

Oracle Financial Services Model Management and Governance User Guide



Release 8.1.3.3.0

G51848-01

February 2026

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Copyright © 2014, 2025, Oracle and/or its affiliates.

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

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

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

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

Contents

1 About This Guide

2 Getting Started

About this Guide	1
About Oracle Financial Services Model Management and Governance	1
Workflow	1
Components	2
Workspaces	2
Datasets	3
Model Pipelines	3
Model Actions	4
Graphs	4
Scheduler Service	4
Audit Trail	4
Data Studio Options	4
Object Migration	5
Model Libraries	7
Model Techniques	7
Model Catalog	7
Data Management	7
Managing User Groups	8
User Groups	8
Role Mapping	9
Function-Role Matrix for Conda CRUD Operations	11
Sign in	11

3 Execute the Production Workspace Models from Analytical Applications Infrastructure

4 User Interface

Home Page Components	2
----------------------	---

Home	2
Health	3
Notification	4
Profile	5
Mega Menu Components	6
Modeling	6
Orchestration	6
More	7
Using the Application	7

5 Access MMG from Analytical Applications Infrastructure

6 Workspaces

Create a Workspace	4
Configure Basic Details	5
Configure Workspace Schema	7
Configure Data Sourcing	8
Configure Metadata Sourcing	9
Validate Workspace	10
Display Summary	11
Populate a Workspace	11
Retry a Failed Workspace	17
Delete a Workspace	17
Download a Workspace	18
Managing Data Sources	19
Details Screen	19
Edit a Workspace	28

7 File Manager

8 Administration

Data Stores	2
Adding a Data Store	3
Viewing the Data Store Details	8
Conda Environments	8
Register a Conda Environment	10
Create a Conda Environment	12
Edit a Conda Environment	13

View a Conda Environment	14
Deregister a Conda Environment	14
Inspect a Conda Environment	15
Remove Environment	16
Clone Environment	16
Export Environment	17
Edit a Conda Environment Description	17
Add or Remove Environment Tags	17
Search All Packages	18
View Package Properties	18
Update Package	18
Remove Package	19
Clone Environment	19
Export Environment	20
Modify Conda Settings	20
View Conda Properties	21
Using the .condarc Conda Configuration File	21
API Registry	24
Configure an API	25
View an API Registry	28
Edit an API Registry	28
Delete an API Registry	29
Kafka Topics	30
Create a Kafka Cluster	31
Delete a Kafka Cluster	32
Register a Kafka Topic	32
View a Kafka Topic	33
Edit a Kafka Topic	34
Delete a Kafka Topic	35
OFSAA Environment	36
Data Studio Options	37
Configure Interpreters	37
Manage Tasks	38
Manage Tasks	38
Manage Credentials	38
Configure Templates	39
Audit Trail	39
Identity Management	40

9 Interpreter Configuration and Connectivity

Configure Interpreters	2
------------------------	---

python Interpreter	8
Configure a Python Interpreter	9
Change Version in the Python Interpreter	9
Create Zeppelin Interpreter Output Limit in the Python Interpreter	10
Setting the Maximum Number of Results in the Python Interpreter	15
jdbc Interpreter	19
Configure a jdbc Interpreter	20
Link Wallet Credentials to jdbc Interpreter	24
md Interpreter	24
PGX Interpreter	25
Spark Interpreter	26
Spark Interpreter in Local Mode	26
Enable Additional Spark Interpreter	29
Pyspark Interpreter	29
Prerequisites	29
Configuration	29
Use Python Virtual Environments with PySpark	30
Configure Pyspark Interpreter	32
Enable Additional PySpark Interpreter	32
R Interpreter	32
Configuration	34
Create a Credential	37
Link Credentials	40
Create an Interpreter Group	42
Clone an Interpreter	42

10 Data Management

Data Modeling	1
Data Population	7

11 Modeling

Dataset	1
Access the Dataset Summary	1
Create a Dataset	5
Creating a Dataset	5
View a Dataset	16
Overview	17
Data	18
Profile Summary	18
Model Monitor	23

Data Drift Summary	26
Fairness	29
Data Lineage	30
Audit History	31
Create a Cache Snapshot	32
Edit a Dataset	33
Delete a Dataset	33
Python functions for accessing Dataset from Model Pipelines/Notebooks	33
List tags of Datasets Cached from UI	33
Delete Cached Dataset	34
List all Datasets with Metadata saved from UI	34
Fetch Dataset in Notebook	34
Cache User's Dataframe from Notebook	35
Fetching Data Frame Cached from Notebook	35
List Tags of all Data Frames Cached from Notebook	35
Model Libraries	35
Edit a Model Library	38
Delete a Model Library	39
Model Technique	40
Edit a Model Technique	41
Delete a Model Technique	42
Model Catalog	42
Model Objective	42
Add a Model Objective	43
View a Model Objective	45
Delete a Model Objective	54
Pipelines	54
Prerequisites	54
Access Model Pipelines	54
Manage Models	61
Create Objective (Folders)	62
Create Draft Models	63
Create Seeded Models	68
Pipeline Designer	69
Pipeline Canvas	70
Parameter Set	101
Import/Export Parameter Sets using Utility	102
Scheduler Configuration UI	103
Publish Data Studio	104
Scope Detail	105
View Model Details	105
Model Governance	106

Request Model Acceptance	108
Review Models and Move to Approve or Reject	109
Approve Models and Promote to Production	109
Deploy Models in Production and Make it a Global Champion	110
Import a Model Data into a New Model	110
Git Functionalities	112
Execute Models using Scheduler Service	114
Define a Task	114
When Component is Model	116
When Component is Populate Workspace	117
SS \$RUNSK support for processing Delta Data by Data Pipeline and Match Rules	118
Export/Import Models via Utility	118
View Models	120
Edit Models	120
Delete Objectives and Draft Models	120
Programmatic Dynamic Forms and Fixed Dynamic Forms	121
Session Management in Model Pipeline	122
Creating a Session	123
Clearing a Session	124
Reinitializing a Session	125
Switching Between Notebook and Pipeline	126
Graphs	127
Accessing the Graph Summary Page	128
Adding a Graph Pipeline	129
Create a Node	130
Create an Edge	135
Zoom In and Zoom Out	141
Edit a Graph Pipeline	141
Delete a Graph Pipeline	142
Edit a Node	142
Delete a Node	143
Schedule Details	144
Audit History	149
Execution History	149
Analyses	150

12 Orchestration

Scheduler Service	1
API Endpoints for Export/Import Scheduler Objects	1
Batch Import Export Utility	3
Scheduler Service Dashboard	4

Configure Batch and Batch Groups	5
Create a Batch	5
Manage Batch	6
Create a Batch Group	7
Manage Batch Group	8
Define Batch	8
Define Tasks	10
Configure Tasks	12
Add Task to a Batch or Batch Group	12
Manage Tasks	14
Define Task Precedence	15
Exclude or Include Tasks	16
Schedule Batch or Batch Group	16
Execute a Batch/Batch Group	16
Schedule a Batch/Batch Group	17
Edit Dynamic Parameters	23
Monitor Batch or Batch Groups	24
Batch Group	25
Scheduler Configuration	26
Batch to User Configuration	26
General Configuration	27
External Interface Component Scheduler Service	29
Batch Utilities	41

13 More

Object Migration	1
Export Objects	2
Search Objects to Export	3
Create Migration Export Object Definitions	3
View Migration Objects to be Exported	5
Export Migration Objects	7
Edit Exported Migration Definitions	7
Delete Exported Migration Definitions	7
Import Objects	8
Search Objects to Import	8
Create Import Object Definitions for Migrations	9
View Migration Objects to be Imported	12
Import Migration Objects	13
Edit Imported Migration Definitions	14
Delete Imported Migration Object Definitions	14
Prerequisites to Migrate Objects	14

Migration Object Types	15
Schedule	15
Batch	15
Batch Group	15
Pipeline	15
Threshold	15
Job	15
PMF_Process	16
Roles	16
Groups	16
Model Actions	16
Audit Trail	17

A APIs for Model Pipelines

B Public APIs for Scheduler Operations

1

About This Guide

This guide has information about configuring and publishing workspaces using the OFS Model Management and Governance application. Use it for information about workspace and model management or help with the processes in the application.

Audience

This document is intended for the system administrators and users configuring the workspace and models in OFS MMG.

See these Oracle resources:

- [MMG Documentation Library](#)
- [How Oracle Financial Services Solves Real Business Problems](#)

2

Getting Started

This section gives an introduction to OFS Model Management and Governance and the requirements to use the application.

About this Guide

This guide has information about configuring and publishing workspaces using the OFS Model Management and Governance (MMG) application. The further sections in this guide provide information for the creation and deployment of models into production. Use it for information about workspace and model management or help with the processes in the application.

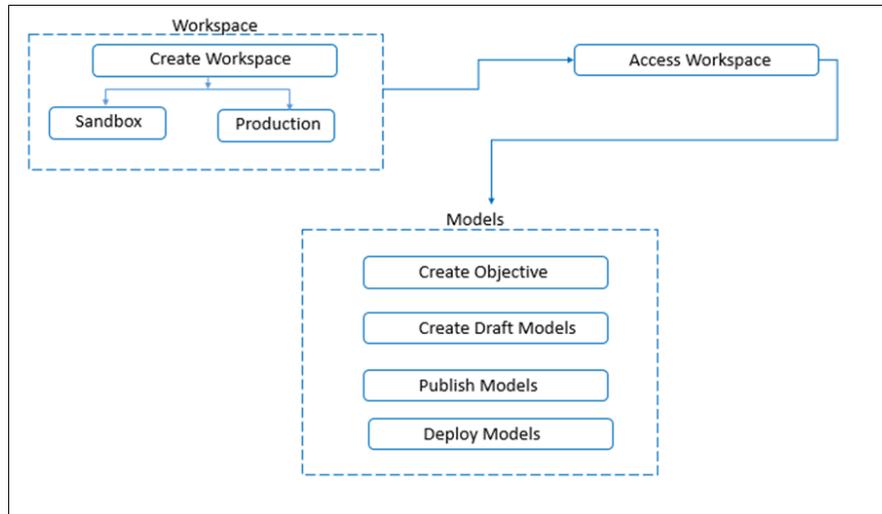
About Oracle Financial Services Model Management and Governance

Financial Institutions require models that work on traditional statistical techniques, modern machine-learning methods, computational and simulation models. Oracle Financial Services Model Management and Governance leverage the Data Studio environment to develop, deploy, and manage models at the enterprise level.

The OFS Model Management and Governance application enables institutions to implement their IT policies while providing flexibility and freedom that Data Scientists and Statistical Modelers desire. OFS MMG's design facilitates financial institutions to manage external regulatory and internal governance policies by building testing models in a workspace environment. A workspace is provisioned and authorized for use (usually by an administrator) before making it available to modelers. Administrative users grant analysts and modelers access to workspaces along with a subset of production data to build models. Validated and approved models can then be promoted from workspaces to the enterprise model repository. Models in the repository can then be woven into analytical application flows crafted by mixing data management tasks, model execution, and deterministic business logic.

Workflow

The workflow involves the creation of workspaces and the creation of Models mapped to the Workspaces. Models are then configured as training models that you can use to perform model visualizations and test for the outcomes. You can then publish a model into production and make it available to users after you have determined that the models and the parameters used to construct the models meet the requirements of your business logic.

Figure 2-1 Oracle Financial Services Model Management and Governance Workflow

Components

The following are the components of Oracle Financial Services Model Management and Governance application:

- [Workspaces](#)
- [Datasets](#)
- [Model Pipelines](#)
- [Model Actions](#)
- [Graphs](#)
- [Scheduler Service](#)
- [Audit Trail](#)
- [Data Studio Options](#)
- [Object Migration](#)
- [Model Libraries](#)
- [Model Techniques](#)
- [Model Catalog](#)
- [Data Management](#)

Workspaces

Workspace Management is where the workspace Administrators define datasets and make them available to modelers in Oracle Financial Services Model Management and Governance. It is an environment where the data is prepared for modelers to use in their modeling creation activity.

Workspaces are provisioned with the data required for modeling by the Administrators, who configure workspaces with subsets of production data. The data in the workspace is made available to modelers using datasets, who then build models. In effect, the data is ready for the modelers, and they do not have to undergo the arduous task of accessing and querying the database. See the [Using Workspaces](#) section for information on how to use Workspace Management in the application.

Datasets

Datasets allow for the creation of a dataframe, a data structure that captures the logic that organizes data into a 2-dimensional table of rows and columns. Datasets allow for the reuse of these dataframes across models, as well as the features derived from the data source(s).

The Dataset allows you to manage the entire operation related to data set. You can perform the following two things using the Dataset window:

- Define a metadata on how you want to create a dataset.
- In addition, a mechanism where you can take a snapshot of real time data and store it. So, it can be used later in the pipeline.

Model Pipelines

Modeling refers to the process of designing a prototype based on a structured data model, for statistical analysis and to simulate real events and processes. Models in a workspace can be created or modified by a user with access to the workspace. Model versions are preserved in the workspace along with execution and output histories. Once a model has been validated in the workspace and considered fit for use, modelers can request to push the model into the production environment.

Note

In Interactive Mode execution, if para 1 and para 2 are executed in parallel, the parameters of para 2 are not listed even if it is present in the same stage. In Interactive Execution of models, parameter names should be different if it is being used in two models and require two values for execution.

Note

Comments section added in Model Pipeline Builder. Comments textbox is available for users to comment and user can view these comments in the Audit Panel in pipeline builder.

Note

The Add new pipeline field is added to set limits on memory usage by notebook session. When a user executes a pipeline, the memory limit will be applied. Ideally, the user should not consume the entire memory available. However, if they exceed the maximum memory limit, they will get a memory error.

Model Actions

The Model Actions window allows you to view the list of models for which the Workflow action has been taken which requires review or approve across the workspace.

Graphs

Unlike traditional relational database management systems, the Graph Pipeline feature allows you to view the data relationships in a graphical format.

This feature uses the latest technology to harness the power of Graph Analytics to give Financial Institutions the ability to monitor the data financial institutions effectively. The data is organized as nodes, relationships, and properties (property data is stored on the nodes or relationships). The results of analytics algorithms are stored as transient properties of nodes and edges in the Graph.

Scheduler Service

The Scheduler Service is a service in the Infrastructure system that automates behind-the-scenes work that is necessary to sustain various enterprise applications and functionalities. This automation helps the applications to control unattended background jobs program execution.

The Scheduler Service contains a graphical user interface and a single point of control for the definition and monitoring of background executions.

Scheduler Configuration/Admin UI

Audit Trail

The Audit Trail window provides the complete details of model. This shows the information such as, when Model was created, who created the Model, workflow of Model, for example, when this Model became champion or deployed, and so on.

Data Studio Options

The application's Notebook Server has the following configurable options:

- Interpreters
- Tasks
- Permissions
- Credentials
- Templates

To access the Data Studio Options page:

1. Click **Launch Workspace** to launch the workspace to display the **Dashboard** window with the application configuration and model creation menu.
2. Hover your mouse over the Data Studio Options widget . The following options are available:
 - Interpreters
 - Tasks

- Permissions
- Credentials
- Templates

Note

DS 23.4.10 has introduced 'Select-Multiple' forms in Data Studio. The same should now be handled in MMG (parameters sets, dashboard, execute pipeline drawer, simulations, and so on).

Object Migration

Object Migration is the process of migrating or moving objects between environments.

You may want to migrate objects for several reasons such as managing global deployments on multiple environments or creating multiple environments so that you can separate the development, testing, and production processes.

Object Migration Rest Endpoints to support Migration of Object available in MMG

Table 2-1 Object Migration Rest Endpoints to support Migration of Object available in MMG

Sr No	Object Migration API	Technical Standards to Follow	Method	Endpoint	Sample Request	Sample Response
1	Save Export Definition	Headers: ofs_workspace_id ofs_remote_user_locale (default set as en-US)	POST	/v1/public/migrate/export/outline	{ "migrationCode": "testexp42", "exportObjects": [{"objectCode": "TEST_population", "objectType": "BATCH" }] }	output success-> { "response": { "message": "Migration outline definition saved successfully." } } output failed -> { "migrationCode": "testexp42", "errorMessage": "objectCode/ Type missing", "statusCode": "INVALID_INPUT", "status": "FAILED" }


```
(default set as aaaiuser) locale (default set as en-US) output success-> { "response": \
{ "message": "Migration outline definition saved successfully." } } output failed -> \
{ "migrationCode": "testexp42", "errorMessage": "objectCode/Type missing", "statusCode":
"INVALID_INPUT", "status": "FAILED" }
```

Invoke Export

```
eg:[https://<mmg_ui_gateway_url/ui_url>/<contextname>|https://host/]/v1/public/migrate/
export/{migrationCode} output-> { "response": \{ "message": "Object Migration Triggered
Successfully" } } headers:- ofs_workspace_id (default set as TEST) ofs_remote_user (default
set as aaaiuser) locale (default set as en-US)
```

Save Import Definition

```
eg:[https://<mmg_ui_gateway_url/ui_url>/<contextname>|https://host/]/v1/public/migrate/
import/outline input Json:- \{ "migrationCode":"imp1_exp41",
"fileName":"TEST_TESTEXP41_06082024_680336", "code":"imp1_exp41" } headers:-
ofs_workspace_id (default set as TEST) ofs_remote_user (default set as aaaiuser) locale
(default set as en-US) output success-> { "response": \{ "message": "Migration outline
definiton saved successfully." } }
```

```
}
```

Invoke Import

```
eg:[https://<mmg_ui_gateway_url/ui_url>/<contextname>|https://host/]<context>/v1/public/
migrate/import/{migrationCode} headers:- ofs_workspace_id (default set as TEST)
ofs_remote_user (default set as aaaiuser) locale (default set as en-US) output success-
>output-> { "response": \{ "message": "Object Migration Triggered Successfully" } }
```

```
}
```

Model Libraries

The Model Library information is used to bind a particular technique and its details to one unique Model Library. The Starting point for the Model builder is to register a model library with MMG application after it is properly set up in the python environment to be used.

Model Techniques

Model Technique is the algorithm/technique used to create python model using the library/ package which was created in the Model Library Screen. It is the actual information captured in the MMG application that helps in training the model (Upload and Build).

Model Catalog

The Model Catalog page allows you to add and manage the Model Technique, Model Library, and Model Objectives. You can either import the models from external sources or create, train and label it as champion model.

Data Management

The Data Management module allows users to efficiently manage data stores within a workspace. It provides key features for importing both data models and data from Source Schemas. With features like Data Modelling and Data Population, users can replicate database structures, create data ingestion jobs, and schedule or manually run those jobs to bring in new

data. This module simplifies schema integration, supports data migration, and ensures consistent data flow across environments.

Managing User Groups

Users must be mapped to User Groups that are mapped to access OFS MMG. The following subsections provide information about the user groups and roles required in addition to the information about configuring the user groups.

User Groups

The following table gives details about the User Groups in the Oracle Financial Services Model Management and Governance application.

Table 2-2 User Groups

User Group Name	User Group Description
IDNTYADMN	Identity Administrator group
IDNTYAUTH	Identity Authorizer group
MDLREV	The Modeling Reviewer Group. Users mapped to this group have access to the menu items in the OFS MMG application that are related to model review activities.
MDLAPPR	The Modeling Approver Group. Users mapped to this group have the rights to approve models created by the users.
MDLBATCHUSR	The Modeling Batch User. Scheduler can use this Group for executing batches.
WKSPADMIN	The Workspace Administrator Group. Users mapped to this group have access to create and populate workspaces. For vieweing the MMG landing page this group is required.
MDLUSR	The Modeling User Group. Users mapped to this group have access to all the menu items in the OFS MMG application that is related to model creation.
DSUSRGRP	Data Studio User Group This User Group provide access to modify Interpreter configurations.
DSREDACTGRP	Roles for applying redaction in graph. This group will be applicable to only those users for whom graph redaction is required.
OBJMIGADMIN	Users mapped to this group have access to Object Migration links and UI to perform import or export of objects.
GRPADMIN	The Graph Administrator Group. Users mapped to this group have access to all the menu items in the OFS MMG Application related to graph and Pipeline/Refresh graphs related health services.

Table 2-2 (Cont.) User Groups

User Group Name	User Group Description
GRPUSR	The Graph User Group. Users mapped to this group have access to all the menu items in the OFS MMG Application related to graph and Pipeline/Refresh graphs related health services.

Note

At the first-time login, User Group mappings are initialized from AAI/IDCS for the newly provisioned users. These will be reflected in MMG Admin Console in next MMG login.

If User Group mappings are deleted in AAI/IDCS, it would not delete in MMG Admin Console. Admin needs to delete this in MMG Identity screens too.

Only the group with MDLSUMM role will be displayed in the Workspace provisioning steps.

MDLSUMM function is mapped to the MDLACCESS role.

Note**Behavior of USER_USERGROUP Map when User directly creates in MMG**

- If new mappings are created in MMG Identity (and not in AAI/IDCS), those will not get deleted (synced with AAI/IDCS) on next login. The Usergroups which are present in MMG Schema and not in AAI/IDCS will not get deleted unless manually removed from MMG.
- If mappings are deleted in MMG Identity (and not in AAI/IDCS), mappings will re-appear on next login.

Role Mapping

Map the user groups in the application to the roles in the following table to enable access to the OFS MMG application.

