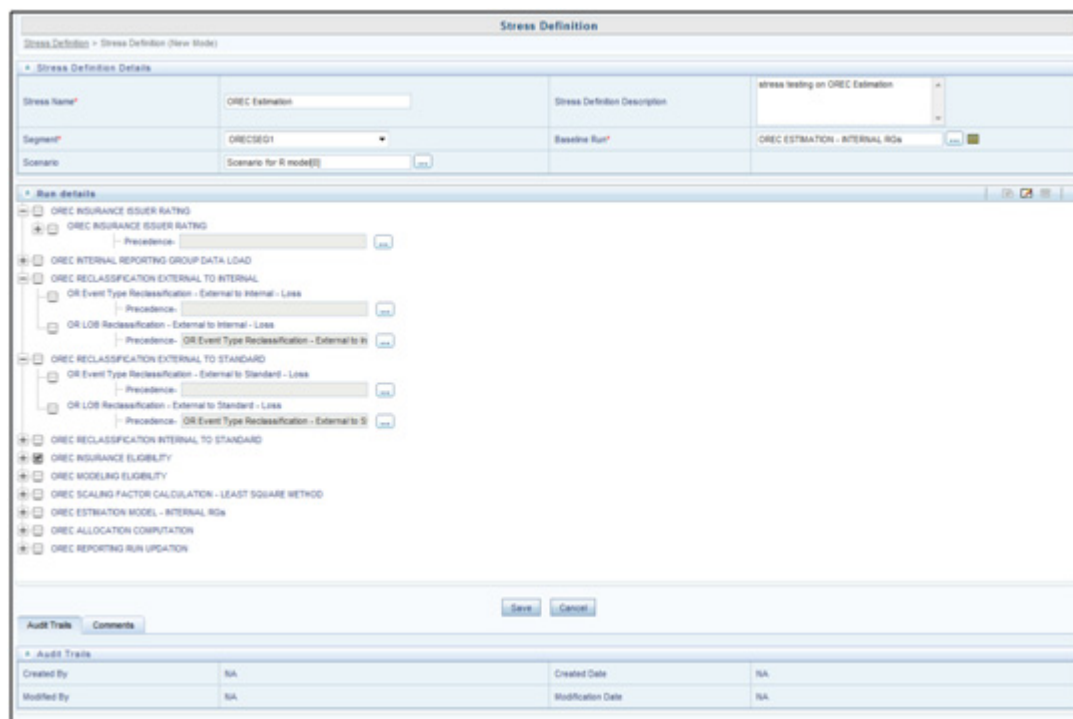
Figure 52. Stress Definition Page

2. Enter the Stress Definition details as described in the following table::

Table 13. Stress Definition - Field Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Stress Name	Enter a name for the stress definition. Ensure that there are no special characters like `, {,},", ', ~, <,>, /, \, and multiple spaces.
Stress Definition Description	Enter the required description for the stress definition.
Segment	Select the Segment from the drop down list. The list consists of the segments defined for the selected Production Infodom.

Table 13. Stress Definition - Field Descriptions

Field	Description
Base Line Run	1. Click [icon] and open the Baseline Run browser. The available Baseline Runs are listed in the Hierarchical pane. **Note**: Depending on the configuration selected in System Configuration & Identity Management tab > Configuration > Others tab as either PR2 or RRF, the corresponding Base Line Runs are displayed for selection. 2. Select the required Baseline Run based on which stress definition is to be created and click [icon]. 3. Click **OK**. 4. Click [icon] and view the selected baseline Run details.
Scenario	If the stress definition is based on variable shocks, select the scenario which you have defined with the required variable shocks. If it is non-variable based as in liquidity risk management, you need not select the scenario. 1. Click [icon] and open the Scenario browser. The available Scenarios are listed in the Hierarchical pane. 2. Select the required Scenario based on which stress definition is to be created and click [icon]. 3. Click **OK**.

In the Run Details grid, all the processes and tasks associated with the selected baseline run is displayed.

You can replace a process in the selected baseline run with another process. The process replacement can happen only for the processes which are directly placed under the run.

1. Select the checkbox adjacent to the process and click [icon] to replace it with another process. The Tasks Browser window displays all the processes which are defined in the selected segment and infodom.

2. Select the required process from the list and click [icon]. You cannot select a process which is already present in the baseline run.

You can modify or replace a task within a process with another task.