Table 2-3 User Group to Role Mapping

Group Name	Role Name
DSREDACTGRP	DSREDACT
IDNTYADMN	Batch Advance Role
IDNTYADMN	Batch Write Role
IDNTYADMN	Admin Link Role
IDNTYADMN	User Advanced Role
IDNTYADMN	Group Advanced Role
IDNTYADMN	Role Advanced Role

Table 2-3 (Cont.) User Group to Role Mapping

Group Name	Role Name
IDNTYADMN	Function Advanced Role
IDNTYAUTH	Group Authorize Role
IDNTYAUTH	User Authorize Role
IDNTYAUTH	Group Read Role
IDNTYAUTH	Admin Link Role
IDNTYAUTH	Function Read Role
IDNTYAUTH	Role Read Role
IDNTYAUTH	Role Authorize Role
MDLAPPR	DSINTER
MDLAPPR	Model Authorize
MDLAPPR	Model Deployment
MDLAPPR	Workspace Read
MDLAPPR	Model Read
MDLAPPR	Model Access
MDLAPPR	Workspace Access
MDLAPPR	DSAPPROVER
MDLBATCHUSR	DSBATCH
MDLREV	Workspace Read
MDLREV	Model Review
MDLREV	Model Access
MDLREV	Workspace Access
MDLREV	DSUSER
MDLREV	Model Read
MDLUSR	Model Advanced
MDLUSR	Model Write
MDLUSR	Model Read
MDLUSR	Batch Advance Role
MDLUSR	Model Execute
MDLUSR	DSUSER
MDLUSR	Model Access
MDLUSR	Workspace Access
MDLUSR	Workspace Read
MDLUSR	Datasource Access
MDLUSR	Datasource Write
MDLUSR	Datasource Read
WKSPADMIN	Workspace Access
WKSPADMIN	DSADMIN
WKSPADMIN	Identity MGMT advanced
WKSPADMIN	Workspace Authorize
WKSPADMIN	Workspace Read
WKSPADMIN	Workspace Write
DSUSRGRP	DSADMIN
GRAPHUSER	Graph Administrator
GRAPHUSER	Graph Read Role

Table 2-3 (Cont.) User Group to Role Mapping

Group Name	Role Name
GRAPHUSER	Graph Read Role
GRAPHUSER	Graph Execute Role
GRAPHADMINISTRATOR	Graph Administrator Role

Function-Role Matrix for Conda CRUD Operations

The following table provides details about the Functions and Roles required to perform CRUD operations for Conda in the Oracle Financial Services Model Management and Governance application.

For more information, see the [Conda Environments](#) section.

Table 2-4 User Groups

Function	Role	Groups Mapped	Access
CONDAENVSUMM	CONDAENVACCESS	<ul style="list-style-type: none"> MDLUSR MDLREV MDLAPPR 	Summary view
CONDAENVVIEW	CONDAENVREAD	<ul style="list-style-type: none"> MDLUSR MDLREV MDLAPPR 	Read
CONDAENVEXP	CONDAENVREAD	<ul style="list-style-type: none"> MDLUSR MDLREV MDLAPPR 	Export yml file
CONDAENVEXP	CONDAENVWRITE	<ul style="list-style-type: none"> MDLREV MDLAPPR 	Export yml file
CONDAENVDEL	CONDAENVWRITE	<ul style="list-style-type: none"> MDLREV MDLAPPR 	Delete a registered Conda environment
CONDAENVEDIT	CONDAENVWRITE	<ul style="list-style-type: none"> MDLREV MDLAPPR 	Edit a Conda environment
CONDAENVADD	CONDAENVWRITE	<ul style="list-style-type: none"> MDLREV MDLAPPR 	Add a Conda environment

Sign in

After the application is installed and configured, you can access the Oracle Financial Services Model Management and Governance application.

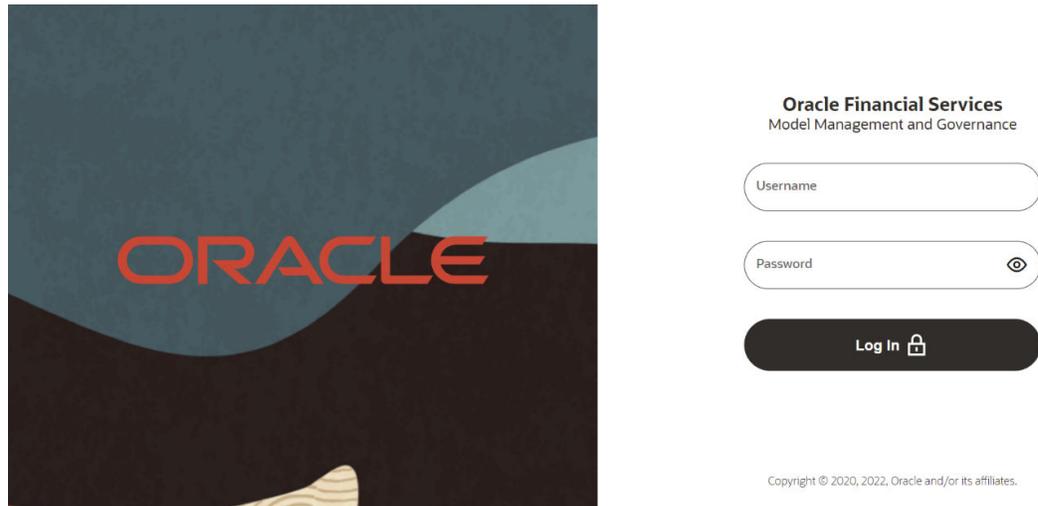
To access the application:

1. Enter the application URL in your browser.

The **Login** page is displayed.

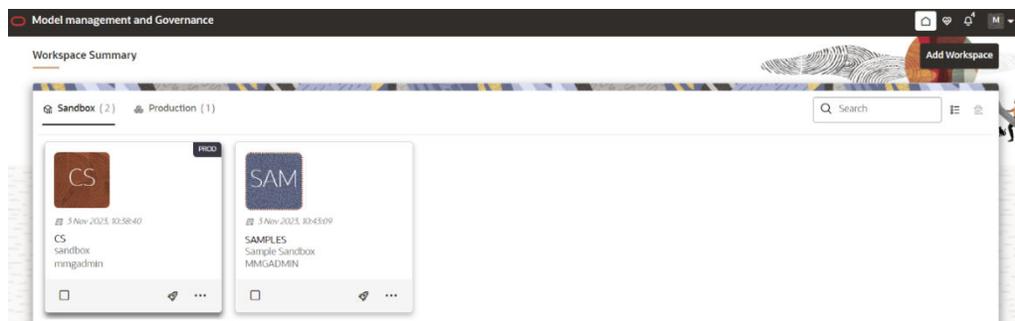
Note

The below screen is an example of AAI type login. The log in page is different for saml type based on the configuration.

Figure 2-2 OFS Model Management and Governance Application Login Page

2. Enter your **Username** and **Password**.
3. Click **Log in**.

The **Home Page** is displayed.

Figure 2-3 Home Page

3

Execute the Production Workspace Models from Analytical Applications Infrastructure

To execute the Production Workspace Models from AAI, follow these steps:

- To execute the Production Workspace Models from AAI, log in to [My Oracle Support](#), and search for **33603448** patch under the **Patches & Updates** tab, download and apply it on the OFSAAI sever.
 - Ensure the AAI Infodomain and the Workspace names are same before you execute the Production Workspace Models.
 - The Logged In user should also be same as the Workspace Create user.
1. In the AAI Configuration Schema, update the value for `MMG_SERVICE_URL` in the `nextgenemf_config` table with the values in the installation.

Example: `MMG_SERVICE_URL : <FQDN/IP>:<mmg_be_port>/<mmg_context_name>`

Note

If the Target Application is using secured connection, import the Target Server Certificate to the AAI Application Java Truststore and restart the services.

2. Restart the AAI services.

A new component `STUDIOMODEL` will be available in ICC, RRF, and PMF modules for selecting the Production Workspace Models of the application.

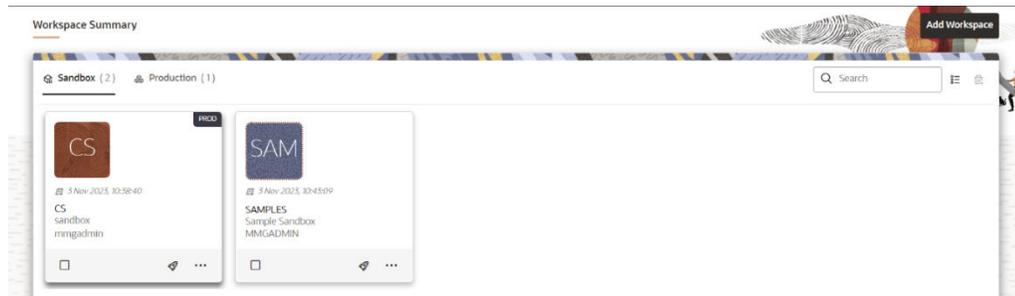
For more details, refer to the Component: STUDIO section of [AAI User Guide](#).

4

User Interface

Access all features from the home page. Familiarize yourself with the menus, icons, and their location to quickly complete your tasks.

Figure 4-1 Workspace Summary page



The following table provides descriptions for the fields and icons on the Workspace Summary page.

Table 4-1 Fields and Icons of the Workspace Summary page

Field or Icon	Description
Search	<p>The field to search for Workspace.</p> <p>Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.</p> <p>You can search a Workspace using Workspace Code, Owner, Creation Date, or Workspace Type fields.</p> <p>The search list shows the Workspace details with Workspace Code, Owner, Creation Date and time, and Type.</p>
Card and Grid View	<p>Click the Card View or Grid View on the top-right corner of this pane to view the workspace as a block or a table, respectively.</p>
Delete	<p>Click Delete to delete the Workspaces.</p>
Workspace Code	<p>The code of the Workspace.</p> <p>NOTE: Use up to 20 characters consisting of alphanumeric and space characters only. Do not use "ALL" as a workspace code.</p>
Description	<p>The description for the Workspace.</p>
Owner	<p>The owner of the Workspace.</p>
Creation Date	<p>The date and time on which the Workspace was created.</p>
Data Source Type	<p>Shows the type of the Data Source.</p>

Table 4-1 (Cont.) Fields and Icons of the Workspace Summary page

Field or Icon	Description
Add Workspace	Click Add Workspace to create a new Workspace. See the Create a Workspace section for more information.
Launch Workspace	Click Launch next to corresponding workspace to display the Workspace Dashboard with application configuration and model creation menu. See the Launch a Workspace section for more information.
View Workspace	Click Action next to corresponding Workspace and select View to view the workspace with dataset data. See the View a Workspace section for more information.
Populate Workspace	Click Action next to corresponding Workspace and select Populate to populate the workspace with dataset data. See the Populate a Workspace section for more information.
Edit Workspace	Click Action next to corresponding Workspace and select Edit to edit the Workspace.
Delete Workspace	Click Action next to corresponding Workspace and select Delete to delete the Workspace. See the Delete a Workspace section for more information.
Download Workspace	Click Action next to corresponding Workspace and select Download to download the Workspace. See the Download a Workspace section for more information. The file is downloaded in .cfg format.

Home Page Components

The Home page contains the following sections:

- Navigation Menu
- Home
- Data Source Summary
- API Configurations
- Conda Environments
- Kafka Topics
- Health
- Notification
- Profile

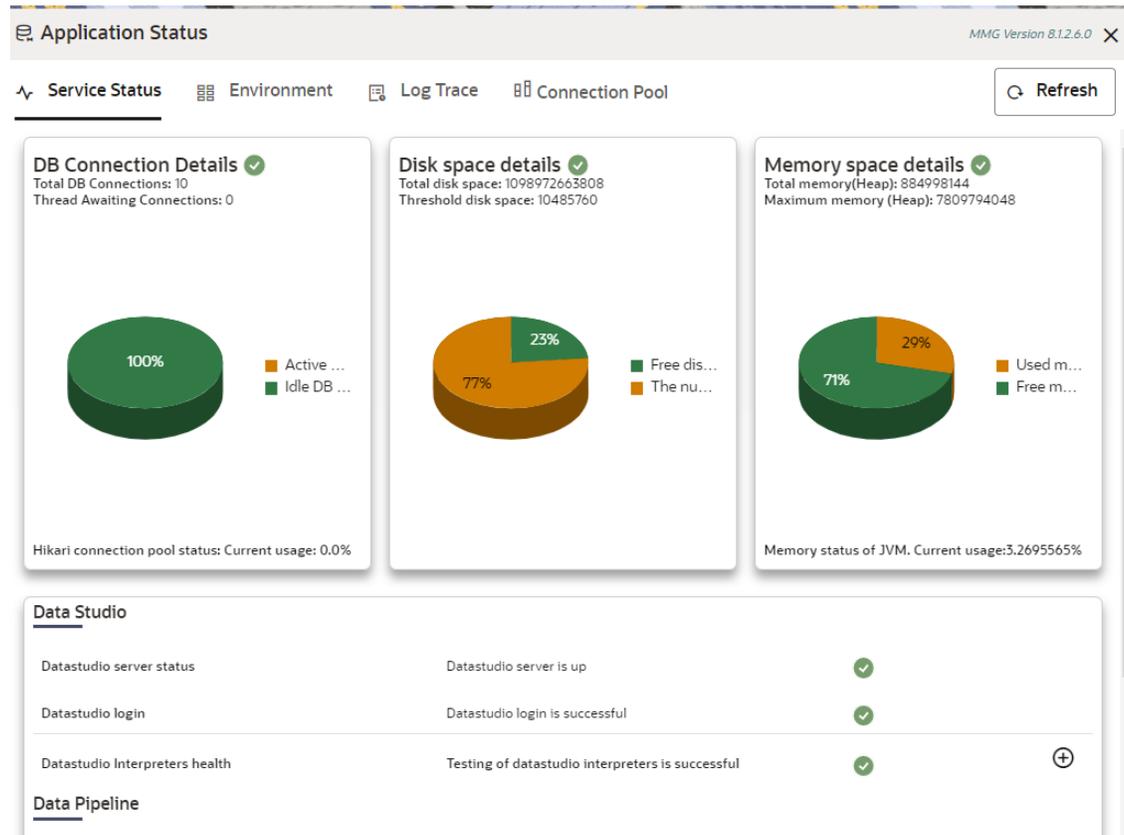
Home

Click **Home** to navigate to **Workspace Summary** page from any other window in the application. You can add and manage Workspaces from this page.

Health

Click **Health** icon to view the application status and version details. The Health page consists of four tabs:

Figure 4-2 Application Status



- **Service Status:** The Database connection, Disk space, Memory allocation details are displayed in graphical chart format. Hover on the chart to view more details. In addition, the status information of the services that are running such as Data Studio, Data Pipeline, and Graph Services are displayed. Click on the + icon to view individual services status.

Note

The Data Pipeline Service Health status is displayed as "Not healthy" when the response is other than "UP" status. You must be mapped to the Graph role 'GRPSUM' as the Data Pipeline Health check is present in Graph services.

- **Environment:** The System and Application properties details are displayed.
- **Log Trace:** The individual logs are displayed that will help you during analysis and debugging of the issue. You can set the last lines of the log based on your requirements. For example, if you wish to see only 10 last lines of the log, enter 10 in the Show last lines text box. By default, the files are displayed in alphabetical order and **mmg-service.log** is displayed. Click on the required log file to view specific details.

Note

Access logs are available in the following location for UI and Services.
 \$LOG_HOME/services/access/mmg-ui-access.log
 \$LOG_HOME/services/access/mmg-service-access.log

- **Connection Pool:** The information of connections used for the data source are displayed. Select the data source from the drop-down list to view the connections and the maximum capacity. This will help you in monitoring the connections of the specific data source. Clicking on the **Refresh** icon will refresh the application status details.

Note

Logs can be downloaded from Health UI screen, under log trace tab. User can download the zip files of individual logs, and Bulk download is also supported. The user needs to have LOGADMN role to access the download option.

Notification

Click **Notification** icon to display the notifications from the application. In the current release, only data population related notifications are displayed. You can delete the notifications by clicking on the **Delete** when no longer needed.

To add or modify the notification details, perform the following:

1. In the header of the Home page, click on the **Notifications** icon.
The Notifications are displayed.
2. In the Notifications screen, click **Notification Settings** icon.
The Notifications Settings page is displayed.
3. Select the required options from the **Notification Settings** page and click **Update Data Population**. If you select the User group, they will be notified via User Interface with the alerts under Notification icon and if you want to get the notifications in email, add the user name and email ID so that whenever a data population is triggered, an email notification is shared to the user.

Figure 4-3 Notification Settings

The screenshot shows the 'Notification Settings' interface. It features a 'DATA POPULATION' section with a 'Default Setting' area containing two dropdown menus: 'Priority' (set to 'High') and 'Category' (set to 'Alert'). Below this is a 'User Group List' section with a text input field containing 'Default mapped user groups x'. At the bottom, there is an 'E-mail List' section with a 'User Name' input field, an 'E-mail' input field, and an 'Add' button. A message at the bottom of the list area states 'No items to display.'

The notification alerts details are updated.

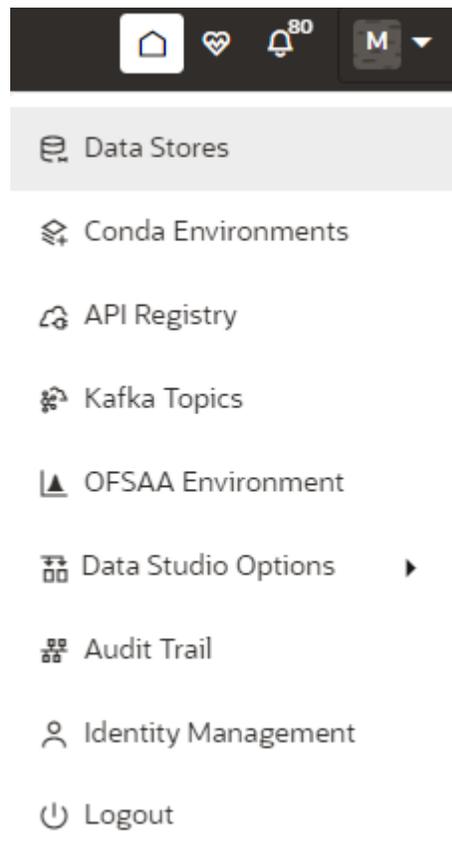
Note

An email notification is sent to the default email id added in AAICL_SMS_USER_PROFILE present in the Config Schema.

Profile

The Header displays icons, buttons, and text for generic information and access to the OFSAA application's features. The following user-interface elements are displayed.

Figure 4-4 User Interface Elements



- **User Name:** Displays the first letter of the logged-in user name. Hover over to view the complete name of the user. Click this to select from the following options in the drop-down list:
 - **Data Stores:** Click to navigate to Data Store Summary page. You can add and manage data stores from this page.
 - **Conda Environments:** Click to navigate to Environment Summary page. You can configure and manage conda environments from this page.
 - **API Registry:** Click to navigate to API Summary page. You can configure and manage APIs from this page.
 - **Kafka Topics:** Click to navigate to Kafka Topics page. You can configure and manage kafka topics and clusters from this page.

- **OFSSA Environment:** Click to navigate to OFSSA Environment page. You can register and OFSSA environments (Simulation and Production) from this page.
- **Data Studio Options:** Contains an underlying Notebook Server which has the following configurable options:
 - * Interpreters
 - * Tasks
 - * Permissions
 - * Credentials
 - * TemplatesClicking on the option will navigate you to the specific page. For more details, see Data Studio Options section.
- **Admin:** Navigates to IDCS page.
- **Logout:** Select this to log out of the application.

Mega Menu Components

To view the Mega Menu components, launch a Workspace. The Dashboard page is displayed with menu options such as Modeling, Orchestration, More, and other sections of the Application.

Modeling

The Modeling component contains the following features:

- **Datasets:** Explore data, engineer features and monitor data drifts.
- **Model Libraries:** Manage python libraries for model creation and usage.
- **Model Techniques:** Manage techniques and algorithms from registered model libraries.
- **Model Catalog:** Centralized hub for all your machine learning models.
- **Pipelines:** Design, deploy and govern end-to-end machine learning pipelines for seamless model development.
- **Graphs:** Analyze complex relationships within your data through interactive graph representations.

For more details, see Modeling section.

Orchestration

The Orchestration component contains the following features:

- **Scheduler Dashboard:** Get an overview of scheduled tasks and processes.
- **Define Batch:** Manage and configure batch definitions.
- **Define Tasks:** Create tasks, configure parameters and set execution dependencies within a batch process.
- **Schedule Batch:** Set execution schedules for your batch processes.
- **Monitor Batch:** Track and monitor batch process executions.

For more details, see [Orchestration](#) section.

More

The More component contains the following features:

- **Export Objects:** Utilize Object Migration feature to efficiently export and save objects.
- **Import Objects:** Import previously exported objects into the system.
- **Model Actions:** Review and approve model deployments in bulk.
- **Audit Trail:** Monitor all system activities and changes.
- **Data Pipelines:** Create and manage data pipelines for efficient data flow and processing.

For more details, see [More](#) section.

Using the Application

The Application displays windows and pages that are interconnected to the OFS AAI. Primarily, you will create Workspace. Within these Workspaces, you will create models, compare them for the best fit, and promote one to production.

Use the information in the following topics to create the workspaces and models:

- [Workspaces](#)
- [Pipelines](#)

The prerequisites for Workspace are as follows:

- To create a workspace, your user profile must be mapped to the Workspace Administrator role.
- To use a workspace to create models, your user profile must be mapped to the Model User role (such as a modeler).

5

Access MMG from Analytical Applications Infrastructure

You can now access the MMG Application and its features from AAI.

Prerequisite

- To access the MMG application from AAI, log in to [My Oracle Support](#), and search for **36336349** patch under the **Patches & Updates** tab, download and apply it.
- A valid MMG setup should be up and running.

Note

Embedding the OFS MMG UI from the OFS AAI page is supported only when both OFS AAI and OFS MMG use the same protocol (either both HTTP or both HTTPS). Mixed protocol usage may cause authentication/session failures due to browser security restrictions.

To access the MMG application, perform the following steps:

1. In the AAI Configuration Schema, update the value for `MMG_UI_URL`, and `MMG_SERVICE_URL` in the `nextgenemf_configtable` with the values in the MMG installation.

Example:

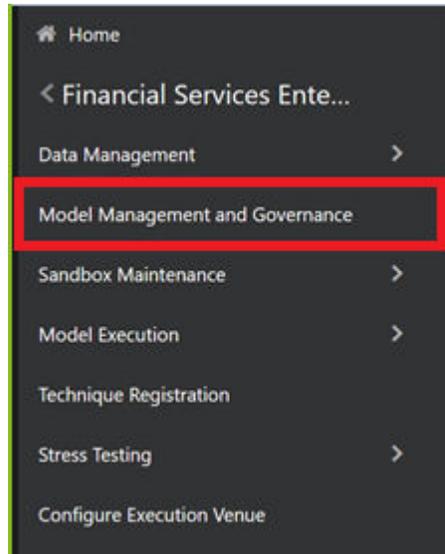
- `MMG_UI_URL:http(s)://<FQDN/IP>:<mmg_ui_port>/<mmg_context_name>/home`
- `MMG_SERVICE_URL :<FQDN/IP>:<mmg_be_port>/<mmg_context_name>`

Note

If the target MMG is using secured connection, import the target server certificate to the AAI application Java Truststore and restart the services.

2. You must map the **MMGACCR** role to the required user. In addition, **MMGACCGRP** group is preceeded with the above role.
3. You must map the `OFS_MMG_HOME` Menu ID to the desired Parent ID in the `AAI_MENU_TREE` table in the AAI Configuration Schema.
4. Restart the AAI services.

After successful login to the AAI application, a menu named **Model Management and Governance** is displayed. Based on your menu tree configuration, the left pane is displayed as shown below:

Figure 5-1 Model Management and Governance menu

5. When you click on Model Management and Governance, the MMG Login page is displayed, enter the credentials to access the MMG application.

Note

MMG Login - Mandatory User Mapping is required to the SAML Application: Before the recent SAML fix from aai, the SAML users created did not have any direct mapping to the SAML Application. All the users that were mapped to the user groups were able to access the respective mapped MMG Applications. However, now after the AAI Authorization and SAML Authentication fix, the user is expected to be mapped to the application.

6

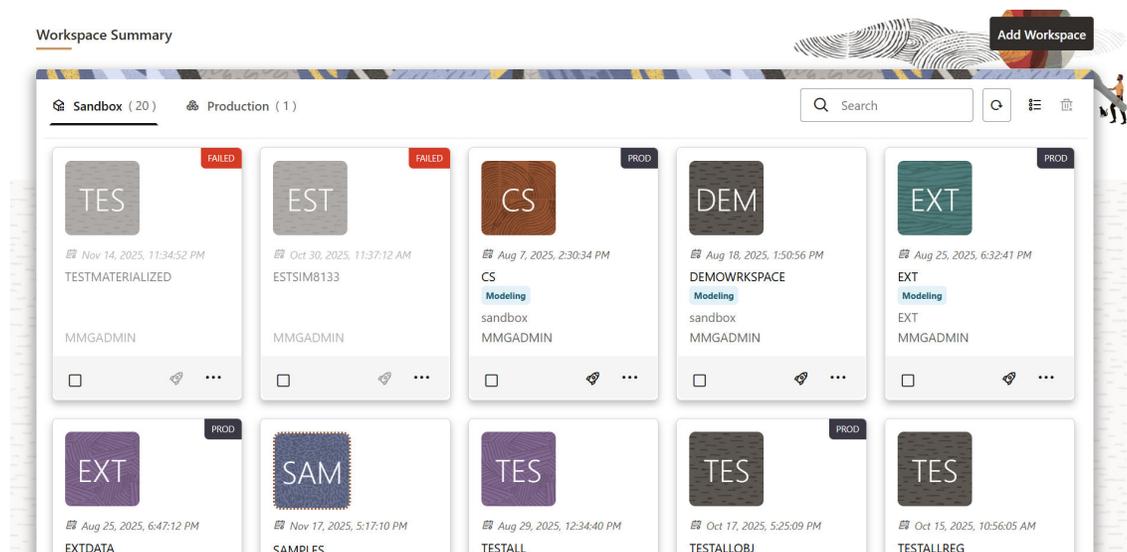
Workspaces

Use this page to create and manage workspaces. By default, a sample workspace is created so that that you can provision and use it to understand the workspace workflow.

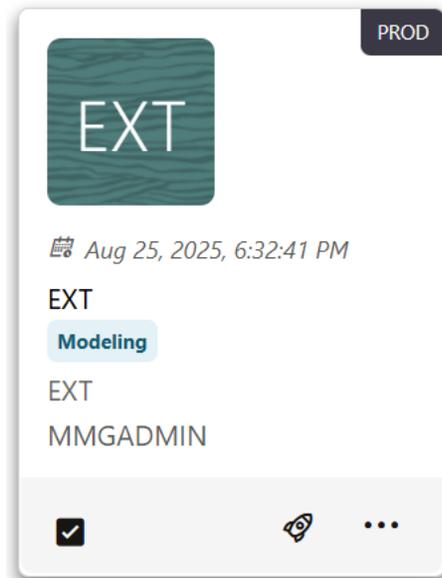
Workspaces can be in the production environment (deployed), or they can exist in a separate instance on their own (on a local system for testing) with a copy of the data that comes from the desired data source (production or external data source).

You can view the following details on the **Workspace Summary** page:

Figure 6-1 Workspace Summary Page



- **Add Workspace icon:** Click this link to [create a workspace](#).
- **Sandbox tab:** This tab is open by default and also displays the number of sandbox workspaces.
- **Production tab:** Click this tab to work in the production workspace. This tab also displays the number of production workspaces.
- **Search field:** This enables you to search for a specified workspace.
- **Refresh icon:** Click this icon to refresh the summary page with the latest data.
- **Grid View or Card View:** Click this icon to toggle between the grid view or the card view.
- **Delete icon:** Select a workspace to enable this icon and delete it.
- **Workspace tiles:** Click any of the available workspace tiles to view its details on the dashboard.

Figure 6-2 An Example of a Workspace Tile

The following list provides information on the workspace tile:

- If a workspace is in production, it will contain the *PROD* status.
- Workspaces with the *Failed* status cannot be selected.
- **Launch icon:** Click this icon within a non *Failed* status workspace to view its details on the dashboard.
- **Action icon:** Click this icon to open the Action menu. The available options are:
 - * [Populate](#)
 - * [Details](#)
 - * [Delete](#)
 - * [Download](#)
- **Page Navigation:** If there are workspaces that span across pages, click the respective page links or the **Next** icon or the **Last** icon to navigate to a specific page.
- **File Manager:** The [File Manager](#) is an ftpshare home path. It enables users to upload a new file, create a new folder, search for a file, preview the file, delete, mark a folder as a favourite and download them.

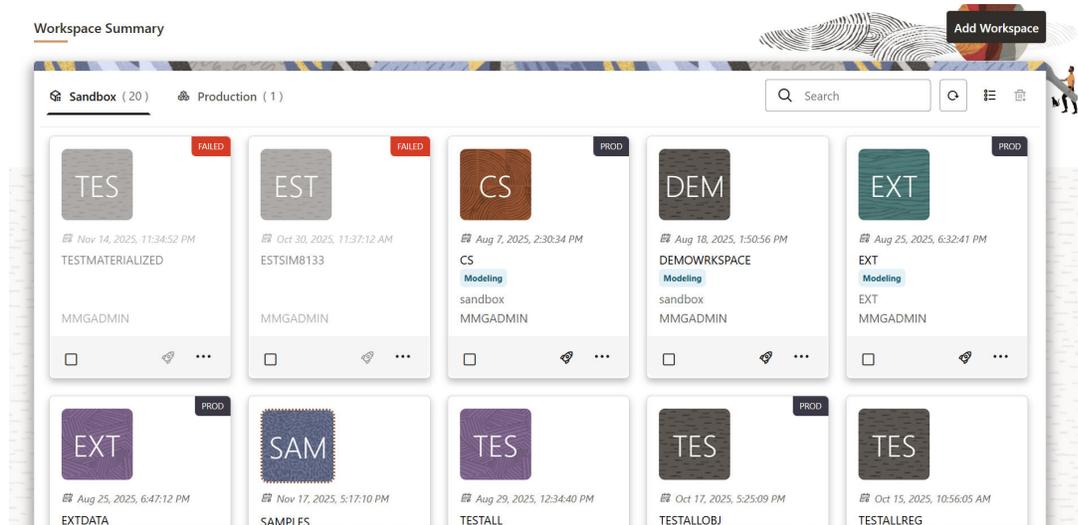
Note

For **File Manager** to be visible on the **Workspace Summary** page, the *File Admin User* must contain the following roles mapped to it

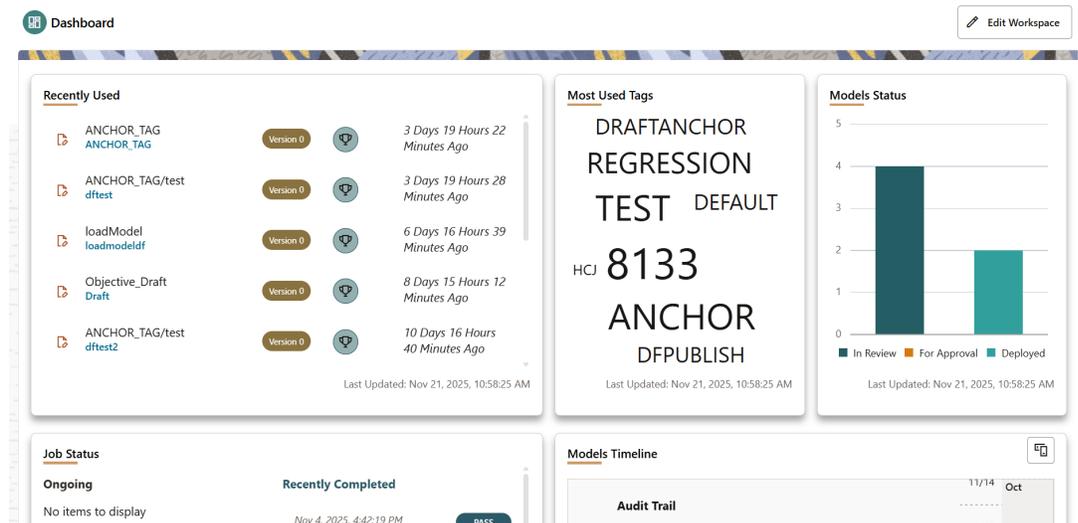
- File Read
- File Delete
- File Write

The workspace **Dashboard** allows you to view the models of the launched workspace. To access the **Dashboard** window, perform the following steps:

1. Navigate to **Workspace Summary** page.
The page displays workspace records in a tile format.



2. Click the **Launch** icon that is within the desired workspace tile view the **Dashboard** page.



The **Dashboard** page displays the following features of a launched workspace:

- **Edit Workspace:** Click this icon to [edit a workspace](#).
- **Recently Used:** This tile displays the most recently used models. When any model from this list is clicked, it navigates the user to the [Model Pipelines](#) page.
- **Most Used Tags:** This displays the tags that have been used the most in the models. Hover over each tag to view the number of models it has been used in.
- **Models Status:** This tile displays the status of the models in a bar graph format. The statuses are; *In Review*, *For Approval*, and *Deployed*.
- **Job Status:** This displays the list of ongoing and recently completed jobs. It also displays the status of the recently completed jobs.

- **Models Time line:** This displays the timeline of the models. Hover over the timeline tile to view the **Zoom In** or **Zoom Out** icons and use them. You can also click the **Orientation** icon to change the orientation of the time line display.

Create a Workspace

The Workspace creation requires entry of the source of dataset, validation, and deployment. Besides, the application may require users of different function groups to create and approve a Workspace. In other words, a user associated with the modeler function group creates a Workspace and the approval and deployment are done by a user associated with the Modeling Administrator function group.

UGDOMMAP function should be mapped to the user performing sandbox creation operation. Otherwise, the create operation will fail.

Two new objects are added during data sourcing in workspace creation:

- Materialized View
- Triggers

Create workspace process can be time-consuming, typically may take between 2 to 3 hours. However, if certain scenarios are not carefully considered, the creation may encounter failures, leading to unwanted delays. To prevent such setbacks and ensure a seamless creation, it is crucial to address these potential issues beforehand.

The possible issue and solution are as follows:

- **Adding Views, Materialized Views, or Triggers without selecting their respective parent tables**
 - **Issue:** If a user adds views, materialized views, or triggers without confirming the selection of their corresponding parent tables, it can lead to the workspace creation failure.
 - **Solution:** It is essential to verify that all dependent objects, such as Views, Materialized views, and Triggers, have their corresponding parent tables included in the selection before initiating the creation process.

To create a Workspace, follow these steps:

1. Navigate to **Workspace Summary** page.

The page displays workspace records in a table.

2. Click **Add Workspace**.

The **Create Workspace** page is displayed.

Figure 6-3 Create Workspace

The window displays a progress indicator at the left that indicates the active window where you are entering details. Click **Previous** to go back a step and click **Next** to go to the next step.

The following steps show the various phases from workspace creation to deployment:

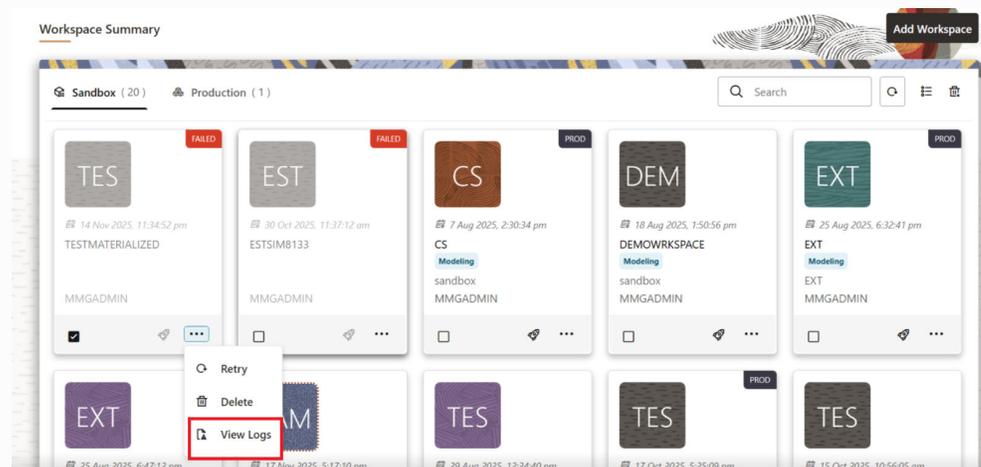
- a. [Basic Details](#)
- b. [Workspace Schema](#)
- c. [Data Sourcing](#)
- d. [Metadata Sourcing](#)
- e. [Validate](#)
- f. [Summary](#)

Note

During the Workspace creation, simulation segment code parameter is added. Simulation segment code can be provided as anything. It should be 10Char Name and it can be same as the workspace name.

Note

The **View Logs** option has been added for failed workspaces to view error log details.



Configure Basic Details

Enter basic configuration details in this window.

Figure 6-4 Basic Details

To configure the basic details for the workspace, follow these steps:

1. Enter the required details in the **Basic Details** pane as shown in the following table:

Table 6-1 Details for Basic Details pane

Field	Description
Use Template	Click on this field to create a workspace from a template. The zip files stored on the path: <Installation path> /scratch/ ofsaadb/ftpshare/mmg/seeded/workspace-templates will be displayed in the Library drop-down. On selecting the template, any pre-filled values will be overridden with the template provided values.
Workspace Code	Enter the code of the workspace.
<div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>Note</p> <p>Use up to 20 characters consisting of alphanumeric and space characters only. Do not use "ALL" as a workspace code.</p> </div>	
Purpose	Enter the purpose of the creation of the Workspace.
User-group	Click on this field to display a list of User-group values. Select the required value. For Example: Modeling Approver, Modeling Reviewer, and Modeling User.
Type	Select the type of Workspace as Modeling or Simulation. NOTE: If Simulation is selected, fields to add the simulation data are displayed. The simulation option is displayed only when the user is mapped to SIMULATIONACCESS role.
Production Workspace	Select the subtype of Workspace as Sandbox or Production. NOTE: This option is not displayed when the Type field is selected as Production. Move the toggle switch to the right to enable this option. Enabling it attaches the production schema, which selects the Workspace being created for automatic model promotion. Based on the selection of the Source Workspace, the model is promoted to the environment.

Table 6-1 (Cont.) Details for Basic Details pane

Field	Description
Import archive file	Enter the archived file to import for basic details. If you use this feature, the other fields described in the preceding rows are auto populated. Click on the box to open the file selector dialog and select the required configuration file or drag the file from its directory and drop it in the box.

- Click **Next** to go to the next step.

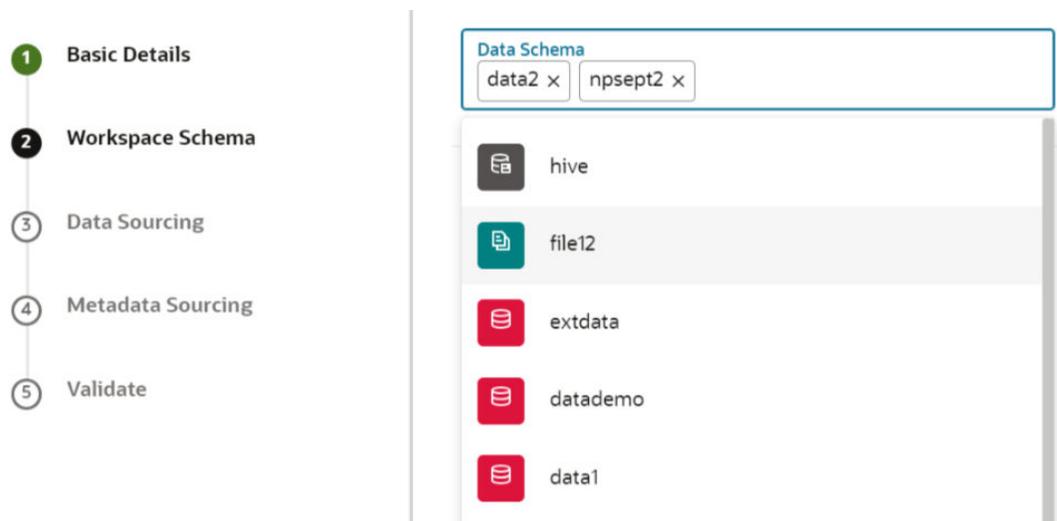
Configure Workspace Schema

Enter schema operation and enter connection details in this window.

Figure 6-5 Workspace Schema

To configure the Workspace Schema:

- Select the required schemas from the **Data Schema** drop-down list. This is the actual data used for model building. You can use multiple Data Schemas for one Meta Source. You can select an existing Schema or add a new Schema by clicking on + icon. The data source options such as Oracle, Hive, and File are displayed.

Figure 6-6 Data Schema

Note

For details on adding a new data store, see [Data Store](#) section.

2. Select the required options from the **Kafka Topics** list. The selected Kafka topics are listed in the right side menu with the order of selection. For more details, see [Kafka Topics](#).
3. Select the required options from the **API Registry** list. The selected API Registry are listed in the right side menu with the order of selection. For more details, see [API Registry](#).
4. Select the required options from the **Conda Environments** list. For more details, see [Conda Environments](#).
5. Click **Next** to go to the next step.

OR

Click **Skip** to skip the step.

Configure Data Sourcing

The schema type selected in the previous step requires the definition of database objects to be used for model creation. The data sourcing step of the workspace provisioning allows you to select tables from Oracle, Hive, or File based data sources through which data has to be pulled into the Oracle based Workspace Data Schemas. However, unlike the data sourcing from RDBMS data sources, the tables will not get physicalized in the target schema, hence it is expected that the tables with compatible structures are already present in the target RDBMS Schema. You can also select DMM operations such as procedures, functions, sequences, and package while configuring the data sourcing. Once a workspace has been provisioned by using DMM, it is stored in ftpshare path - ftpshare/dmm/DATE.

In the case that the selected tables are not present in the target schema, those tables are included in the failed objects count in the workspace provisioning summary.

This window shows the different icons for Oracle, File, and Hive data sources.

Enter the details in this window.

Figure 6-7 Data Sourcing - External Data Source

To configure Data Sourcing, perform the following steps:

1. You can select Data Source from Data Source Name drop-down list or create a new Data Source. To create a new Data Source, see the [Configure Workspace Schema](#) section.
2. Select the Target Data Schema. You can select multiple Data Sources for a Target Data Schema.