1. Select the checkbox adjacent to the task and click [icon]. The Tasks Browser window displays all the tasks which are defined in the selected segment and infodom.

2. Select the required process from the and click [icon].

Select a task and click [icon] to delete it.

Set the precedence in which the tasks need to be executed.

1. Click adjacent to the Precedence field of a task. The available scenarios/ tasks are listed in the Precedence Browser window.

2. Select the tasks in the order in which they need to be executed and click . You can change the order by clicking and .

3. Provide the Parameters for the task in the Parameters field.

Note: If the configuration is set to RRF, then the Parameters field value must be specified within double quotes.

Click **Save**. The stress definition details are saved and displayed in the Stress Definition window.

Executing Stress Definition

After creating stress definition, you need to create Manage Run Definition. Based on the Run Type (Single Request or Multiple Request), you need to execute the stress definition from Batch Execution window or Batch Group Execution window respectively. You can also monitor the status of stress definition execution.

To execute stress definition, follow these steps:

1. Go to **Rules Framework > Manage Run Execution**. The Manage Run Execution window is displayed.

2. Click from the List toolbar. The Manage Run Execution window is displayed.

3. Click in the Run field and select the Stress Definition that you want to execute from the Run Selector window. For more information, refer to the *Manage Run Definition* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

4. Click **Save**.

5. Execute the stress definition:

● If you have selected Type as Single Request in the Manage Run Execution window:

1. Go to **Operations> Batch Execution**. The Batch Execution window is displayed. For more information on how to execute a Batch, refer to the *Execute Batch* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

2. Click **Execute Batch**.

● If you have selected Type as Multiple Request in the Manage Run Execution window:

1. Go to **Operations> Batch Group Execution**. The Batch Execution window is displayed. For more information on how to execute a Batch Group, refer to the *Batch Group Execution* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

2. Click **Start**.

To monitor the stress execution:

● For Batch Execution:

1. Go to **Operations> Batch Monitor**. The Batch Monitor window is displayed.

2. Select the Stress Definition whose execution you want to monitor and select the Batch Run details. For more information, refer to the *Monitor Batch* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

3. Click to start monitoring.

● For Batch Group Execution:

1. Go to **Operations> Batch Group > Batch Group Monitor**. The Batch Group Monitor window is displayed.

2. Select the Batch Group and Run IDs from the drop-down lists. For more information, refer to the *Batch Group Monitor* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

3. Click **Refresh** to start monitoring

● To view log files:

1. Go to **Operations> Batch Processing Report.** The Batch Processing Report window is displayed.

2. Select the Information Date and Batch Status from the drop-down lists. For more information, refer to the *Batch Processing Reports* section in the *Oracle Financial Services Analytical Applications Infrastructure User Guide*.

Viewing Stress Definition

You can view individual Stress Definition details at any given point.

To view the existing Stress Definition details, follow these steps:

1. Select the checkbox adjacent to the Stress ID.

2. Click in the Stress Definition tool bar. The Stress Definition View window is displayed with the stress definition details along with the mapped scenarios.

3. Click in the Model Variable Shock Mapping toolbar. The Model Variable Shock Mapping window is displayed with the list of variable shocks associated with the selected scenario.

Modifying Stress Definition

You can modify only the Stress Description and Scenario for any selected Stress Definition.

To modify an existing Stress Definition, follow these steps:

1. Select the checkbox adjacent to the Stress ID whose details are to be updated.

2. Click in the Stress Definition List tool bar. The Edit button is disabled if you have selected multiple Stress IDs. The Stress Definition Edit window is displayed.

3. Edit the Stress Definition details as required. For more information, refer to *Add Stress Definition*.

   Note: While editing Stress Definition, you cannot unmap a scenario that is mapped to the stress definition. You can only change the version of the scenario.

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

Deleting Stress Definition

You can remove the stress definitions which are no longer required in the system, by deleting from the Stress Definition Summary window.

Note: Sress definitions, which use PR2 Base Line Run and has batch registered for processing, cannot be deleted. However, Stress definitions, which use RRF Base Line Run, can be deleted irrespective of whether it has batch registered for processin.

To delete a Stress Definition, follow these steps:

1. Select the checkbox adjacent to the Stress ID whose details are to be removed.

2. Click in the Stress Definition List tool bar.

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

Searching Stress Definition

The Stress Definition page allows you to search specific model execution request.