3. For example, if there are D1, D2 and D3 Data Sources, then you can select the tables from all these Data Sources, tables from two Data Sources, tables from one Data Source, or as required. Here, multiple combination of tables are possible with Data Source and Target Data Source.
4. If two Data Sources are having same tables (from different Data Sources), then the columns from the first selected table will be used. For example:
5. If table A has columns C1, C2, C3 and Table B has columns C1, C2, and C4, then the data from the first table will be used.
6. During the data population, only columns C1 and C2 will be used and those will be marked in Green color.
7. Select the type of objects to be displayed in the pane that follows the drop-down list. The Object Type drop-down list will be enabled after selecting the Data Source from Data Source Name drop-down list. The following are the options in the drop-down list:
 - Table
 - View
 - Synonym
 - Function
 - Procedure
 - Package
 - Sequence
 - Materialized View
 - Trigger
 - Type
8. Click **Next** to go to the next step. or Click **Skip** to skip the step.

Configure Metadata Sourcing

The database objects selected in the previous step can be added with metadata for selected objects. Metadata Sourcing is a stage during Workspace provisioning to allow seeding of metadata like scheduler batches at the time of workspace provisioning.

Also, by default there will be seeded metadata. However, if you wish to seed the metadata, navigate to <installed path>/ftpshare/mmg/seeded/batches and drag and drop the metadata in SQL format.

Note

This step is optional.

Figure 6-8 Metadata Sourcing

To configure Metadata Sourcing, follow these steps:

1. Select the required schema from the **Object Type (Optional)** drop-down list.
The **Available Objects** are displayed in **Selected Objects** pane.
2. Click **Next** to go to the next step. or Click **Skip** to skip the step.

Validate Workspace

The **Validate** pane displays a preview of the configuration values entered in the previous panes.

Figure 6-9 Validate Workspace

To validate the workspace and deploy it, perform the following steps:

1. Review the details in the **Validate** pane. You can edit the workspace by clicking **Edit**.
2. Click the **Finish** drop-down list. The available options are:
 - **Physicalize Workspace:** When you click this option, it creates the actual workspace.
 - **Download Configurable Archive:** When you click this option, it exports the metadata information of the workspace in a .zip format. This can be used later by using the **Import** option.

Display Summary

The **Summary** pane displays the status of the workspace creation.

Figure 6-10 Workspace Creation Summary



- Click **Download** to download the configurable archive (workspace created).

Note

During the workspace creation process, if any of the DMM operations such as procedures, functions, packages, and so on *fails* then the workspace creation will be rolled back. You can verify the details in the **MMG_SANDBOX_DETAILS**, **MMG_SANDBOX_MASTER**, and **MMG_SANDBOX_SCHEMA** tables.

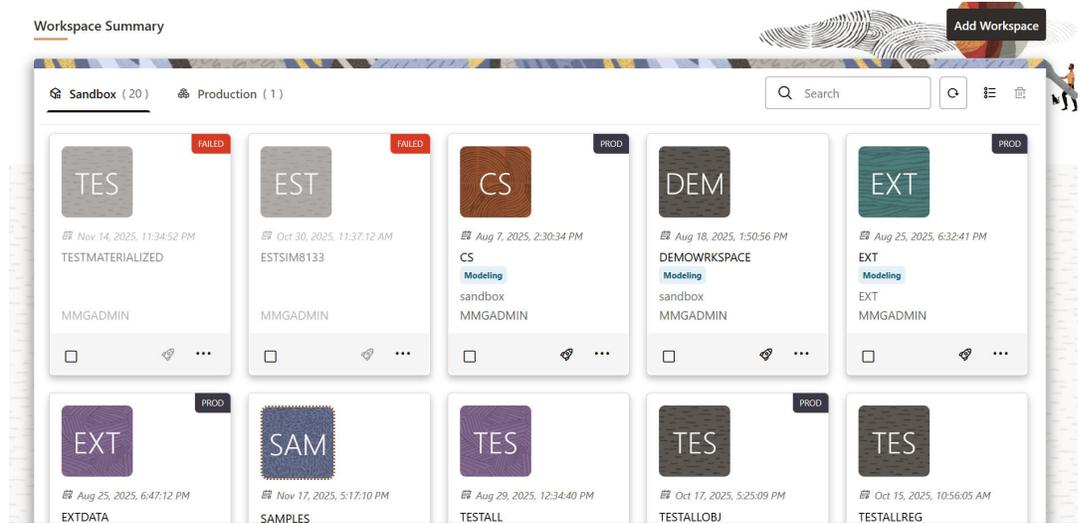
Populate a Workspace

The workspace is populated with data from the datasets in external sources.

To populate the workspace, perform the following steps:

1. Navigate to the **Workspace Summary** page.

The page displays workspace records in a table.

Figure 6-11 Workspace Summary Page

2. Click the **Action** icon within the desired workspace tile and then select **Populate** to populate the workspace with data from a dataset in the **Populate Workspace** window.

Figure 6-12 Populate Workspace window

Populate Workspace
✕

Workspace Code CS	Purpose sandbox	Creation Date 2024-02-12 14:46:15	Data Store Type External Data Source
-----------------------------	---------------------------	---	--

Write Mode ?

Write Mode
Append
▼

In this mode, the underlying tables will be populated in append mode.

Data Load SELECTIVE ALL

In this mode, only the tables filtered (selected in the Table level Data Filters) will be populated.

Data Filters - Global level ? Use Template

Data Filters - Global

Data Filters - Table level ? +

Tables

SQL Filter

🗑️

Data Filter - Hint ? +

Tables

Source Hint

Target Hint

🗑️

Required

Additional Parameters ?

Source Prescript

Target Prescript

Fetch Size
10

Batch Commit Size
1,000

Select Unlimited or Customize the Rejection Threshold

Unlimited Custom Rejection Threshold

Rejection Threshold
UNLIMITED

Cancel
Populate Workspace ▼

The following table provides descriptions for the fields in the Populate Workspace window.

Table 6-2 Details for the Populate Sandbox Window

Field	Description
Workspace Code	Displays the code of the workspace.
Purpose	Contains the description of the workspace.
Creation Date	The date and time on which the workspace was created.
Data Source Type	The source of the data. The value can be the <i>OFSAA Data Schema</i> or an <i>External Data Source</i> .
Write Mode	You can either overwrite the existing data (truncate and insert) or append to the existing data. <ul style="list-style-type: none"> • Overwrite: In this mode, the underlying tables are truncated (overwritten on existing data) followed by an insert operation. • Append: In this mode, the underlying tables will be populated (added to the existing data) in the append mode.
Data Load	You can select a few tables or add all the tables from the source. The available options are: <ul style="list-style-type: none"> • SELECTIVE: In this mode, only the filtered tables (selected in the Table level Data Filters) are populated. • ALL: In this mode, all the underlying tables mapped to the workspace are populated along with the specified filters for specific tables.
Data Filter- Global Level	Enter the data filter that needs to be applied on all the tables selected for data sourcing. For example: If MISDATE is equal to Today , then it is applied to all tables (wherever it is available) for selected Data Sources during population. If this field is not found (MISDATE) in the tables, it is not updated. Or Click Use Template to open the Template pop-up window. Select a template from the Template drop-down list. On selecting the template, any pre-filled values will be overridden with the template provided values.
Data Filter- Table level	You can provide the data filters individually for the tables. Select the table and enter the SQL filter. Additionally, click the Add icon to add more rows or click the Delete icon to delete a row. Note: You can provide multiple table names for the same SQL filter. For example, there are two tables called <i>Student</i> and <i>Employee</i> in the target data source, and the following filters are applied: MISDATE as Today for <i>Student</i> and <i>Employee</i> tables ID as 1 for the <i>Student</i> table Then, the <i>Student</i> table will be populated with the MISDATE and ID filters. The <i>Employee</i> table will be populated with only the MISDATE filter. Note: Global filters are not applicable for those tables on which filters have been applied individually. If the same table name is provided in more than one row, then the filter condition is generated as a conjunction of all the provided filters.

Table 6-2 (Cont.) Details for the Populate Sandbox Window

Field	Description
Data Filter - Hint	<p>You can provide database hints at table-level and SQL prescripts at the schema level for data load performance improvement during the workspace population. Additionally, click the Add icon to add more rows or click the Delete icon to delete a row. Select the table and enter source and target hint filters. You can add multiple filters by clicking the Add icon.</p> <p>Note:</p> <p>You can also pass runtime values in the workspace population for the user-defined parameters in the Data Filter – Global/Table field.</p> <p>Example: Table Filter /Global Filter</p> <pre>[{"id":1,"filter":"CUSTOMER_ID =\$CUSTID and INCOME =\$income and CUSTOMER_NAME =\$customerName","tables":["CUSTOMERS"]}]]</pre> <p>Additional Parameters:</p> <pre>{"fetch_size" :10, "batch_commit_size" :1000, "rejection_threshold" : "UNLIMITED", "write_mode" : "APPEND", "dynamic_parameters": [\ {"key": "\$CUSTID", "value": "125"} , {"key": "\$income\$", "value": "30000"} , {"key": "\$customerName", "value": "Cust125"}]}</pre> <p>The Runtime parameters can be passed on as a part of the additional parameters json .Key = "dynamic_parameters" .It will replace all the \$ values in Table filter /Global Filters.</p>
Source Prescript	Enter the source prescript of JDBC properties for data upload.
Target Prescript	Enter the target prescript of JDBC properties for data upload.
Fetch Size	Enter the Fetch size of JDBC properties for data upload
Batch Commit Size	Enter the Batch Commit size of JDBC properties for data upload
Write Mode	You can either overwrite the existing data (truncate and insert) or to append to the existing data. You can choose to either overwrite the data or append to the existing data.
Rejection Threshold	<p>The following two options are available:</p> <ul style="list-style-type: none"> • Unlimited - All the errors will be ignored during the data population. • Custom Rejection Threshold - Enter the maximum of number of inserts that may fail any of the selected tables. You can provide the maximum number of inserts that can fail while loading data to a given table from all the sources. In the case of threshold breach, all the inserts into the particular target schema will be rolled back. However, it will continue populating the next target schema.

3. Click **Populate Workspace** drop-down list to select between the two options; **Create Batch** or **Create and Execute Batch**.

You can create the batch by using **Create Batch**, or create and execute by using the **Create and Execute Batch** option. When you select either one of these options, a workspace population task gets added to the batch.

- **Create Batch:** When you select this option, it allows you to prepare the batch and then execute or schedule the batch at a later time through the **Scheduler Service** window. The workspace population task execution can be tracked in the **Monitor Batch** window in the **Scheduler Service**.

- **Create and Execute Batch:** When you select this option, it allows you to create the batch and triggers the batch as well.
- For a workspace population task, you have to mandatory check the logs to check if the target table is getting populated. Workspace population logs can be found under:
<MMG_HOME>/OFS_MMG/logs/execution/<MISDATE>/<WORKSPACENAME>/workspace-population/

Figure 6-13 Task Parameters

▼ Task Parameters

Parameter	\$BATCHDATE\$	Value	Batch Date	<input checked="" type="checkbox"/>
Parameter	\$BATCHRUNID\$	Value	BATCHRUNID	<input checked="" type="checkbox"/>
Parameter	Global Filter	Value	<input type="text"/>	<input type="checkbox"/>
Parameter	Table Filters	Value	<input type="text"/>	<input type="checkbox"/>
Parameter	Additional Parameters	Value	<input type="text"/>	<input type="checkbox"/>

4. Enter the following parameters for workspace population.

This step is optional.

- **Additional Parameters:** Enter the Additional Parameters in following format:

```
{ "fetch_size" :10, "batch_commit_size" :1000,
  "rejection_threshold" : "UNLIMITED", "write_mode" : "OVERWRITE" }
```
- **Global Filter:** The provided input will be applied as a data filter on all the tables that are selected for data sourcing.
- **Table Filter:** You can provide data filters individually on the tables here. You must provide multiple table names for the same SQL filter. Global Filters will not be applicable for those tables on which filters have been applied individually. In case the same table name is provided in more than one rows here, the filter condition will be generated as a conjunction of all the provided filters.

Enter the Table filters in following format:

```
[{"id":1,"filter":"<filter condition>","tables":["TABLE1",
"TABLE2"]}, {"id":2,"filter":"<filter condition>","tables":["TABLE2"]}]
```

📘 Note

You can run a workspace population for a given workspace any number of times. New tables may be added to the definition. Any new table added to the definition, that is not present in the target schema will be physicalized on update of the workspace. You can also add new sources if required. Any table that is deselected from the data sourcing definition will **NOT** be dropped.

Retry a Failed Workspace

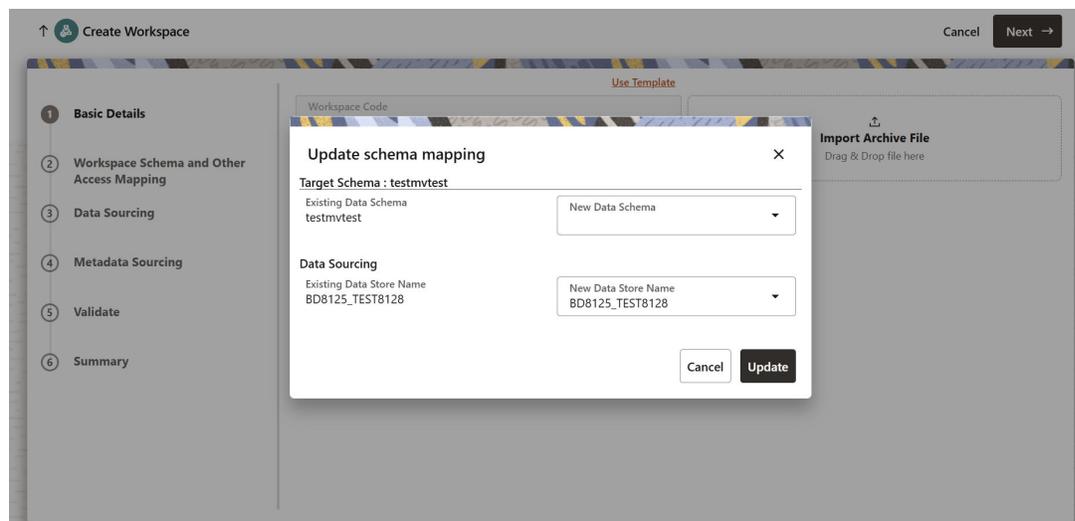
Perform the following steps to retry a workspace that has failed.

Note

This option is only available for failed workspaces.

1. Click the **Action** icon adjacent to the workspace with a *Failed* status.
2. Click **Retry**.
The **Create** workspace window opens with the **Update schema mapping** window.

Figure 6-14 Update schema mapping Window



3. Select a new data schema and new data store name from the **New Data Schema** and **New Data Store Name** drop-down lists respectively.
4. Click **Update**.
5. Additionally, after updating the data schema and data store, on the Create Workspace page, you can modify all the details in the respective workflow except for the workspace code.

Delete a Workspace

Note

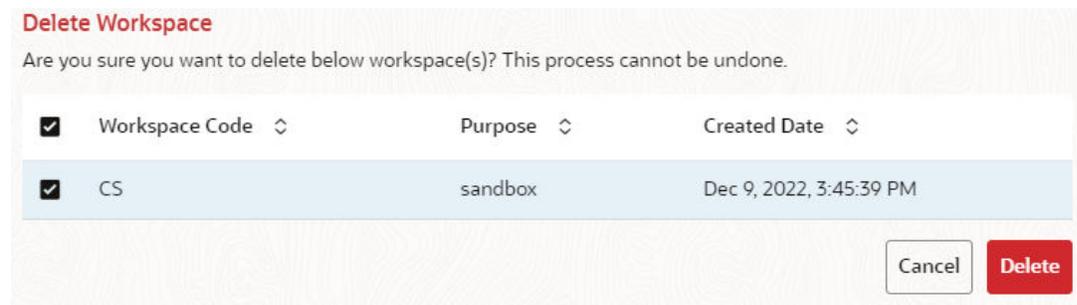
When you delete a Workspace, all the underlying objects such as Dataset, Scheduler service metadata and so on from the associated tables are deleted.

To delete a Workspace, follow these steps:

1. Navigate to **Workspace Management** to display the **Workspace Summary** page. The page displays workspace records in a table.

2. Click Action next to corresponding Workspace and select **Delete** to delete the workspace.

Figure 6-15 Delete Workspace window



The following table provides descriptions for the fields in the Delete Workspace window.

Table 6-3 Details on the Delete Sandbox window

Field	Description
Workspace Code	The code of the workspace.
Purpose	The description for the workspace.
Created Date	The date on which the workspace was created.

3. Click **Delete**.
4. Click **OK** on the confirmation dialog box to confirm or click **Cancel** to cancel.

Note

Ensure that you de-link the Sandbox workspaces from Production Workspace before deleting the Production Workspace.

Download a Workspace

Downloading a workspace allows you to export the workspace metadata definition (without underlying objects such as Models, Pipeline, Graphs, and so on) in a zip format. Further, the same can be used to re-create the workspace.

To download a workspace, follow these steps:

1. Navigate to **Workspace Management** to display the **Workspace Summary** page. The page displays Workspace records in a table.
2. Click Action next to corresponding workspace and select **Download** to download the workspace.

OR

Navigate to the Summary screen during the creation of a Workspace and click Finish and select **Download Configuration Archive** option.

Note

The file is downloaded in .cfg format.

Managing Data Sources

This feature allows you to manage the Data Schemas registered with the application. The Data Source Summary window shows the list of Data schemas registered with the application. These Data schemas can be used either for workspace or for sourcing data. Click **Data Source Summary** to navigate to Data Source Summary window.

This window also allows you to manage these registered external sources.

The Data Source Summary is divided into two sections: Used Data Source and Unused Data Source.

To view the Data Source details., Click **Action** next to corresponding Workspace and select **View**

- **Used Data Source:** This shows the list of Data Sources registered with any workspace. Here, you can only view the Data Source details. The count of Used Data Sources also displayed at the top of Data Source Summary page.
- **Unused Data Source:** This shows the list of Data Sources those are not registered with any workspace. Here, you can only view, edit, or delete the Data Source details. The count of Unused Data Sources also displayed at the top of Data Source Summary page.

To add a Data Source from Workspace Summary window, follow these steps:

1. Navigate to **Workspace Summary** window.
2. Click **Data Source Summary** to navigate to **Data Source Summary** window and click **Add Datasource**
The Add Data Source window is displayed.

To add a Data Source from Data Source Summary window, follow these steps:

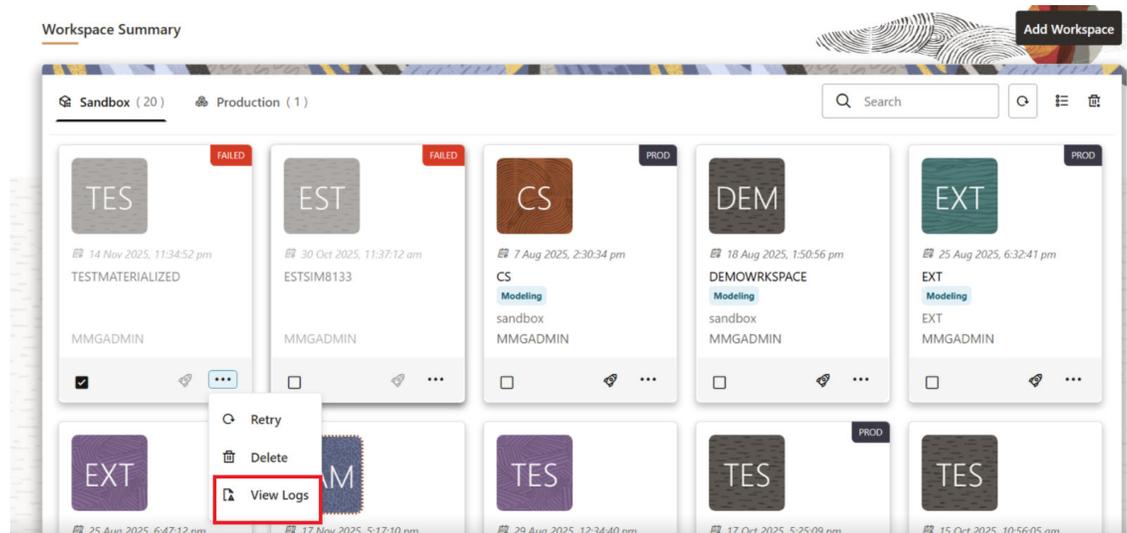
1. Navigate to **Data Source Summary** window.
2. Click **Add Datasource** .

For more information, see the [Configure Data Sourcing](#) section.

Details Screen

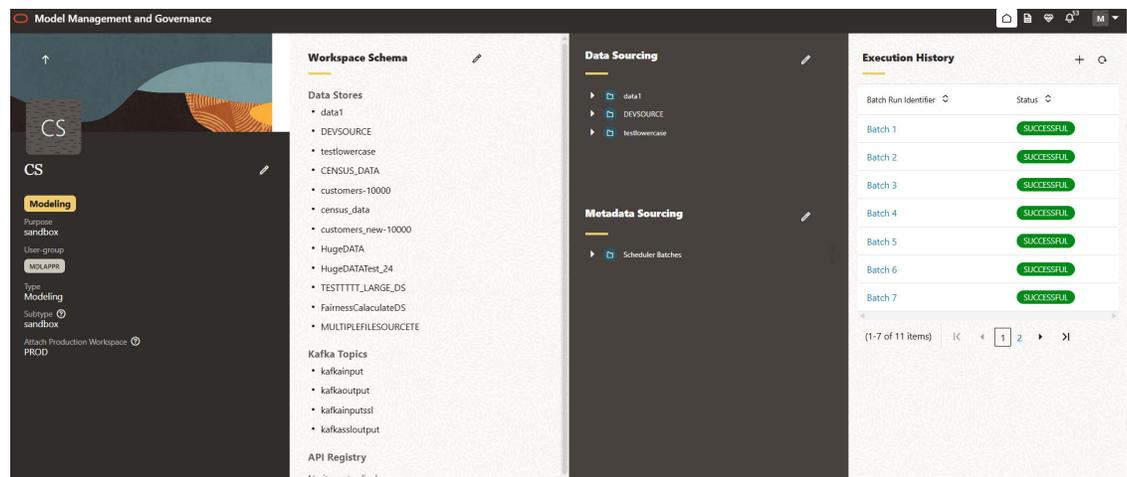
The **Details** screen enables you to edit and view a workspace.