To search for Stress Definition, follow these steps:

1. Navigate to the Stress Definition page.

2. Enter the Stress Definition Name.

3. Click . The filtered list is displayed.

APPENDIX A

This section of the document consists of information related to intermediate actions that needs to be performed while completing a task. The procedures are common to all the sections and are referenced where ever required. You can refer to the following sections based on your need.

R Scripts and Oracle R Enterprise (ORE) Statistical Functions

Data usage when R script is used entirely

While defining a model scripted in R, user can select **Dataset** and **Variables** to assign data (table columns) to the R/ORE objects used within the script. Oracle Financial Services Enterprise Modeling application prepares data from the dataset, variables and other attributes like filters chosen for the model and will make the same available as user specified R objects/ frames.

No specific treatment is required in the script for using the data. User can just have an R data.frame object say 'GDP' and a variable (table.column) assigned to it and use 'GDP' as is in the script.

Data usage when ORE native implementation is employed - Oracle Financial Services Enterprise Modeling application - Specifics to be considered

The data which is prepared from the dataset and variables as explained in the previous section, will be available in this case as a named ORE ore.frame object ('OFSDATASET') when ORE native implementations are used (that is, when the **Is ORE implementation used?** checkbox is selected). Hence, the R objects(data) must always be accessed through the ore.frame object 'OFSDATASET'.

For instance, if two objects say 'x' and 'y' are used and dataset/ variables are chosen for 'x' and 'y' in the model definition, then the objects 'x' and 'y' should be accessed in the script as 'OFSDATASET$x' and 'OFSDATASET$y' respectively.

An illustration to explain the R and ORE cases is given in the following section:-

For a simple regression model which is entirely scripted in R, that uses a dataset and three variables (DependentVariable, IndependentVariable1 and IndependentVariable2), the user should not select the **Is ORE implementation used?** checkbox. In this case the R script is as follows:

```
art.mod<-lm(DependentVariable ~ IndependentVariable1 + IndependentVariable2)
art.summ<-summary(art.mod)
coef( art.summ )
art.summ[[ "r.squared" ]]

#do some line plots
new.x.datafrme = data.frame(x=seq(from=range(IndependentVariable1)[ 1 ],to=range(
IndependentVariable1)[2],length=length(DependentVariable)))
```

Here the variables chosen for the model are accessed directly in the script as IndependentVariable1, IndependentVariable2, and DependentVariable.

Whereas for a parallel ORE implementation of the same (a dataset and three variables DependentVariable, IndependentVariable1 and IndependentVariable2) using ORE statistical functionalities, user must check the **Is ORE implementation used?** checkbox and the script is as follows:

```
art.mod<-ore.lm(DependentVariable ~ IndependentVariable1 + IndependentVariable2,
data=OFSDATASET)
art.summ<-summary(art.mod)
coef( art.summ )
art.summ[[ "r.squared" ]]

#do some line plots
new.x.orefrme = data.frame(x=seq(from=range(OFSDATASET$IndependentVariable1 )[ 1
],to=range(OFSDATASET$IndependentVariable1)[2],length=
length(OFSDATASET$DependentVariable)))
#perform some operations on the new data
```

Here the variables chosen for the model are accessed from OFSDATASET as OFSDATASET$IndependentVariable1, OFSDATASET$IndependentVariable2, OFSDATASET$DependentVariable.

Data Handling

It is highly recommended that data required from the database should be pulled through the framework provided mechanism, that is, using dataset and variables and not with any explicit DB connections. This ensures proper security, authenticity, and auditing.

Auditing is enabled in the definition windows by introducing audit trials that captures and displays the user details and the date of creation/ modification, along with comments.

For instance, here is a sample script where data is fetched from the DB directly (not through framework). This way of accessing the database resident data in a model is not recommended.

```
con<-dbConnect(Oracle(),"userName","password")
qry<-"select EventLoss as Y, CardType as X1, AccBalance as X2, CustSalary as X3 from
CustTable where  Default= 'Y' "
res<- dbSendQuery(con, qry)
OperationalData<-fetch(res)
dbDisconnect(con)

#Model Logic
NewRegModel<-lm(Y~X1+X2+X3, data= OperationalData)
Plot(NewRegModel)
NewRegModel
```

Here is how a script for the same purpose can be created for accessing data via framework dataset and variables. Define 'EventLoss', 'CardType', 'AccBalance', and 'CustSalary' from the table 'CustTable' as variables in the Oracle Financial Services Enterprise Modeling application. while defining the model, select these variables and assign them

to R objects (the R names used within the script) say, 'IndepVariable1', 'IndepVariable2', 'IndepVariable3' and 'DepVar' from the Configure Inputs window.

```
#Assuming EventLoss, CardType, AccBalance, CustSalary from #CustTable are defined as
variables in a data set, and that #dataset is selected for variable assignments to the
respective R #variable names: Y, X1, X2, and X3.

NewRegModel<-lm(DepVar ~ IndepVariable1+ IndepVariable2+ IndepVariable3)

#The variables are made directly available to the R
#environment by the framework

Plot(NewRegModel)
NewRegModel
```

Logging

R processing log that captures script processing information, any warnings, errors or exceptions from the script, gets generated in the database server at '$ORACLE_HOME/dbs', since the R executable runs completely on the database server. Purging of the files is recommended at a regular basis.

Fitting and Prediction Using R Script

In order to support fitting and prediction using R script, the output objects can be stored in datastore using ore.save() during fitting and retrieve it using ore.load() during prediction.

ore.save() saves an R object or a list of R objects to the specified datastore in the current user's schema of the connected database.

ore.load() loads all of the R objects stored in a specified datastore from the current user's schema of the connected database.

For more information, refer to the *Oracle® R Enterprise User's Guide* (E36761-08).

The approach to use ore.save() and ore.load() is explained with the following example:

R scripting

Fitting:

```
fit<-lm(y~x)
ore.save(fit, name='regression_fit', overwrite=TRUE)
```

ore.save() will store the R object 'fit' in the datastore with the name 'regression_fit'. overwrite is a logical value specifying whether to overwrite the datastore if it already exists; the default is FALSE.

The overwrite flag is set to true to make sure that ore.save() does not fail. This is one way of doing it; otherwise the user's script should check for the existence of the datastore object.

Prediction:

```
ore.load('regression_fit')
predict<-predict(fit)
```

`ore.load()` is going to load all the R objects in the datastore 'regression_fit'.

From the datastore user should know the R object name to be used in the predict call.

ORE scripting

The approach is same as explained above, but the **Is ORE implementation used?** checkbox to be checked.

Fitting:

```
ore_fit<-ore.lm(y~x, data=OFSDATASET)
ore.save(ore_fit, name='ore_regression_fit', overwrite=TRUE)
```

Prediction:

```
ore.load('ore_regression_fit')
predict<-ore.predict(ore_fit,  newdata=OFSDATASET)
```

List of Prepackaged ORE Techniques

The table lists the prepackaged ORE Techniques.

Table 14. List of Prepackaged ORE Techniques

Button	Description
Regression Techniques	● R_Linear Regression ● R_GLM Logistic Regression ● R_Stepwise Regression
Clustering Techniques	R_K-means Clustering
Correlation Techniques	● R_Kendall Rank Correlation ● R_Pearson Correlation Coefficient ● R_Spearman Rank Correlation

List of NAG Techniques

The OFSAA Infrastructure aims to address the statistical computation needs of a Bank for its operations. The techniques that are supported including the regression models as well as Static Variable Transformations are as follows:

Table 15. List of NAG Techniques

Button	Description	
Business Models	Credit Risk	• CashFlow Model • Conditional Default Model • Credit Metrics Structural Model • Distribution Fitting based Future Value Model • Historical Loss Distribution Fitting Model • Historical Pool Average Default Rate Model • Merton Model • Time to Default Model • VAR Reader
	Market Risk	• EWMA Model • GARCH Model • Market Risk VaR Model
	Operational Risk	Loss Distribution Approach
Statistical Techniques	Analysis of Variance	Two Factor Without Replication
	Classification and Regression Trees	• Entropy • GINI
	Copula	• Cook-Johnson • Empirical • Gaussian • Gumbel - Hougaard • Student t
	Data Analysis	• Bivariate Analysis • Descriptive Statistics
	Factor Analysis	• Maximum Likelihood Extraction Method • Principal Component Extraction method
	Goodness of Fit Tests	• Anderson - Darling Test • Chi-Square Test • Kolmogrov-Smirnov Test