Log in to the application. From the **Workspace Summary** screen.



Click the **Action** icon and then click **Details**, to navigate to the **Details** screen.

Figure 6-16 Details screen



This screen is divided into four columns: The first three columns, basic details, **Workspace Schema**, and **Data Sourcing**, provides information about the workspace and fourth column, **Data Population Jobs**, focuses on the execution history of the workspace population.

The information present in first three columns is the one that users configure when creating or editing a workspace.

The workspace data or information is given by the user when they create or edit a workspace.

The first column is the basic details screen. It contains the workspace name, purpose, user-group, type, subtype, and the attach production workspace details. You can edit the details in this section if you have edit rights.

Workspace Schema

The Workspace Schema is divided into four sections:

- Data Stores
- Kafka Topics
- API Registry
- Conda Environment

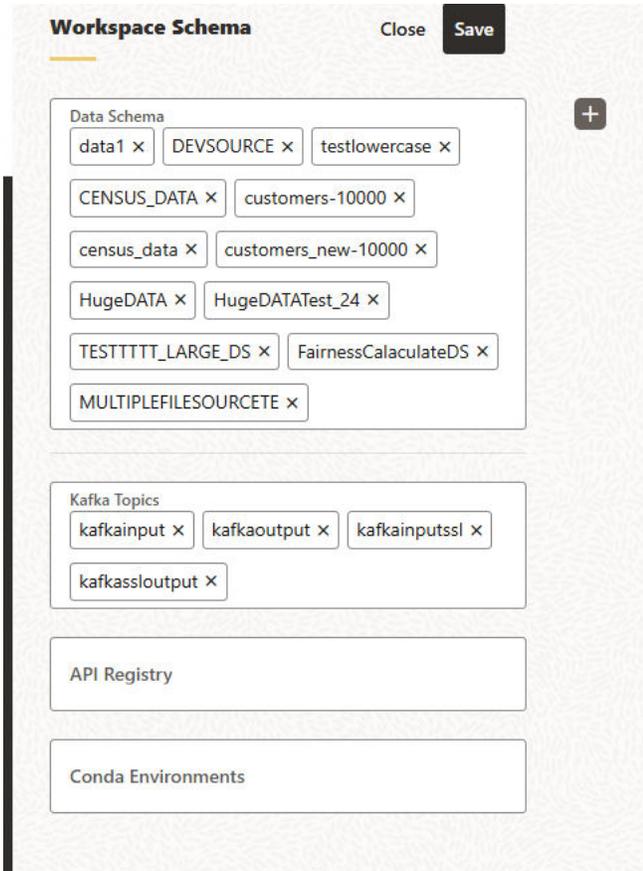
Figure 6-17 Workspace Schema



Click the **Edit** icon adjacent to **Workspace Schema** to edit the same.

The view changes to edit mode. Modify the details and then click **Save** to save the information.

Figure 6-18 Workspace Schema



Click the **Create Data Store** icon adjacent to the Data Schema screen to create or add a **Datstore**.

Figure 6-19 Add a Datastore

The screenshot shows a dialog box titled "Add Data Store" with a close button (X) in the top right corner. The dialog contains the following fields and controls:

- Data Store Name**: A text input field with a "Required" label to its right.
- Description**: A text input field with a "Required" label to its right.
- File Availability**: A dropdown menu currently set to "JDBC".
- Database Type**: A dropdown menu currently set to "Oracle".
- Wallet Alias**: A text input field with a "Required" label to its right.
- Table Owner**: A text input field.
- Additional Properties**: A section header with a help icon (?) and a question mark.
- JDBC Connection String**: A text input field.
- User Name**: A text input field.
- Password**: A text input field with a toggle icon (eye) to its right.

At the bottom right of the dialog, there are three buttons: "Test Connection", "Cancel", and "Create". The "Create" button is highlighted in black.

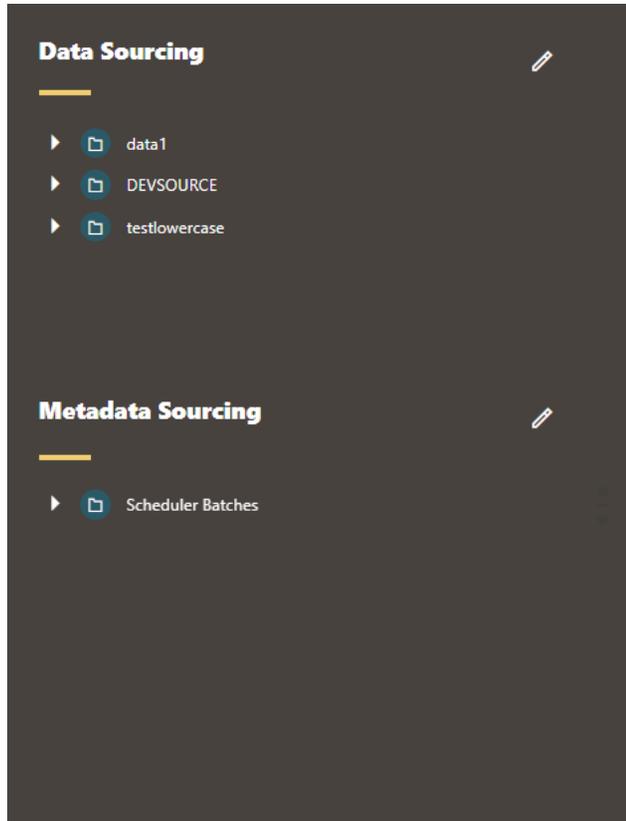
Data Sourcing

Data Sourcing is divided into two sections:

- Data Sourcing
- Metadata Sourcing

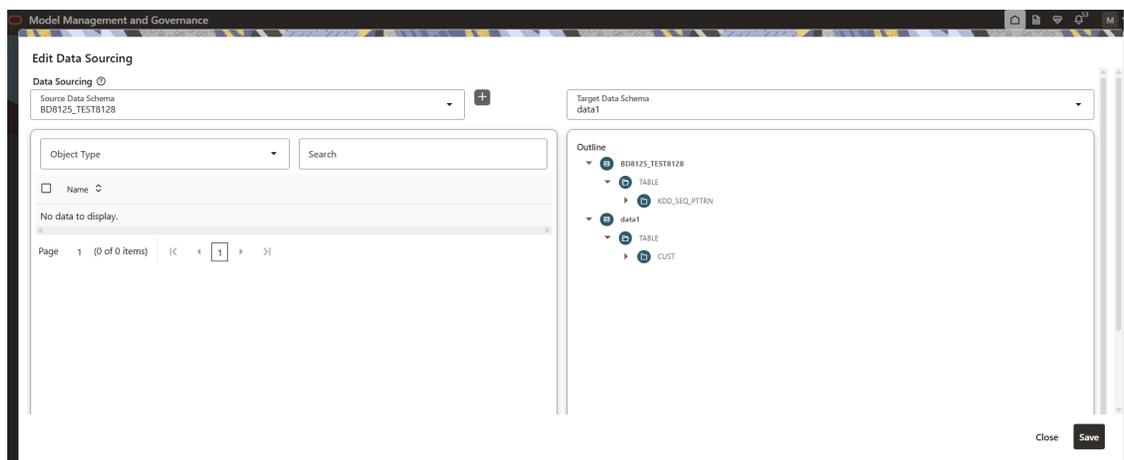
Click the **Add** icon adjacent to **Data Sourcing** to edit it.

Figure 6-20 Data Sourcing



The Edit Data Sourcing pop-up window appears.

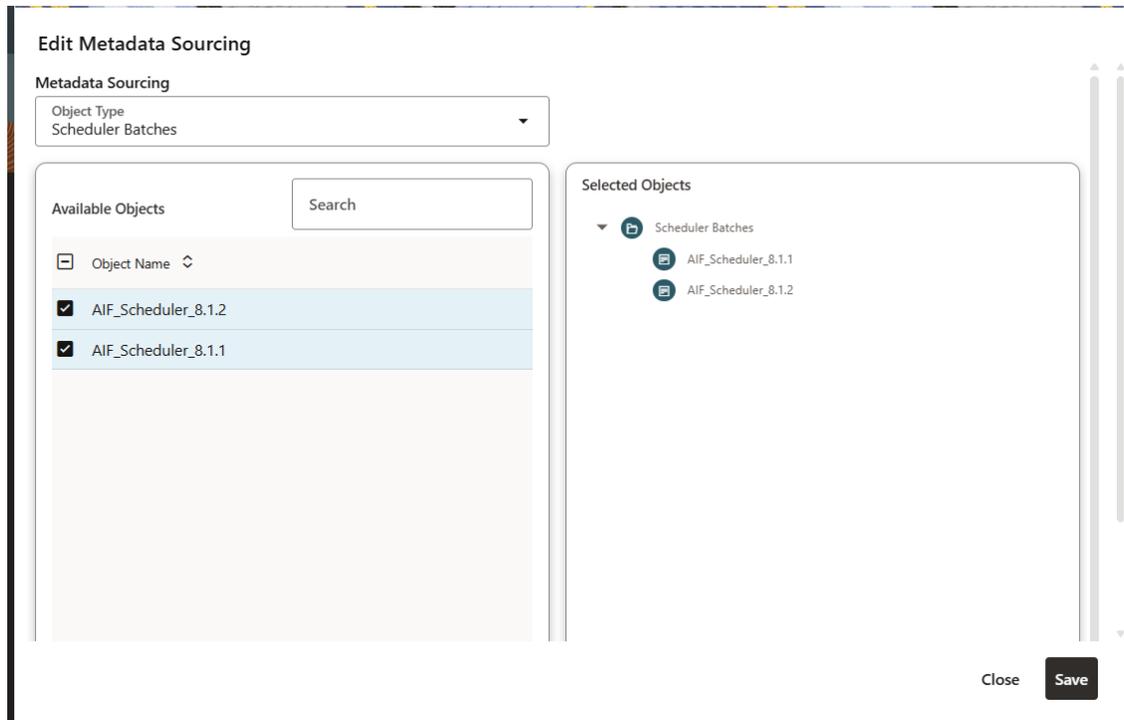
Figure 6-21 Edit Data Sourcing



Modify the desired details and then click **Save**.

Click the **Add** icon adjacent to **Metadata Sourcing** to edit it.

The Edit Metadata Sourcing pop-up window appears.

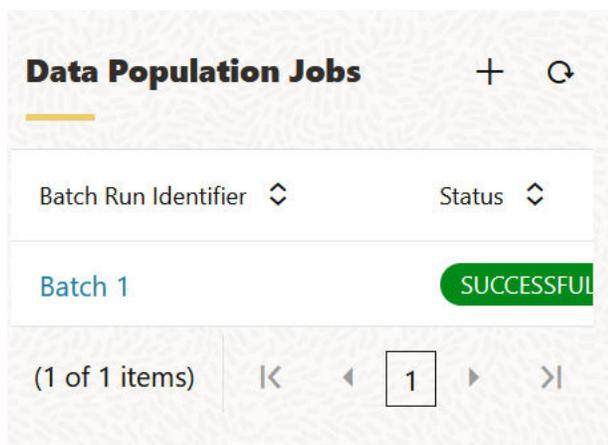
Figure 6-22 Edit Metadata Sourcing page

In this window, you can add new objects or edit the existing ones.

Modify the desired details and then click **Save**.

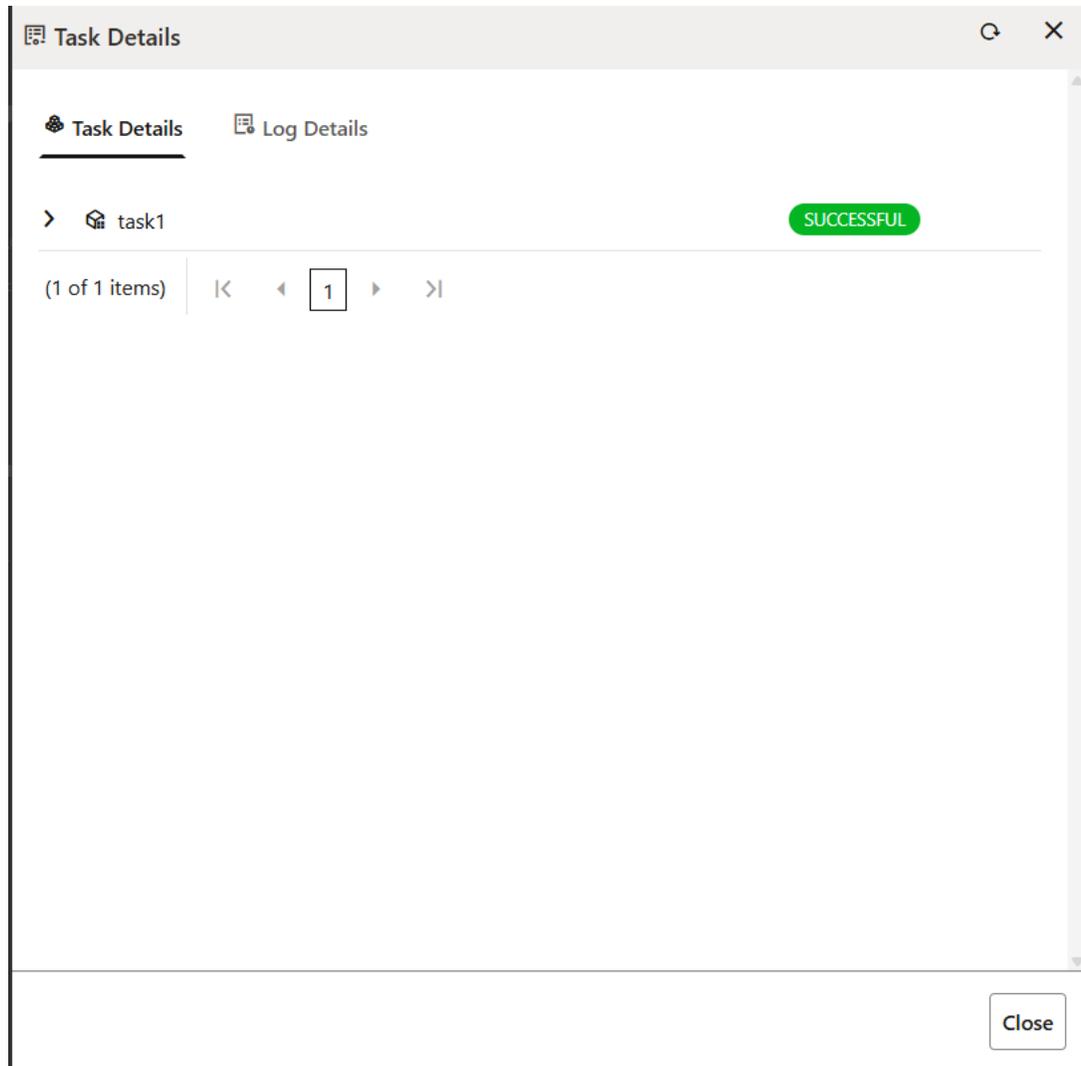
Data Population Jobs

In the Data Population Jobs section, the you can view the output of the population under **Batch Run Identifier** and **Status** columns for the workspaces.

Figure 6-23 Data Population Job

To view details about a listed batch, click the batch link to open the **Task Details** window.

Figure 6-24 Task details



Click **Close** to close the window.

Click the **Populate** icon to create a new data population job.

Figure 6-25 Populate Workspace page

Populate Workspace

Workspace Code	Purpose	Creation Date	Data Store Type
CS	sandbox	2025-02-12 10:39:22	External Data Source

Write Mode ⓘ

Write Mode
Overwrite

In this mode, the underlying tables will be populated in truncate and followed by insert operation

Data Load SELECTIVE ALL

In this mode, all the underlying tables mapped to the workspace will be populated along with the filters mentioned below for specific tables.

Data Filters - Global level ⓘ [Use Template](#)

Data Filters - Global

Data Filters - Table level ⓘ

Tables SQL Filter

Hints ⓘ

Tables Source Hint Target Hint

Required

Cancel Populate Workspace

Hover over the ? icon over each field to view more information about it. Populate the fields in this window:

- **Write Mode** - Select either *Overwrite* or *Append*.
- **Data Load** - Select either *SELECTIVE* or *ALL*.
- **Data Filters - Global level** - Enter values or click the **Use Template** link to select a template from the Template the drop-down list.
- **Data Filters - Table level** - Select a table from the **Table** drop-down list. Add an SQL filter in the **SQL Filter** field. Additionally, click the **Delete** icon to delete the data.
- **Hints** - Select a table from the **Table** drop-down list. Add a value for the **Source Hint** and **Target Hint** fields. Additionally, click the **Delete** icon to delete the data.
- **Additional Parameters** - Enter values in the **Source Prescript** and **Target Prescript** fields. You can also modify the values in the **Fetch Size** and **Batch Commit Size** fields.

- **Additional Parameters** - Select either **Unlimited** or **Custom Rejection Threshold**. If **Custom Rejection Threshold** is selected, then the Rejection Threshold fields is enabled for editing.

Click the **Populate Workspace** drop-down list and click either **Create Batch** or **Create and Execute Batch**.

Click **Save**.

Note

The **View Workspace** screen and the [Edit Workspace](#) screen has been merged into one screen and is known as the Details screen.

Edit a Workspace

The Edit Workspace process can be time-consuming, typically may take between 2 to 3 hours. However, if certain scenarios are not carefully considered, the update may encounter failures, leading to unwanted delays. To prevent such setbacks and ensure a seamless update, it is crucial to address these potential issues beforehand.

The possible issue and solution are as follows:

- **Adding Views, Materialized Views, or Triggers without selecting their respective parent tables**
 - **Issue:** If a user adds views, materialized views, or triggers without confirming the selection of their corresponding parent tables, it can lead to the failure of the edit workspace update.
 - **Solution:** It is essential to verify that all dependent objects, such as Views, Materialized views, and Triggers, have their corresponding parent tables included in the selection before initiating the update process.
- **Manually adding additional tables in the Target Workspace Schema instead of using Workspace functionality**
 - **Issue:** If a user manually creates a table (for example, ABC) in both the Source and Target Schemas, independent of the workspace functionality, and then tries to map it to an existing workspace, the edit process may fail. This would lead to an inconsistency between the Target Schema and Source Schema Database objects.
 - **Solution:** To resolve this issue you need to sync up the inconsistency between the Workspace Source and Target Database objects, for which you can execute the following API with the respective headers and payload.
 - * **HTTP Method:** POST
 - * **URL:** `http(s)://<MMG_BE_HOSTNAME>:<MMG_BE_PORT>/<MMG_CONTEXT_NAME>/dmm-service/v1/modelupload/dbcatalog`
 - * **Header Parameter:**
 - * **ofs_workspace_id:** MMG Workspace Code.
 - * **ofs_remote_user:** Login User of the application.
 - * **locale:** Locale in languageCode-countryCode format. The values should be en-US.
 - * **ofs_service_id:** Target Workspace Data Store Name.

- * **ofs_connection_type:** Connection type and the value should be OFSAA-DATA.

- * **Payload:**

- * **Case 1:** This will retrieve all table definitions, and since the list is empty, it will fetch all tables, making the process time-consuming.

```
{
    "tableNames": []
}
```

- * **Case 2:** This will retrieve only the tables specified in the request body. If you want to include specific tables, ensure they are listed.

```
{
    "tableNames": ["A", "B"]
}
```

- **Response:** The response will serve as an acknowledgment that the request has been received, and it will be asynchronous due to the time-consuming nature of the process.

- * **Success:**

```
{
    "status": "received",
    "errorMessage": null,
    "additionalInfo": null
}
```

- * **Error:**

```
{
    "status": "error",
    "errorMessage": <Error Message>,
    "additionalInfo": null
}
```

Once the execution is successful, utilize the workspace edit functionality to update the workspace by selecting the externally created table. This update is expected to proceed without issues.

To edit a workspace, follow these steps:

1. Navigate to **Workspace Management** to display the **Workspace Summary** page. The page displays Workspace records in a table.
2. Click Action next to corresponding workspace and select **Edit** to edit the workspace.

Figure 6-26 Edit Workspace window
Note

You can modify the Workspace Type from Sandbox to Production or vice versa.

For more details on the Basic Details fields, see [Configure Basic Details](#) section.

Note

While updating a workspace, if any of the DMM operations such as Procedures, Functions, Packages, and so on fails, DMM operations alone will be rolled back and workspace will not be rolled back. You can verify the details in MMG_SANDBOX_DETAILS, MMG_SANDBOX_MASTER, MMG_SANDBOX_SCHEMA tables.

3. Click Save.**Note**

The **Details** option has been added, replacing Edit Workspace in Action list of workspace. In the Details screen, the Edit Workspace is available. The background concept has been added while adding/editing workspace. Once a user clicks on update option, the confirmation message will pop up and the user will be redirected to Sandbox Summary screen where the workspace will display as loading, indicating add/edit process is going on. Meanwhile, the user can proceed with other tasks and operations, instead of waiting for the workspace to be created or edited.

Note

The users should not perform manual modifications on the Target and Source Schema objects.

Currently, there are scenarios where in manual changes are performed directly on the Target Schema and this leads to failure when user performs Workspace Edit operation. Therefore, the user should ensure not to perform any manual modification to underlying database objects in Source and Target Schema.