Table 15. List of NAG Techniques

Button	Description	
	Monte Carlo Simulation for Continuous Distributor	• Beta • Burr • Chi - Square • Empirical • Exponential • Gamma • Generalized Pareto • Gumbel • Log Gamma • Log Logistic • Log Normal • Log Weibull • Multivariate Normal • Pareto Distribution • Snedecor F • Student t • Truncated Burr • Truncated Exponential • Truncated Gamma • Truncated Gumbel • Truncated Log- Gamma • Truncated Log- Logistic • Truncated Log- Normal • Truncated Pareto • Truncated Uniform • Truncated Weibull • Uniform • Univariate Normal • Weibull
	Monte Carlo Simulation for Discrete Distributions	• Binomial • Negative Binomial • Poisson
	Multivariate Methods	• Discriminant Analysis • Hierarchical Clustering • k-means • k-means and Boundary Based Prediction

Table 15. List of NAG Techniques

Button	Description	
	Non-Parametric Tests	• Kruskal Wallis • Mann Whitney • Median Test • Run Test • Sign Test • Two sample KS Test • Wilcoxon Sign Rank
	Optimization techniques	• Linear Programming • Quadratic Programming
	Parametric Tests	• Two Sample t-Test
	Regression Techniques	• Generalized Linear Models - Gamma errors • Generalized Linear Models - Gaussian errors • Linear Regression • Linear Regression with Mixed Effects - ML • Linear Regression with Mixed Effects - REML • Logistic Regression • Monte Carlo Expectation Maximization • Poisson Regression • Stepwise Regression
	Simultaneous Equations	• Hermitian Linear Equations • Simultaneous Linear Equations with Choleskey Factorization • Simultaneous Linear Equations with LU Factorization
	Time Series	• ARIMA • Autocorrelation • Cross Correlation between two Time Series

Table 15. List of NAG Techniques

Button	Description	
	Transition Matrix	• Transition Matrix - EWMA • Transition Matrix - Linear Regression • Transition Matrix - Multi Factor • Transition Matrix - Stepwise Regression • Transition Matrix - Time Series
	Variance Covariance and Correlation Matrix	• Kendall Rank Correlation • Pearson Correlation Coefficient • Spearman Rank Correlation

Prediction Techniques

The table lists the Prediction Techniques used in Oracle Financial Services Enterprise Modeling Application

Table 16. List of Prediction Techniques

Button	Description
Regression Techniques	• Generalized Linear Mixed Models with Gamma Errors • Generalized Linear Mixed Models with Gaussian Errors • Linear Regression • Logistic Regression • Linear Regression with Mixed Effects - ML • Linear Regression with Mixed Effects - REML • Monte Carlo Expectation Maximization • Poisson Regression • Stepwise Regression
Clustering	• K-means and Boundary Based Prediction • Discriminant Analysis • Hierarchical Clustering
Classification and Regression Trees	GINI
Factor Analysis	Principal Component Extraction Method
Time Series	ARIMA

Transition Matrix

Transition Matrix is a statistical technique used across multiple applications in OFSAA. Transition Matrix is defined as a set of measures that quantify the probability of moving data from one state to another. Transition Probability defines the probability of transitioning data from one state to another over the time interval. The time interval and the horizon on which probabilities are estimated are derived from the User Input in the model definition interface.

Transition Matrix technique parameters are completely based on historical data. You can create multiple transition matrices and calibrated a set of data. For example institutions may calibrate different transition matrices for wholesale and retail exposures.

You can calibrate the parameters of a Transition Matrix technique by defining the required options in the Model Definition window. The Model Definition window within the Model Management section of Oracle Financial Services Enterprise Modeling application consists of the following five different types of transition matrix techniques. Click on the required section to view the details.

- Transition Matrix - EWMA
- Transition Matrix - Linear Regression
- Transition Matrix - Multi Factor
- Transition Matrix - Stepwise Regression
- Transition Matrix - Time Series

Transition Matrix - EWMA

In EWMA (Exponentially Weighted Moving Average) method the Decay factor is the mandatory parameter required as user input. There is no calibration associated with the EWMA Model. The framework predicts transition probabilities based on user input of decay factor and time interval.

Transition Matrix - Linear Regression

Linear Regression method is used to establish relationship between Explanatory Variable(s) with a Scalar Variable. Linear Regression technique uses Linear functions for data modeling which can also estimate the unknown model parameters.

Regression scenario: Suppose you want to learn more about the purchasing behavior of customers of different ages. You can build a model to predict the ages of customers as a function of various demographic characteristics and shopping patterns. The prediction can then be done using a regression algorithm.

Transition Matrix - Multi Factor

The Multi Factor technique of model calibration refers to the process of estimating measures such as the Average Z-Score (difference between Standard Deviation from Mean), Average Transition, and so on. The historical data is transferred to the Sandbox Information Domain for model calibration. The following are the steps required for the estimation of the above measures:

Calculate the historical average transitions.

Calculate the z-scores.

Subtract the average z-score from each historical time series.

In addition to calculating realized z-scores, the calibration process also computes the average z-score for each row of historical z-score time series and a parameter Alpha which is the standard deviation of average z-score shifts.

Transition Matrix - Stepwise Regression

In stepwise regression, the independent variables are selected automatically and the model is constructed in an iterative fashion. The three approaches for achieving the stepwise regression are:

- **Forward Selection**: In which the inclusion of a variable completely depends on its statistical significance.

- **Backward Elimination**: In which the included candidate variables are tested for xsstatistical significance and are eliminated as they fail.

- **Combination**: This includes both Forward Selection and Backward Elimination process in which the variables are tested at different steps (check points) to determine whether they need to be carried along or not.

Transition Matrix - Time Series

Time series regression includes a series of data points considered at consecutive time intervals. This type of analysis is done to validate the time series data and to extract the statistics of the data under consideration. This model can also predict the future values using the previously observed values.

APPENDIX B

User Groups

Table 17. User Groups

User Group Name	User Group Description
Modeling Administrator	User mapped to this group will have access to all the menu items for Enterprise modeling and will be have authorization rights for sandbox population, model deployment and modeling technique authorization.
Modeler Group	User mapped to this group will have access to all the menu items for Enterprise modeling but will not have authorization rights for sandbox population, model deployment and modeling technique authorization.

User Group - Role Mapping

Table 18. User Group to Role Mapping

Group Name	Role Name
Modeler	Alias Access
Modeler	Alias Read Only
Modeler	Alias Write
Modeler	Atomic excel upload write
Modeler	BMM Hierarchy Access
Modeler	BMM Hierarchy Read Only
Modeler	BMM Hierarchy Write
Modeler	BMM Processor Access
Modeler	BMM Processor Read Only
Modeler	BMM Processor Write
Modeler	Batch Access
Modeler	Batch Read Only
Modeler	Batch Write
Modeler	Config excel advanced
Modeler	DEFQ Manager
Modeler	DEFQ write
Modeler	DI Write
Modeler	DMM Write
Modeler	DQ Access
Modeler	DQ Advanced
Modeler	DQ Read
Modeler	DQ Write
Modeler	DT Write
Modeler	Dataset Access
Modeler	Dataset Read Only
Modeler	Dataset Write
Modeler	Derived Entity Access
Modeler	Derived Entity Read Only
Modeler	Derived Entity Write
Modeler	Dimension Access
Modeler	Dimension Read Only
Modeler	Dimension Write
Modeler	ETL Analyst
Modeler	Essbase Cube Access
Modeler	Essbase Cube Read Only
Modeler	Essbase Cube Write
Modeler	MDB Write
Modeler	MF Access

Table 18. User Group to Role Mapping

Group Name	Role Name
Modeler	MF Advanced
Modeler	MF Read
Modeler	MF Write
Modeler	Manage Run Access
Modeler	Manage Run Read Only
Modeler	Manage Run Write
Modeler	Measure Access
Modeler	Measure Read Only
Modeler	Measure Write
Modeler	Obj Migration Access
Modeler	Obj Migration Read
Modeler	ObjectAdmin advanced
Modeler	Oracle Cube Access
Modeler	Oracle Cube Read Only
Modeler	Oracle Cube Write
Modeler	Process Access
Modeler	Process Read Only
Modeler	Process Write
Modeler	Rule Access
Modeler	Rule Read Only
Modeler	Rule Write
Modeler	Run Access
Modeler	Run Read Only
Modeler	Run Write
Modeler	STF Access
Modeler	STF Advanced
Modeler	STF Read
Modeler	STF Write
Modeling Administrator	Alias Access
Modeling Administrator	Alias Authorize
Modeling Administrator	Alias Read Only
Modeling Administrator	Alias Write
Modeling Administrator	Atomic excel advanced
Modeling Administrator	Atomic excel upload write
Modeling Administrator	BMM Hierarchy Access
Modeling Administrator	BMM Hierarchy Authorize
Modeling Administrator	BMM Hierarchy Read Only
Modeling Administrator	BMM Hierarchy Write
Modeling Administrator	BMM Processor Access
Modeling Administrator	BMM Processor Authorize