Note

The Edit workspace icon with details has been added post launching the workspace.

7

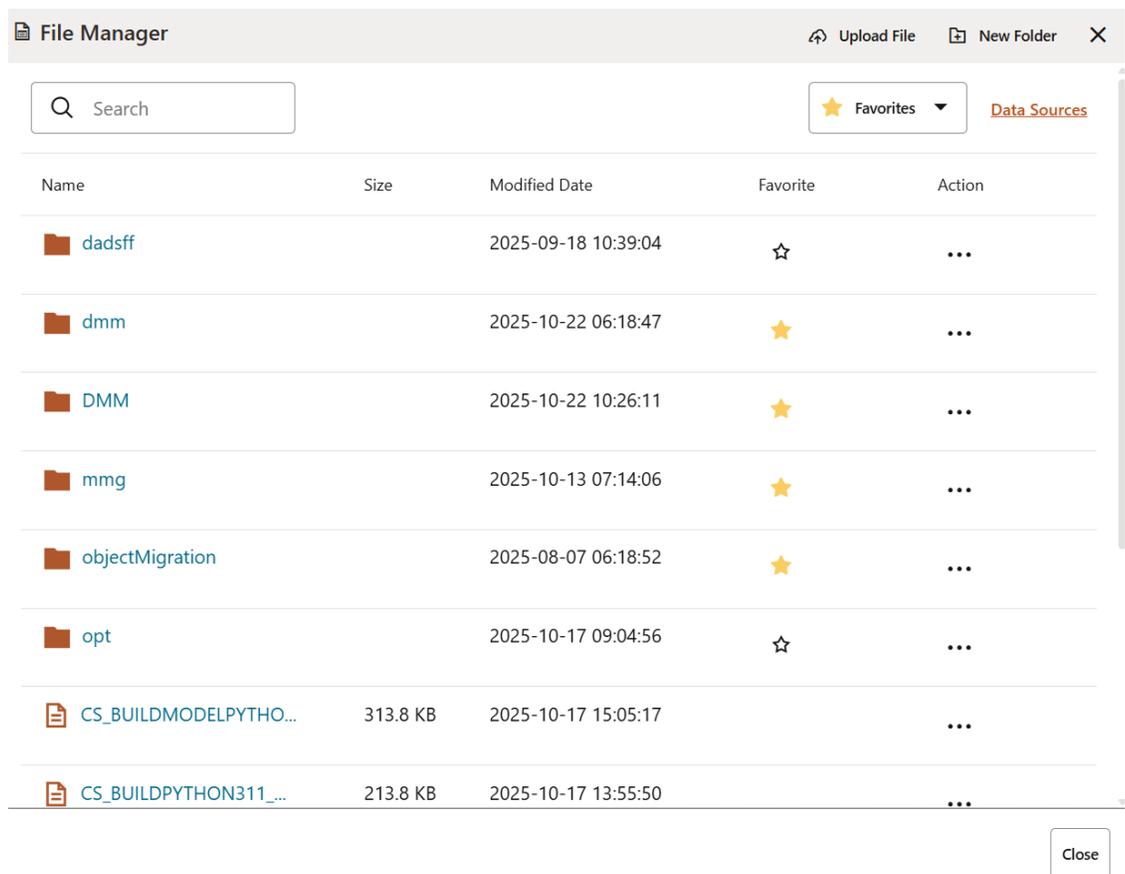
File Manager

File Manager is a new feature that is added in the **Workspace Summary** screen.

Note

File manager will always open the last opened directory from anywhere within the application.

Figure 7-1 File Manager



The screenshot shows the File Manager interface with a search bar, a Favorites dropdown, and a table of files and folders. The table has columns for Name, Size, Modified Date, Favorite, and Action. The files and folders listed are:

Name	Size	Modified Date	Favorite	Action
 dadsff		2025-09-18 10:39:04		...
 dmm		2025-10-22 06:18:47		...
 DMM		2025-10-22 10:26:11		...
 mmg		2025-10-13 07:14:06		...
 objectMigration		2025-08-07 06:18:52		...
 opt		2025-10-17 09:04:56		...
 CS_BUILDMODELPYTHO...	313.8 KB	2025-10-17 15:05:17		...
 CS_BUILDPYTHON311_...	213.8 KB	2025-10-17 13:55:50		...

Close

You must map the *File Admin User* to view it on the headers of the **Home** screen. The *File Admin User* has the following roles mapped to it:

- File Read
- File Delete
- File Write

The file manager stores the files and folders and replicates the data of the ftpshare folder from the deployed location. Users can use those stored files directly during the pipeline notebook and any other location where the file manager option is added. It has the feature to upload a new file, create a new folder, preview the file, and delete and download options. The user can view only those workspace folders to which he is mapped.

The favourites feature appears as a drop-down list on the **File Manager** page. It enables you to add a folder to the **Favourites** drop-down list by clicking the **Add to Favourites** icon adjacent to a folder. Click the **Add to Favourites** to remove a folder from the **Favourites** drop-down list. You can view all the folders that have been marked as a favourite by clicking the **Favourites** drop-down list. When a favourite is selected from the drop-down list, the application fetches the list of favorite folders and shows reason for it being a favourite as a tool tip.

Note

- If a folder that was added into the Favourites drop-down list was deleted, then it will appear as disabled and the hover over text will inform you that the directory does not exist.
- If the permissions or roles are revoked for a user in accessing a specified folder, then the user will not have access to that folder as it will be disabled for this user. The hover over text in this case will inform the user that the file is not accessible.

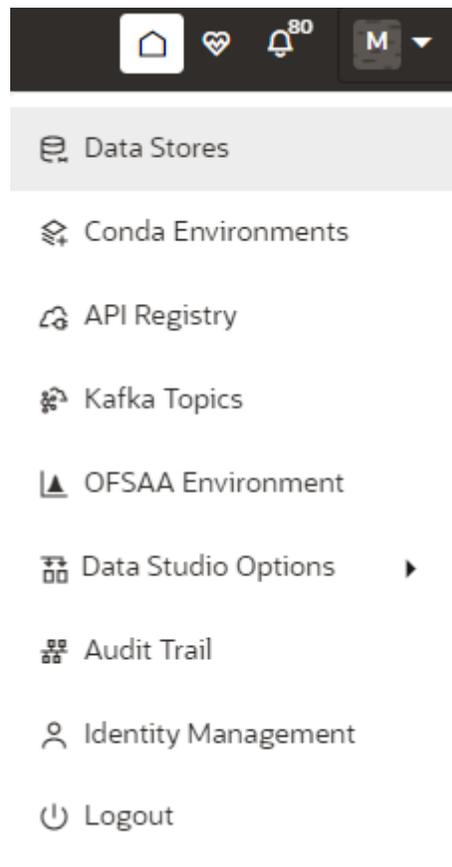
Additionally, workspace-level access controls are also applicable to files/folder. To access files/folders in python see the Access Module for File/Folder Operations section in [Notebook](#).

8

Administration

The Header displays icons, buttons, and text for generic information and access to the OFSAA application's features. The following user-interface elements are displayed.

Figure 8-1 Administration menu



- **User Name:** Displays the first letter of the logged-in user name. Hover over to view the complete name of the user. Click on this icon to select from the following options in the drop-down list:
- **Data Stores:** Click to navigate to Data Store Summary page. You can add and manage data stores from this page.
- **Conda Environments:** Click to navigate to Environment Summary page. You can configure and manage conda environments from this page.
- **API Registry:** Click to navigate to API Summary page. You can configure and manage APIs from this page.
- **Kafka Topics:** Click to navigate to Kafka Topics page. You can configure and manage kafka topics and clusters from this page.

- **OFSAA Environment:** Click to navigate to OFSAA Environment page. You can register OFSAA environments (Simulation and Production) from this page.
- **Data Studio Options:** Contains an underlying Notebook Server which has the following configurable options:
 - Interpreters
 - Tasks
 - Permissions
 - Credentials
 - Templates

Clicking on the option will navigate you to the specific page. For more details, see [Data Studio Options](#) section.

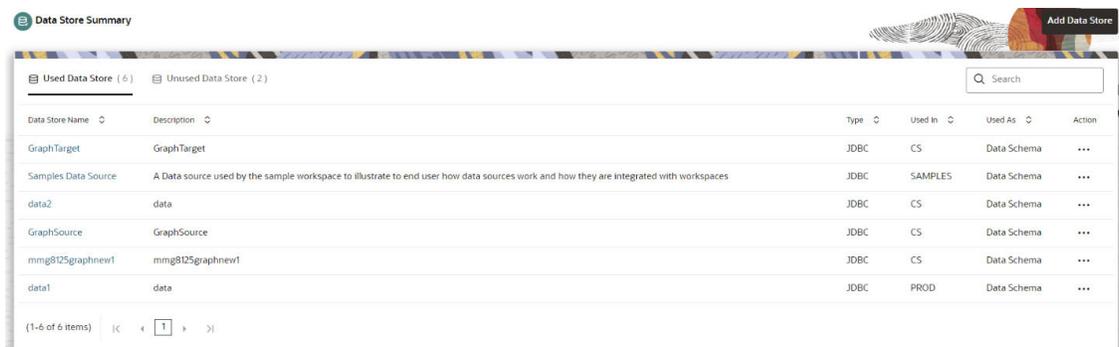
- **Admin:** Navigates to IDCS page.
- **Audit Trail:** Click to navigate to Audit Trail page The details such as start and stop of UI and Service, add or delete of datasource, API Registry, Kafka topics, Conda Environments and so on are displayed. You can view the sequence of actions performed in table view or timeline view (graphical representation) by clicking on Audit Table View, and Audit Timeline View options respectively. You can search the model by entering the name in the search box.
- **Identity Management:** Click to navigate to IDCS page. You can manage users, groups, roles, and functions from this page.
- **Logout:** Click to log out of the application.

Data Stores

This feature allows you to manage the Data Schemas registered with the application. The Data Store Summary window shows the list of Data schemas registered with the application. These Data schemas can be used either for workspace or for sourcing data. Click Data Stores to navigate to Data Source Summary window.

This window also allows you to manage these registered external sources.

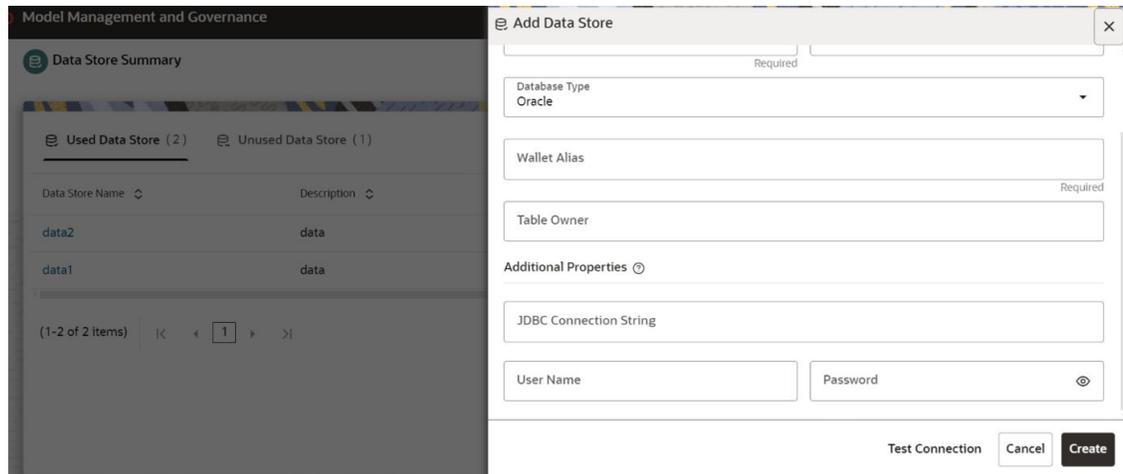
Figure 8-2 Data Source Summary window



The screenshot shows the 'Data Store Summary' window with a table of data sources. The table has columns for Data Store Name, Description, Type, Used In, Used As, and Action. There are 6 items in the table, with the first one selected.

Data Store Name	Description	Type	Used In	Used As	Action
GraphTarget	GraphTarget	JDBC	CS	Data Schema	...
Samples Data Source	A Data source used by the sample workspace to illustrate to end user how data sources work and how they are integrated with workspaces	JDBC	SAMPLES	Data Schema	...
data2	data	JDBC	CS	Data Schema	...
GraphSource	GraphSource	JDBC	CS	Data Schema	...
mimg8125graphnev1	mimg8125graphnev1	JDBC	CS	Data Schema	...
data1	data	JDBC	PROD	Data Schema	...

Figure 8-3 Add Data Store



The Data Store Summary consists of two sections: Used Data Source and Unused Data Source.

- **Used Data Source:**
 - This shows the list of Data Sources registered with any workspace. Here, you can only view the Data Source details. The count of Used Data Sources also displayed at the top of Data Source Summary page.
- **Unused Data Source:**
 - This shows the list of Data Sources those are not registered with any workspace. Here, you can only view, edit, or delete the Data Source details. The count of Unused Data Sources also displayed at the top of Data Source Summary page.

Note

Data Store name should not be allowed to be created with spaces at the start and end of Data Store name and with multiple spaces in between.

Adding a Data Store

To add a Data store, follow these steps:

1. Navigate to **Workspace Summary** window.
2. Click **Profile** icon and select **Data Stores**.
The **Data Store Summary** page is displayed.
3. Click **Add Store**.
The **Add Data Store** window is displayed.

Figure 8-4 Add Data Store Window

Add Data Store [Close]

Data Store Name Required

Description Required

Type
JDBC

Database Type
Oracle

Wallet Alias Test With Wallet Alias

Table Owner

Additional Properties ?

JDBC Connection String

User Name Password [Eye Icon]

Test With Properties

Cancel **Create**

4. Enter the required details as shown in the following table.

Table 8-1 Add Data Source

Field	Description
Data Store Name	Enter the connection URL to the database for the data schema.
Description	Enter the description of database connection.

Table 8-1 (Cont.) Add Data Source

Field	Description
Type	<p>Enter the type of the database connection. The supported types are JDBC and File.</p> <ul style="list-style-type: none"> • JDBC : If selected, the Database type options Oracle and Hive are displayed. • File: If selected, the following options are displayed. <p>Global: Select this option if you want to fetch the global level datasource details. You need to place the file with the datasource details in JSON format in the following location: <installed path>/ftpshare/mmg/Data Sources/datasources/##Data Source Name##/datafiles/##MISDATE##/*</p> <p>Workspace-limited: Select this option if you want to fetch the workspace level datasource details. You need to place the file with the datasource details in JSON format in the following location: <installed path>/ftpshare/mmg/##Workspace##/datasources/##Data Source Name##/datafiles/##MISDATE##/*</p> <p>Upload Data File: Click to upload data file in text or csv format. Any pre-filled values will be overridden with the selected data file. Once the file is uploaded, you can derive the schema which will auto populate the values based on the file.</p> <p>File Type : Select the Type as text or csv format.</p> <p>Escape Delimiter: An escape delimiter is simply a sequence that is recognized and ignored during parsing. Its purpose is to allow the use of escape sequences to embed byte sequences in data that would otherwise be seen as delimiter occurrences. This option differentiates multiple records in one single column from second column. Escape delimiter (eg. "), added in from of multiple records will treat it as multiple records for one single column. For example, if there is a normal delimiter "+" at a given level, and you define an escape delimiter "\+" as shown in the following figure, then aaa+b\+c+ddd will parse as three fields: aaa, b\+c, and ddd. If the escape delimiter were not defined, the sequence would then parse as four fields: aaa, b\, c, and ddd.</p> <p>Record Delimiter: There is a separation of the records using a delimiter character like a comma, semicolon, hyphen, and so on for the rows. Enter the delimiter in the Record Delimiter field. This is a mandatory field and limited to two characters.</p> <p>Field Delimiter: There is a separation of the records using a delimiter character for the columns. Enter the delimiter in the Field Delimiter field. This is a mandatory field.</p> <p>Derive Schema: It is an optional feature that populates the data grid by analyzing schema details from any data file stored in the above</p>

Table 8-1 (Cont.) Add Data Source

Field	Description
	<p>specified directory. This is applicable when you upload data file in previous steps.</p> <p>You can either add the file details using data template or manually.</p> <p>Click Data File Template icon to select the Data Source entities and click Save.</p> <p>OR</p> <p>Click Add icon to add the details such as Physical Name, Logical Name, Data Type, and Field Order manually and click Save.</p> <p>Click Reorder Grid option to reorder the field orders.</p>

Table 8-1 (Cont.) Add Data Source

Field	Description
Database Type	<p>If the Type field is selected as JDBC, the following options are displayed. Select the Database Type as Oracle or Hive. NOTE: Selected tables during Hive sourcing should be preexisting in the RDBMS data schema before the workspace population. If you select Database Type as Oracle (see Figure 10), then following additional fields are displayed to enter details:</p> <ul style="list-style-type: none"> • Wallet Alias : Enter the Wallet Alias. This value should be same as configured using Oracle Wallet. For more information, see the Oracle Financial Services Model Management and Governance Installation Guide • Table Owner: Enter the Oracle Database schema name. <p>If you select Database Type as Hive (see Figure 11), then following additional fields are displayed to enter details:</p> <p>User Name: User Name / Principal is used for Kerberos authentication. Example: mmg/hostname@ORACLE.COM. Table Owner: Enter the Hive schema. JDBC Connection String: Enter the JDBC Connection String. Example: jdbc:hive2://hostname:10000/default;principal=hive/hive-service-hostname@ORACLE.COM. JDBC Driver: Supports org.apache.hive.jdbc.HiveDriver and com.cloudera.hive.jdbc4.HS2Driver. Keytab File Name: Enter the Name of the keytab file present in conf directory. Example: mmg.keytab Realm File Name: Enter the Name of the configuration file present in conf directory. Example: krb5.conf NOTE:</p> <ul style="list-style-type: none"> • Schema population for Hive as target is not supported. • This is applicable only for Sandbox Workspace. <p>For more information on setting the environment, see the Oracle Financial Services Model Management and Governance Installation Guide.</p>

5. Click Create.

The Data Store is created.

Note

Click **Test Connection** to check the connection. A success message is displayed.

Note

Data Store needs to be created with additional properties giving JDBC connection string, username and password.

The user should not provide '/' for the JDBC connection string. For example:
jdbc:oracle:thin:@//sevicehost:port/servicename

The correct JDBC connection string which the user should provide is:
jdbc:oracle:thin:@sevicehost:port/servicename

The reason because AAI does not accept the string with //, thereby leading to workspace creation failure.

Viewing the Data Store Details

To view the Data Store details, click the **Action** icon next to corresponding Workspace and select **View**.

Figure 8-5 View Data Store

The screenshot shows a 'View Data Store' dialog box with the following fields:

- Data Store Name:** GraphTarget
- Description:** GraphTarget
- File Availability:** JDBC
- Database Type:** Oracle
- Wallet Alias:** graph_als
- Table Owner:** mmg8125graphnew1

Conda Environments

Conda as a package manager helps you to find and install packages. With the capability of environment manager, you can set up a totally separate environment to run different versions of Python. In addition, you can continue to run your usual version of Python in your normal environment. You can configure the channels required for the Conda environment creation in the .condarc file.

Note

The supported version of Conda is 3.9.x.

You must add the miniconda/bin directory in the following path variable as shown below:

PATH=/scratch/ofsaadb/miniconda3/bin:\$PATH

For proxy support, you need to manually configure the .condarc file present in the root directory.

Limitation: Conda support currently does not include linking externally created environments to the application.

To configure:

Add the below in the .condarc file.

proxy_servers:

- http: http://www-proxy.idc.oracle.com:80
- https: http://www-proxy.idc.oracle.com:80

Note

- Conda package search is restricted only to the package name and does not include versions.
- Python version displayed in the Register Conda UI.

Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application. You can register and manage Conda environments from this page.

Figure 8-6 Environment Summary page

Name	Python Version	Creation Status	Registered By	Registration Date	Modified By	Modified Date	Action
Python3 Python3 virtual env ...	3.9.16	[Inspect]	MMGSAML	8 May 2023, 12:30:56	N.A.	N.A.	...
Python3.16 Fraud reporting ...	3.9.16	[Inspect]	MMGSAML	8 May 2023, 12:31:49	N.A.	N.A.	...
Test Test ...	3.9.15	[Create]	MMGSAML	8 May 2023, 12:52:42	N.A.	N.A.	...

The following table provides descriptions for the fields and icons on the **Environment Summary** page.

Table 8-2 Fields and icons on the Sandbox Summary page

Field or Icon	Description
Search	The field to search for a Conda environment. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results. NOTE: Conda package search is restricted only to the package name and does not include versions.
Name	The name of the Conda environment.
Python Version	The python library version of the Conda environment.
Creation Status	The status of the Conda environment. The status can be either Inspect or Create. <ul style="list-style-type: none"> Inspect: The environment is registered and created. You can perform functions such as Edit/View/Inspect/Clone/Export/Remove for the Inspect status environments. Create: The environment is registered but not created. You can perform functions such as Create/Edit/View/Deregister for the Create status environments
Registered By	The User Id of the User who registered the Conda environment.
Registration Date	The date on which the Conda environment has been registered.
Modified By	The ID of the Last Modified by user who has modified the Conda environment.
Modified Date	The date on which the Conda environment was modified.
Register Environment	Click to register a new Conda environment.
Conda Settings	Click to add or modify the Conda components settings.
Conda Properties	Click to view the Conda properties.
Action	Click the three dots to perform Edit/View/Inspect/Clone/Export/Remove functions on selected Conda environment. The functions might vary based on the selected Conda environment.