Table 18. User Group to Role Mapping

Group Name	Role Name
Modeling Administrator	BMM Processor Read Only
Modeling Administrator	BMM Processor Write
Modeling Administrator	Batch Access
Modeling Administrator	Batch Advanced
Modeling Administrator	Batch Authorize
Modeling Administrator	Batch Phantom
Modeling Administrator	Batch Read Only
Modeling Administrator	Batch Write
Modeling Administrator	Config excel advanced
Modeling Administrator	DEFQ Manager
Modeling Administrator	DEFQ access
Modeling Administrator	DEFQ advanced
Modeling Administrator	DEFQ authorize
Modeling Administrator	DI Write
Modeling Administrator	DMM Write
Modeling Administrator	DQ Access
Modeling Administrator	DQ Advanced
Modeling Administrator	DQ Authorize
Modeling Administrator	DQ Read
Modeling Administrator	DQ Write
Modeling Administrator	DT Write
Modeling Administrator	Dataset Access
Modeling Administrator	Dataset Authorize
Modeling Administrator	Dataset Read Only
Modeling Administrator	Dataset Write
Modeling Administrator	Derived Entity Access
Modeling Administrator	Derived Entity Authorize
Modeling Administrator	Derived Entity Read Only
Modeling Administrator	Derived Entity Write
Modeling Administrator	Dimension Access
Modeling Administrator	Dimension Authorize
Modeling Administrator	Dimension Read Only
Modeling Administrator	Dimension Write
Modeling Administrator	ETL Analyst
Modeling Administrator	Essbase Cube Access
Modeling Administrator	Essbase Cube Authorize
Modeling Administrator	Essbase Cube Read Only
Modeling Administrator	Essbase Cube Write
Modeling Administrator	MDB Write
Modeling Administrator	MF Access

Table 18. User Group to Role Mapping

Group Name	Role Name
Modeling Administrator	MF Advanced
Modeling Administrator	MF Authorize
Modeling Administrator	MF Read
Modeling Administrator	MF Write
Modeling Administrator	Manage Run Access
Modeling Administrator	Manage Run Read Only
Modeling Administrator	Manage Run Write
Modeling Administrator	Measure Access
Modeling Administrator	Measure Authorize
Modeling Administrator	Measure Read Only
Modeling Administrator	Measure Write
Modeling Administrator	Obj Migration Access
Modeling Administrator	Obj Migration Advanced
Modeling Administrator	Obj Migration Read
Modeling Administrator	Obj Migration Write
Modeling Administrator	ObjectAdmin advanced
Modeling Administrator	Oracle Cube Access
Modeling Administrator	Oracle Cube Authorize
Modeling Administrator	Oracle Cube Read Only
Modeling Administrator	Oracle Cube Write
Modeling Administrator	PR2 Administrator
Modeling Administrator	Process Access
Modeling Administrator	Process Advanced
Modeling Administrator	Process Authorize
Modeling Administrator	Process Read Only
Modeling Administrator	Process Write
Modeling Administrator	Publish Metadata
Modeling Administrator	Rule Access
Modeling Administrator	Rule Advanced
Modeling Administrator	Rule Authorize
Modeling Administrator	Rule Read Only
Modeling Administrator	Rule Write
Modeling Administrator	Run Access
Modeling Administrator	Run Advanced
Modeling Administrator	Run Authorize
Modeling Administrator	Run Read Only
Modeling Administrator	Run Write
Modeling Administrator	STF Access
Modeling Administrator	STF Advanced

Table 18. User Group to Role Mapping

Group Name	Role Name
Modeling Administrator	STF Read
Modeling Administrator	STF Write

APPENDIX C

This section consists of information related to Command Line Utility. You can refer to the following sections based on your need.

Command Line Utility

There are three approaches available for object migration namely Command Line Object Migration Utility, Object Migration from Administration>Utilities, and Metadata Archive/Restore available in Data Module Management. You can choose an approach based on whether the objects you intend to migrate are supported in that approach. Command Line Object Migration is the common integrated approach and is recommended over other methods. The Data Source should be created for the Sandbox Infodom in the target setup as prerequisite for the migration of R / ORE based models.