Register a Conda Environment

To register a Conda environment:

1. Click **Conda Environments** to navigate to the **Environment Summary** page from any window in the application.
This page displays the Conda environments.
2. In the **Environment Summary** screen, click **Register Environment**.
The **Register Environment** page is displayed.

Figure 8-7 Register Environment Window

Register Environment

Name Required

Description Required

Environment Tags

Location
/scratch/ofsaobie/miniconda3/envs/

Package
Python

Version Required

Import YML File

Cancel Register

3. Enter the **Name** for the environment. This field is mandatory.
4. Enter the **Description** for the environment. This field is mandatory.
5. Enter a tag in the **Environment Tags** field. You can add multiple tags.

Note

The **Location** of the Conda environment is displayed. This field is greyed out and cannot be modified.

6. The **Location** and **Package** fields are populated by default.
7. Click the **Version** drop-down list to select a Python version. Different Conda environments can be created by using different Python versions.
8. Click the Import YML slider to import a **.yaml/.yml** file. You must ensure that the environment name and Python versions that have been selected in the **Version** field match those specified in the imported **.yaml/.yml** file.

Package
Python

Version ▼
Required

Import YML File

Import File

Users to verify that the environment name and Python version chosen above precisely correspond to those specified in the imported .yaml/.yml file to avoid discrepancies.

↑
Import Archive File
Drag & Drop file here

9. Click the **Import Archive File** or drag and drop the desired **.yaml/.yml** file from your local system.
10. Click **Register**.
The Conda environment is registered and displayed in the Environment Summary page.
After the environment registration is complete, you must create the Conda environment.
For more details, see the [Create a Conda Environment](#) section.

Create a Conda Environment

To create a Conda environment:

1. Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application.
This page displays the Conda environments in a table.
2. Click **Action** next to corresponding environment and select **Create** to create the Conda environment.

Note

The **Create** option is not displayed for the Conda environment with the creation status **Inspect**.

A confirmation dialog is displayed.

3. Click **Create**.

The Conda environment is created.

These environments can be selected while creating a Workspace. For more details, see the [Workspace Schema](#) section.

In addition, these environments can be selected during the Model pipeline execution.

Edit a Conda Environment

To edit a Conda environment:

1. Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application.

This page displays the Conda environments in a table.

2. Click **Action** next to corresponding environment and select **Edit** to edit the details.

The **Edit Environment** page is displayed.

For more details on the Edit environment fields, see [Register a Conda Environment](#) section.

Note

The **Name** of the Conda environment is greyed out and cannot be modified.

Figure 8-8 Edit Environment page

✎ Edit Environment ✕

Basic Information

Name
 Document

Description
 enhanced version

Environment Tags

Location
 /scratch/ofsaadb/miniconda3/envs/Document

Advanced Features

Package
 Python

Version
 3.9.16 (Recommended) ▼

Cancel

Update

3. Click **Update**.

The details of the Conda environment is updated.

View a Conda Environment

To view a Conda environment details, follow these steps:

1. Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application.

This page displays the Conda environments in a tabular format.

2. Click **Action** next to corresponding environment and select **View** to view the details.

The details of the Conda environment are displayed.

Figure 8-9 View a Conda Environment

View Environment ✕

Basic Information

Name
Python3

Description
Python3 virtual env

Environment Tags
conda python

Location
/scratch/ofsaadb/miniconda3/envs/Python3

Advanced Features

Package
Python

Version
3.9.16 (Recommended) ▼

Deregister a Conda Environment

To Deregister a Conda environment:

1. Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application.

This page displays the Conda environments in a table.

2. Click **Action** next to corresponding environment and select **Deregister** to deregister the Conda environment.

Note

The **Deregister** option is not displayed for the Conda environment with the creation status **Inspect**. A confirmation dialog is displayed.

3. Click **Deregister**.

The registered Conda environment is deregistered and removed from the Environment Summary page.

Inspect a Conda Environment

To inspect a Conda environment:

1. Click **Conda Environments** to navigate to **Environment Summary** page from any other window in the application.

This page displays the Conda environments in a table.

2. Click **Action** next to corresponding environment and select **Inspect** to edit the details.**Note**

The **Inspect** option is not displayed for the Conda environment whose creation status is **Create**.

The Conda environment details are displayed.

Figure 8-10 Conda environment details

The screenshot shows a web interface for a Conda environment. At the top, there are buttons for 'Remove', 'Clone', and 'Export'. Below these, there is a search bar and a table of installed packages. The table has columns for Name, Version, Platform, Base URL, Channel, and Action. The packages listed are: _libgcc_mutex (0.1), _openmp_mutex (4.5), bzip2 (1.0.8), ca-certificates (2023.5.7), and ld_impl_linux-64 (2.40). All packages are from the 'conda-forge' channel and are on the 'linux-64' platform.

Name	Version	Platform	Base URL	Channel	Action
_libgcc_mutex	0.1	linux-64	https://conda.anaconda.org/conda-forge	conda-forge	...
_openmp_mutex	4.5	linux-64	https://conda.anaconda.org/conda-forge	conda-forge	...
bzip2	1.0.8	linux-64	https://conda.anaconda.org/conda-forge	conda-forge	...
ca-certificates	2023.5.7	linux-64	https://conda.anaconda.org/conda-forge	conda-forge	...
ld_impl_linux-64	2.40	linux-64	https://conda.anaconda.org/conda-forge	conda-forge	...

You can perform the following in the above page:

- [Remove Environment](#)
- [Clone Environment](#)
- [Export Environment](#)
- [Edit a Conda Environment Description](#)
- [Add or Remove the Environment Tags](#)
- [Search All Packages](#)
- [View Package Properties](#)

- [Update Package](#)
- [Remove Package](#)

Remove Environment

To remove a Conda environment:

1. In the Conda details page, click **Remove**.

A confirmation message is displayed.

2. Click **Remove**.

The Conda environment is deregistered and removed from the **Environment Summary** page.

Clone Environment

To clone a Conda environment:

1. In the Conda details page, click **Clone**.

The **Clone Environment** page is displayed.

Figure 8-11 Clone Environment

Clone Environment ×

All packages of "Python3" will be cloned into the target environment. 

Target Environment Name Required

Target Environment Description Required

Environment Tags

Show Command 

 **Note**

You can preview the packages of the current Conda environment by clicking the **Preview** icon.

2. Enter the **Target Environment Name** to which the current environment needs to be cloned.
3. Enter the **Target Environment Description**.

- Enter a tag in the **Environment Tags** field. You can add multiple tags.
- Click **Clone**.
The Conda environment is cloned to the targeted environment.

Export Environment

To export a Conda environment:

- In the Conda details page, click **Export**.
The Conda environment is exported in **yml** format.

Edit a Conda Environment Description

To edit a Conda environment description:

- In the Conda details page, click **Edit** icon.

Figure 8-12 Conda details page



- Modify the description and click **Save**.

Note

The **Name** of the Conda environment cannot be modified.

The details of the Conda environment is updated.

Add or Remove Environment Tags

To add or remove Environment Tags:

- In the Conda details page, click **Edit**.

Figure 8-13 Conda details page



2. Add or remove the environment tags, and click **Save**.

The environment tag details of the Conda environment are updated.

Search All Packages

To search all the packages:

1. In the Conda details page, enable the **Search all packages** option.
2. If you want to search the packages in cache, enable the Look up in cache option.

This option is set to true by default for faster processing. The searched package is looked up in the cache and the result published, if found. If not, a new search in the conda-forge channel is performed. You can toggle this option if you expect more recent versions to be available.

3. Enter the package name in the search bar and press **Enter**.

All the packages meeting your search criteria are displayed.

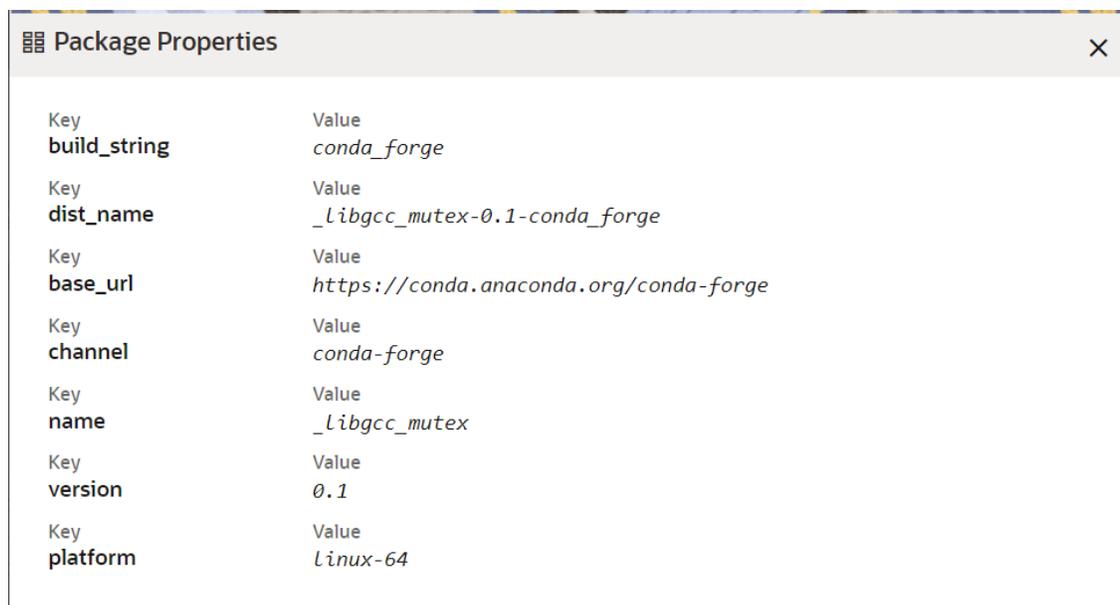
Click the **Action** icon to view and install packages.

View Package Properties

To view package properties of the Conda environment, on the Conda Details page, click **Action** and select **View**.

The details of the package are displayed.

Figure 8-14 Package Properties



Key	Value
build_string	<i>conda_forge</i>
dist_name	<i>_libgcc_mutex-0.1-conda_forge</i>
base_url	<i>https://conda.anaconda.org/conda-forge</i>
channel	<i>conda-forge</i>
name	<i>_libgcc_mutex</i>
version	<i>0.1</i>
platform	<i>Linux-64</i>

Update Package

To update the package of the Conda environment:

1. In the Conda details page, click **Action** and select **Update**.

A confirmation message is displayed.

2. Click **Update**.

The selected package is updated.

Remove Package

To remove a package from the Conda environment:

1. In the Conda details page, click **Action** and click **Remove**.

A confirmation message is displayed.

2. Click **Remove**.

The selected package is removed.

Clone Environment

To clone a Conda environment:

1. In the Conda details page, click **Clone**.

The **Clone Environment** page is displayed.

Figure 8-15 Clone Environment page

Clone Environment ×

All packages of "Python3" will be cloned into the target environment.

Target Environment Name Required

Target Environment Description Required

Environment Tags

Show Command ⓘ

Note

You can preview the packages of the current Conda environment by clicking the Preview icon.

2. Enter the **Target Environment Name** to which the current environment needs to be cloned.
3. Enter the **Target Environment Description**.

4. Enter a tag in the **Environment Tags** field. You can add multiple tags.
5. Click **Clone**.
The Conda environment is cloned to the targeted environment.

Export Environment

To export a Conda environment:

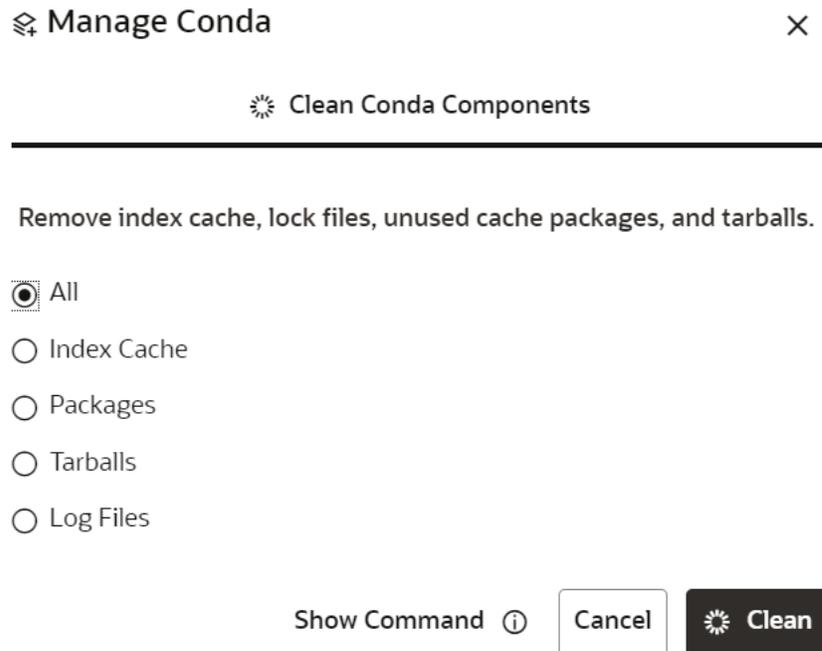
1. Click the **Conda Environments** icon to navigate to the **Environment Summary** page from any other window in the application.
This page displays the Conda environments in a tabular format.
2. Click **Action** and select **Export**.
The Conda environment is exported in **yml** format.

Modify Conda Settings

To modify the Conda Settings:

1. Click the **Conda Environments** icon to navigate to the **Environment Summary** page from any other window in the application.
This page displays the Conda environments in a tabular format.
2. Click the **Conda Settings** icon.
The **Manage Conda** screen is displayed.

Figure 8-16 Manage Conda screen



3. Select the required option and click **Clean**.
 - **All**: Removes index cache, lock files, unused cache packages, and tarballs.

- **Index Cache:** Removes index cache.
- **Packages:** Removes unused packages from writable package caches. **WARNING:** This does not check for packages installed using symlinks back to the package cache.
- **Tarballs:** Removes cached package tarballs.
- **Log Files:** Removes log files.

This will clean up the Conda components based on the selection.

View Conda Properties

To view Conda Properties:

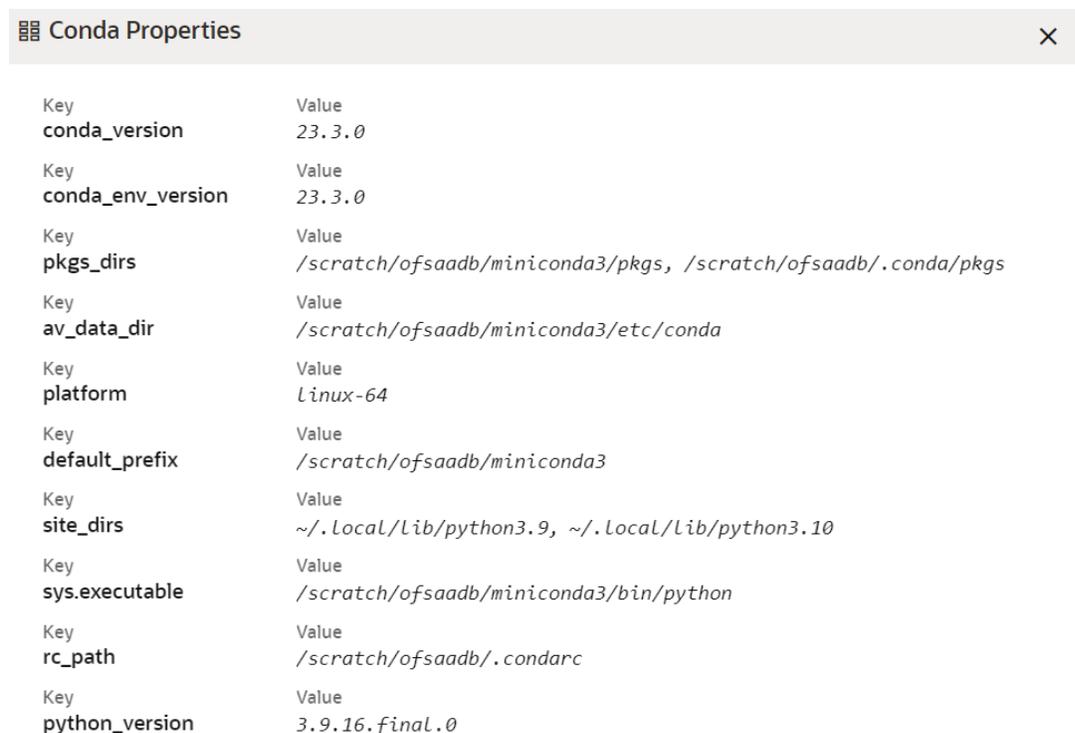
1. Click the **Conda Environments** icon to navigate to the **Environment Summary** page from any other window in the application.

This page displays the Conda Environments in a tabular format.

2. Click the **Conda Properties** icon.

The Conda properties are displayed in a key-value format.

Figure 8-17 Conda Properties



Key	Value
conda_version	23.3.0
conda_env_version	23.3.0
pkgs_dirs	/scratch/ofsaadb/miniconda3/pkgs, /scratch/ofsaadb/.conda/pkgs
av_data_dir	/scratch/ofsaadb/miniconda3/etc/conda
platform	linux-64
default_prefix	/scratch/ofsaadb/miniconda3
site_dirs	~/Local/lib/python3.9, ~/Local/lib/python3.10
sys.executable	/scratch/ofsaadb/miniconda3/bin/python
rc_path	/scratch/ofsaadb/.condarc
python_version	3.9.16.final.0

Using the .condarc Conda Configuration File

Overview

Overview The Conda Configuration file, `.condarc`, is an optional runtime configuration file that allows advanced users to configure various aspects of Conda, such as which channels it searches for packages, proxy settings, and environment directories.

Note

A `.condarc` file can also be used in an administrator-controlled installation to override the users' configuration. See [Administering a multi-user Conda Installation](#).

The `.condarc` file can change many parameters, including:

- Where Conda looks for packages.
- If and how Conda uses a proxy server.
- Where Conda lists known environments.
- Whether to update the Bash prompt with the currently activated environment name.
- Whether user-built packages should be uploaded to [Anaconda.org](#).
- What default packages or features to include in new environments.

Creating and Editing

The `.condarc` file is not included by default, but it is automatically created in your home directory the first time you run the Conda Config command. To create or modify a `.condarc` file, open a terminal and enter the `conda config` command. The `.condarc` configuration file follows simple [YAML syntax](#).

Example:

```
conda config --add channels conda-forge
```

Alternatively, you can open a text editor such as Notepad on Windows, TextEdit on MacOS, or VS Code. Name the new file `.condarc` and save it to your user home directory or root directory. To edit the `.condarc` file, open it from your home or root directory and make edits in the same way you would with any other text file. If the `.condarc` file is in the root environment, it will override any in the Home Directory.

You can find information about your file by typing `Conda info` in your terminal. This will give you information about your `.condarc` file, including where it is located.

You can also download a [sample .condarc file](#) to edit in your editor and save to your user Home Directory or Root Directory.

Example:

To set the `auto_update_conda` option to `False`, run:

```
conda config --set auto_update_conda False
```

For a complete list of Conda Config commands, see the [command reference](#). The same list is available at the terminal by running `conda config --help`. You can also see the [Conda Channel Configuration](#) for more information.

Conda supports a wide range of configuration options. This page gives a non-exhaustive list of the most frequently used options and their usage. For a complete list of all available options for your version of conda, use the `conda config --describe` command.

Searching for .condarc

Conda looks in the following locations for a `.condarc` file:

XDG_CONFIG_HOME is the path to where user-specific configuration files should be stored defined following The XDG Base Directory Specification (XDGBDS). Default to \$HOME/.config should be used. CONDA_ROOT is the path for your base conda install. CONDA_PREFIX is the path to the current active environment. CONDARC must be a path to a file named .condarc, condarc, or end with a YAML suffix (.yaml or .yml).

Note

Any condarc files that exist in any of these special search path directories need to end in a valid yaml extension (".yaml" or ".yml").

Conflict Merging Strategy

When conflicts between configurations arise, the following strategies are employed:

- Lists-merge
- Dictionaries-merge
- Primitive-clobber

Precedence

The precedence by which the Conda Configuration is built out is shown below. Each new arrow takes precedence over the ones before it. For example, config files (by parse order) will be superseded by any of the other configuration options. Configuration environment variables (formatted like CONDA_<CONFIG NAME>) will always take precedence over the other 3.

Obtaining Information from the .condarc file

You can use the following commands to get the effective settings for conda. The effective settings are those that have merged settings from all the sources mentioned above.

To get all keys and their values:

```
conda config --get
```

To get the value of a specific key, such as channels:

```
conda config --get channels
```

To show all the configuration file sources and their contents:

```
conda config --show-sources
```

Saving Settings to your .condarc file

The .condarc file can also be modified via conda commands. Below are several examples of how to do this.

To add a new value, such as <http://conda.anaconda.org/mutirri>, to a specific key, such as channels:

```
conda config --add channels http://conda.anaconda.org/mutirri
```

To remove an existing value, such as <http://conda.anaconda.org/mutirri> from a specific key, such as channels:

```
conda config --remove channels http://conda.anaconda.org/mutirri
```

To remove a key, such as channels, and all of its values:

```
conda config --remove-key channels
```

To configure channels and their priority for a single environment, make a `.condarc` file in the [root directory of that environment](#).