Offline Migration: XML tag and values need to be modified in **OBJECTMIGRATION.XML**

In ObjectMigration.xml, the object code for ETL should be app name followed by tilde, source name followed by tilde and then extract name.

```
<OBJECT Code="AppName~SourceName~ExtractName" Type="122" />
```

For information on Migrating Objects, refer *Command Line Utility* section in *Oracle Financial Services Analytical Infrastructure Applications User Guide*.

Prerequisites

For Modeling Framework and Stress Testing objects:

- For Production infodom, it is assumed that the data model upload is already done.

- Implicit migration of dependent objects is not supported. You need to explicitly migrate the dependent objects to the target environment, prior to migrating the MF or ST objects.

  For example, in case of migrating a Model, the dependent objects such as sandbox, datasets, variables, technique (if any) and hierarchies should be migrated first. In case of Stress Definition, you need to migrate the dependent Stress Run and corresponding Process, along with other dependent objects such as Baseline Run and Scenario. For Sandbox, the dependent objects are datasets and hierarchies.

- Models can be migrated only from one sandbox infodom to another sandbox, not to Production infodom.

- The availability of the application with same APP_ID in source and target environment is a prerequisite for Sandbox Migration.

- After migrating logical sandbox between environments, the user group mapping has to be done manually.

- If the "Data model upload required" checkbox is selected for the sandbox definition, the infodom code provided in SANDBOXINFODOM attribute should be a clean infodom where no model upload is done.

- For the third party techniques, the jar files/executables created will not be migrated. You have to copy those manually.

- Ensure that the technique name in the source system and the target system are the same, that is additional R_ should not be appended to the technique name, when migrtaed to the target system.

- Technique migration is permissible only between Infodoms and cannot be performed between environments.

- Data Source creation for Sandbox Infodom is a prerequisite for migrating R / ORE based models.

  Note: Migration of models from logical sandbox to regular sandbox is not supported. That is, it is supported from one logical sandbox to another logical sandbox or from one schema based sandbox to another schema based sandbox.

Objects Supported for Command Line Migration

Table 19. Objects Supported for Command Line Migration

Object Name	Object Type ID	Support for Wildcard Select ALL Option	Support for Implicit Dependency [1]	Object Code
SANDBOX [2]	1300	No	No	System generated code
VARIABLE	1301	Yes	No	System generated code
TECHNIQUE	1302	No	No	System generated code
VARIABLE SHOCK	1303	No	No	System generated code with '_' and Version number
SCENARIO	1304	No	No	System generated code with '_' and Version number
MODEL	1305	Yes	No	System generated code with '_' and Version number
STRESS	1306	No	No	System generated code

Command Line Utility for Compiling Non-Self Executable Libraries

The Non-Self executable Third Party techniques defined must be compiled prior to executing the models based on such techniques. For compiling such techniques, a command line utility called TechniqueBuilder.sh is provided. For using this utility, the C/C++ compiler must be installed on OFSAAI server. Libraries of type ('.so' and '.a') for C/C++ and ('.class' and '.jar') for JAVA are supported for now.

The `TechniqueBuilder.sh` utility resides under `$FIC_DB_HOME/bin` area.

Prerequisites

- C/ C++ compiler (g++ for Linux, xlC_r for AIX and CC for Solaris) must be installed on OFSAAI server. In case of Multitier setup, you must install on the OFSAAI FIC DB tier. You must also set the compiler path accordingly in the PATH variable present in `.profile`.

- If JAVA libraries are part of Non-Self Executable Third party Techniques, then ensure that JAVA_HOME variable is set and pointing to the JDK Installed Directory in `.profile`. Else this step is not required.

- All the external Non-Self Executable libraries must be placed under the `$FIC_DB_HOME/lib/MF_EXTERNAL_LIBS/lib` folder.

- If the C/C++ library contains any Include Header files, then those files should be placed under `$FIC_DB_HOME/lib/MF_EXTERNAL_LIBS/`include folder. Else this step is not required.

To run this utility from the console:

1. Navigate to *$FIC_DB_HOME/bin* of OFSAAI FIC DB tier.

2. Execute the following command:
   `./TechniqueBuilder.sh`

3. This command compiles all the techniques that are based on Non Self Executable Third Party algorithms. Once compilation is successful, the 'libTPPlugin.so' library will be updated in the `$FIC_DB_HOME/lib` path.

Oracle Financial Services Enterprise Modeling application

8.0.1.0.0 User Guide

Oracle Corporation

World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065

U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com/us/industries/financial-services/

ORACLE®