Sample `.condarc` file

Because the `.condarc` file is just a YAML file, it means that it can be edited directly. Below is an example `.condarc` file:

API Registry

Using API Registry, you will be able to register APIs above workspace level. Configured APIs can be mapped to multiple workspaces and the mapped APIs will be available for use in model pipelines.

Currently, **No Auth**, **Token Based** and **Basic Auth** authentications are supported. Additionally, body parameters can only be provided with text data. All values that needs to be replaced during run time can be given in the format '<<placeholder variable name>>'.

Configured API can be viewed, edited (API code is not editable) and deleted. APIs that are not mapped to any workspace can only be deleted.

Integrating API with Model Pipelines

Mapped APIs will be available for use in model pipelines for the workspace. By selecting a specified API from the dropdown, you can link a node with an API. Additionally, you can provide an output variable name which will be assigned with the response from the API when that node is executed.

Note

API Registry is available only for %python interpreters. Output variable will be a python variable accessible only for paragraphs with %python interpreters.

In pipeline execute screen, values can be entered for all the placeholder variables during run time.

Note

Since variables are fetched during run time, it will be available in notebook or execute screen only after the first execution. This is the default behavior of all programmatic dynamic forms in Datastudio.

On executing the node/pipeline, result of the API executed can be accessed from the variable given under 'Output_variable_name'.

When the pipeline is imported to another workspace, it will seamlessly use the API that is mapped with the target workspace in the same order as that of the API mapped in source workspace.

Click **API Registry** to navigate to **API Summary** page from any other window in the application. You can configure and manage APIs from this page.

Figure 8-18 API Summary page

API Code	Description	Created By	Created Date	Modified By	Modified Date	Action
create API	create	MMGTEST	2023-03-08 11:37:37	MMGTEST	2023-03-08 11:39:15	...
add	add	MMGTEST	2023-03-08 11:38:52	MMGTEST	2023-03-08 11:38:52	...
GetModel	get method for model	MMGTEST	2023-03-08 12:00:35	MMGTEST	2023-03-08 12:00:35	...

The following table provides descriptions for the fields and icons on the **API Summary** page.

Table 8-3 Fields and icons on the Sandbox Summary page

Field or Icon	Description
Search	The field to search for an API. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.
API Code	The code of the API.
Description	The description for the API.
Created By	The User Id of the User who created the API.
Created Date	The date on which the API was created.
Modified By	The ID of the Last Modified by user who has modified the API.
Modified Date	The date on which the API was modified.
Configure API	Click Configure API to create a new API.
Action	Click the three dots to perform View/Edit/Delete functions on selected API.

Configure an API

To configure an API:

1. Click **API Registry** to navigate to **API Summary** page from any other window in the application.

API Summary Configure API

API Code	Description	Created By	Created Date	Modified By	Modified Date	Action
API45		MMGADMIN	2025-10-16 13:41:34	N.A.	N.A.	...
ZSFXD5FD		MMGADMIN	2025-10-15 07:10:41	N.A.	N.A.	...
APITEST		MMGADMIN	2025-10-16 11:46:43	MMGADMIN	Oct 16, 2025, 7:04:52 PM	...
APIBaicauth		MMGADMIN	2025-10-16 13:47:03	N.A.	N.A.	...
asdsd		MMGADMIN	2025-09-26 05:21:36	N.A.	N.A.	...
fgbfchbgf		MMGADMIN	2025-10-15 13:38:02	N.A.	N.A.	...
API1		MMGADMIN	2025-09-18 10:18:19	MMGADMIN	Oct 16, 2025, 5:07:02 PM	...

(1-7 of 7 items) | < < 1 > >

This page displays the API records in a tabular format.

- Click the **Configure API** button.
The **Configure API** page is displayed.

Figure 8-19 Configure API Window

Configure API ✕

Required

REST

POST

Required

*Placeholders for run time variables can be given in the format '<placeholder variable name>'.
Placeholders for python variables can be given in the format '<var:variable name>'.*

Headers Params Body Authorization

Key	Value	✕
-----	-------	---

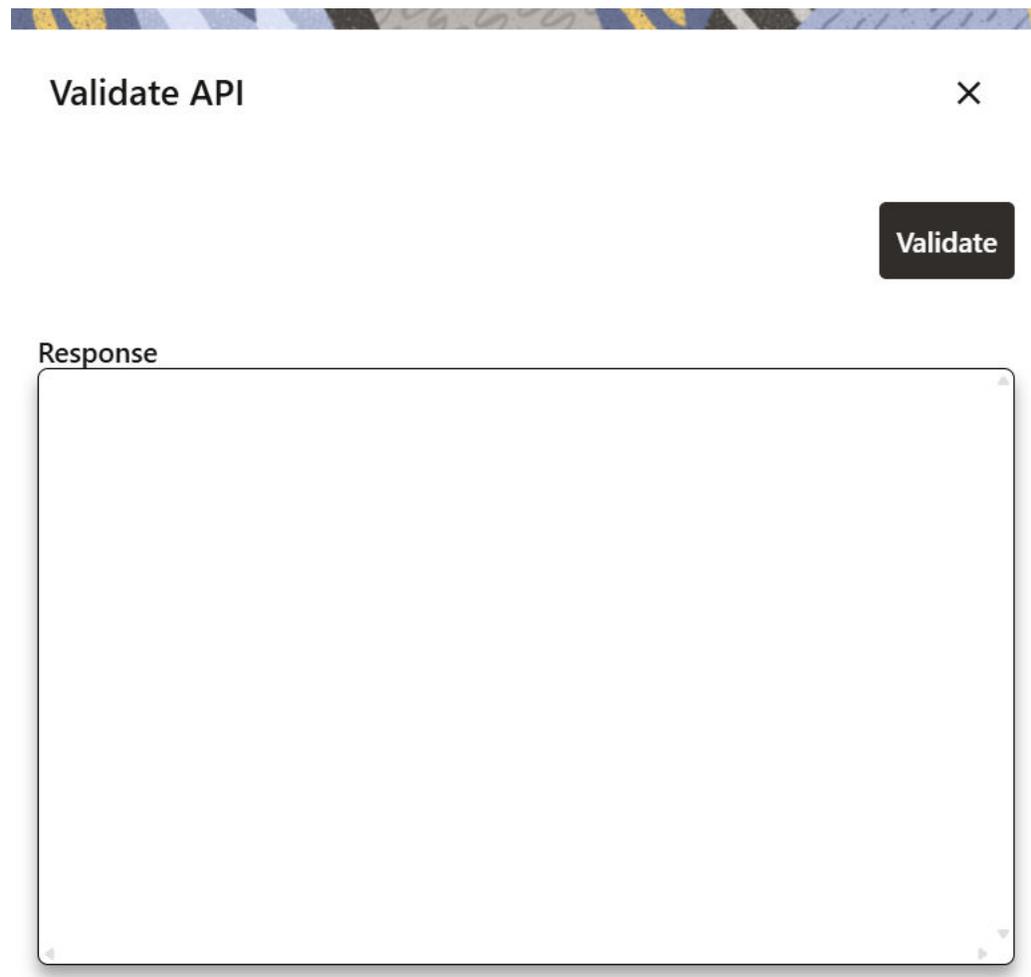
+ Add More

Validate
Cancel
Add

3. Enter the **Code** for the API. This field is mandatory.
4. Enter the **Description** for the API.
5. In the **Request Type** drop-down list, select the required option. Currently, REST type is supported.
6. In the **Method Type** drop-down list, select the required option. The supported types are GET, POST, PUT, and DELETE.
7. Enter the **URL** for the API. This field is mandatory.
8. You can provide the details such as Headers, Params, Body, and Authorization under the respective fields.

In the **Authorization** tab, the **Auth Type** drop-down list contains the following options:

- No Auth
 - Token Based
 - Basic Auth - This option enables you to add a basic authentication, username and password.
9. Click **Validate** to open the **Validate API** window and to validate the API details.



You are prompted to add the values for all the placeholder variables. After entering the values for placeholders, click the **Validate** button to view the API responses.

10. Click Add.

The API is added and displayed in the API Summary page. These API's can be selected while creating a Workspace. For more details, see the [Workspace Schema](#) section.

View an API Registry

To view an API Registry:

1. Click **API Registry** to navigate to **API Summary** page from any other window in the application.

This page displays the API records in a tabular format.

2. Click **Action** and select **View** to view the API.

The details of the API is displayed.

Figure 8-20 View API Details

View API Details

Code
create API

Description
create

Request Type
REST

Method Type
POST

Uri
https://slc07oyt.us.oracle.com:3661/mmg8124/v1/datasource-servi

Placeholders can be given in the format '<<placeholder variable name>>'

Headers Params Body Authorization

Key locale Value <<locale>>

Key user Value <<user>>

3. Navigate to the Headers, Params, Body, and Authorization to view the respective details.

Edit an API Registry

To edit an API configuration:

1. Click **API Registry** to navigate to **API Summary page** from any other window in the application.

This page displays the API records in a table.

2. Click Action next to corresponding API and select **Edit** to edit the API.

The **Edit API Details** page is displayed.

Figure 8-21 Edit API Details page

Code
add

Description
add

Request Type
REST

Method Type
POST

Url
https://slc07oyt.us.oracle.com:3662/mmg8124/v1/datasource-servi

Placeholders can be given in the format '<<placeholder variable name>>'

Headers Params Body Authorization

Key	Value	
locale	<<locale>>	🗑️
user	<<user>>	🗑️
Key	Value	🗑️

Validate Cancel Save

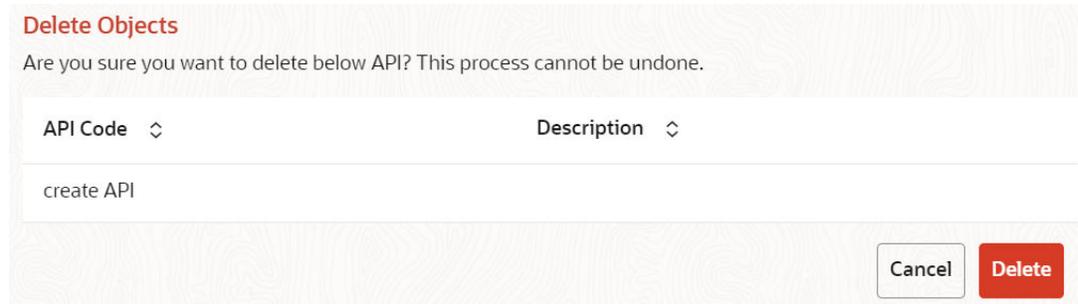
For more details on the Edit API Details fields, see [Configure an API](#) section.

3. Click **Save**.

Delete an API Registry

To delete an API Registry:

1. Click **API Registry** to navigate to **API Summary** page from any other window in the application.
This page displays the API records in a tabular format.
2. Click Action and select **Delete**.

Figure 8-22 Delete Objects screen**3. Click Delete.**

The API Registry is deleted.

Kafka Topics

Kafka is a distributed data source which is used to create real-time data pipelines and allows you to decouple data streams and systems.

Click **Kafka Topics** icon to navigate to **Kafka Topics** page from any other window in the application. You can configure and manage Kafka topics and clusters from this page.

Figure 8-23 API Summary page

Topic Name	Cluster Name	Description	Created Date	Action
1hp7	cluster1	1hp7	2023-04-11 10:47:58	...
2hp7	cluster1	2hp	2023-04-11 10:49:01	...

The following table provides descriptions for the fields and icons on the **Kafka Topics** page.

Table 8-4 Fields and icons on the Sandbox Summary page

Field or Icon	Description
Search	The field to search for a Kafka topics and clusters. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.
Topic Name	The topic name of the Kafka.
Cluster Name	The cluster name of the Kafka.
Description	The description of the Kafka.
Created Date	The date on which the API was created.
Kafka Clusters	Click Kafka Clusters to view/create/delete Kafka Clusters.
Register API	Click Register API to create a new Kafka topic.
Action	Click the three dots to perform View/Edit/Delete functions on selected Kafka topic.

Create a Kafka Cluster

To create a Kafka Cluster:

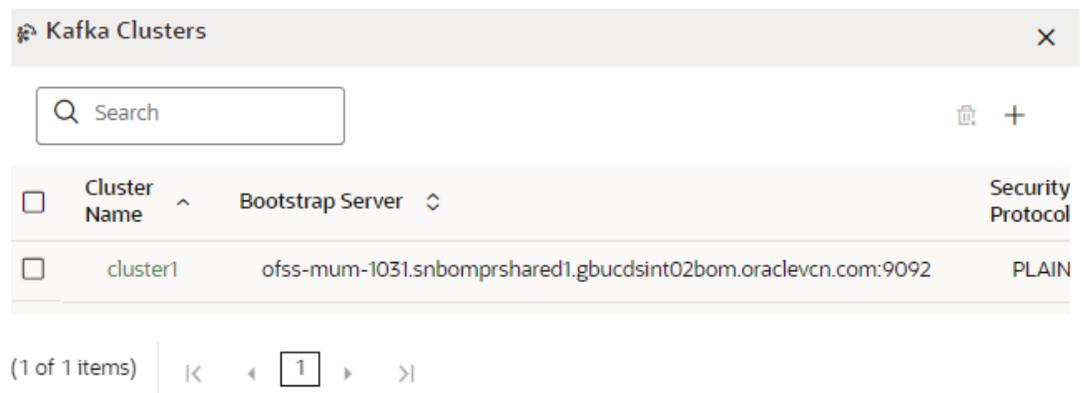
1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Kafka Clusters** option.

The **Kafka Clusters** page is displayed.

Figure 8-24 Kafka Clusters page

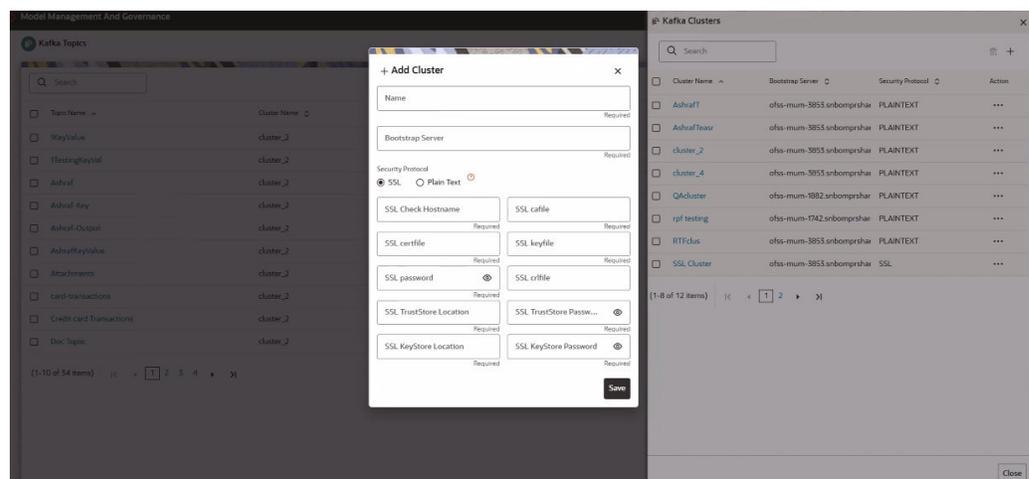


3. Click **Add**.

The Add Cluster window is displayed.

4. Enter the **Cluster Name**. This field is mandatory.
5. Enter the **Bootstrap server** details. This field is mandatory.
6. Select the **Security Protocol** as:
 - **Plain Text**
 - OR
 - **SSL**

Figure 8-25 SSL Type Cluster



Populate the mandatory fields. For detailed reference click the **Help** icon.

Note

In the current release:

- The Kafka feature is not supported in Solaris OS.
- Kafka is not supported with Python version 3.12 onwards. This is because pyflink cannot be downloaded from this version.

7. Click **Save**.

The Kafka cluster is created, and this will be displayed while registering the Kafka Topic.

Delete a Kafka Cluster

To delete a Kafka Cluster:

1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Kafka Clusters** option.

The **Kafka Clusters** page is displayed.

3. Select the Kafka Cluster which you want to delete and click **Delete**.

A confirmation message is displayed.

4. Click **Delete**.

The Kafka Cluster is deleted.

Register a Kafka Topic

To register a Kafka Topic:

1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Register Topic** button.

The Register Topic page is displayed.

Figure 8-26 Register Topic page

Register Topic [X]

Basic Information

Name Required

Description Required

Cluster Name Required + Kafka Topic ID Required

Record Schema (i)

Order	Name	Type	Action
Order *	Name *	Type *	✓ ✕

No data to display.

3. Enter the **Kafka Topic Name**. This field is mandatory.
4. Enter the **Description** for the Kafka Topic. This field is mandatory.
5. In the **Cluster Name** drop-down list, select the required option and enter a unique Kafka Topic ID. This field is mandatory.
6. In the **Record Schema** field, enter the order, name, and the data type for the Kafka topic.

Note

Currently, **PyFlink**, **Python**, and **Java Schema** types are supported. For more details, click the icon next to Record Schema field.

7. Click **Register**.

The Kafka topic is added and displayed in the Kafka Topics page. These topics can be selected while creating a Workspace.

View a Kafka Topic

To view a Kafka Topic:

1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Action** and select **View**.

The details of the Kafka Topic is displayed.

Figure 8-27 Kafka Topics page

View Topic
✕

Basic Information

Topic Name
1hp7

Topic Description
1hp7

Cluster Name
cluster1

Kafka Topic ID
1hp7

Record Schema Schema Type ⓘ

pyflink

Order	Name	Type
1	account_id	Types.STRING()
2	atm	Types.STRING()
3	amount	Types.INT()
4	transaction_id	Types.STRING()

Edit a Kafka Topic

To edit a Kafka Topic:

1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Action** next to corresponding Kafka Topic and select **Edit**.

The **Edit Topic** page is displayed.

Figure 8-28 Edit Topic page

✎ Edit Topic
✕

Basic Information

Topic Name
1hp7

Topic Description
1hp7

Cluster Name
cluster1

Kafka Topic ID
1hp7

Record Schema Schema Type ⓘ

pylink

Order ⚙	Name ⚙	Type	Action
1	account_id	Types.STRING()	✎ ✕
2	atm	Types.STRING()	✎ ✕
3	amount	Types.INT()	✎ ✕
4	transaction_id	Types.STRING()	✎ ✕

For more details on the Kafka Topic fields, see [Register a Kafka Topic](#) section.

3. Click **Update**.

Delete a Kafka Topic

To delete a Kafka Topic:

1. Click **Kafka Topics** icon to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Action** and select **Delete**.

A confirmation message is displayed.

ⓘ Note

You cannot delete a Kafka Topic mapped to a workspace.

Click **Delete**. The Kafka Topic is deleted.

OFSAA Environment

You can register OFSAA environment (Simulation and Production) from this page.

Note

The user must be mapped to ENVACCESS role to view the OFSAA Environment option.

To register a OFSAA environment:

1. Navigate to the OFSAA environment page.

This page displays the OFSAA environments in a tabular format.

2. Click **Register Environment**.

The Register Environment page is displayed.

Figure 8-29 Register a OFSAA Environment

Register Environment

Name
Production

Description
Development

Type
Production

Properties

Key	Value
Required	Required

+ Add More

3. Enter the environment **Name**. This field is mandatory.
4. Enter the **Description** for the environment. This field is mandatory.
5. In the **Type** field, select the environment as either Production or Simulation.
6. In the **Key** field, select the required option from the drop-down and enter value for the selected property. You can add more properties by clicking **Add More** icon.
7. Click **Create**.

The OFSAA Environment is added and displayed in the OFSAA Environment page.

You can view, edit, and delete OFSAA environments by clicking Actions icon.

Data Studio Options

The application's Notebook Server has the following configurable options:

- Interpreters
- Tasks
- Permissions
- Credentials
- Templates

To access the Data Studio Options page:

1. Click **Launch Workspace** to launch the workspace to display the **Dashboard** window with the application configuration and model creation menu.
2. Hover your mouse over the Data Studio Options widget . The following options are available:
 - Interpreters
 - Tasks
 - Permissions
 - Credentials
 - Templates

Note

DS 23.4.10 has introduced 'Select-Multiple' forms in Data Studio. The same should now be handled in MMG (parameters sets, dashboard, execute pipeline drawer, simulations, and so on).

Configure Interpreters

An Interpreter is a program that directly executes instructions written in a programming or scripting language without requiring them previously to be compiled into a machine language program. Interpreters are plug-ins that enable users to use a specific language to process data in the backend. Examples of Interpreters are jdbc-interpreter, spark-interpreters, python-interpreters, etc. Interpreters allow you to define customized drivers, URLs, passwords, connections, SQL results to display, etc.

In OFS Compliance Studio, Interpreters are used in Notebooks to execute code in different languages. Each Interpreter has a set of properties that are adjusted and applied across all notebooks. For example, using the python-interpreter makes it possible to change between versions, whereas the jdbc-interpreter offers to customize the URL, schema, or credentials. You can either use a default interpreter variant or create a new variant for an interpreter. You can create more than one variant for an interpreter. The benefit of creating multiple variants for an Interpreter is to connect different versions of interpreters (Python ver:3, Python ver:2, etc.). This helps to connect a different set of users and database schema. For example, Compliance Studio schema, BD schema, etc. Compliance Studio provides secure and safe credential management such as Oracle Wallet (jdbc wallet), Password (jdbc password), or KeyStores to link to interpreter variants to access secured data.

The application has ready-to-use interpreters and you can configure them based on the use case. Additional variants of interpreters are created as multiple users might require different settings to access the database securely. The jdbc Interpreters use the credentials to enable secure data access.

Note

Pyspark, spark, and ore are a few other available interpreters.
For more information on the Spark interpreter, see the OFS MMG Installation Guide.

Interpreters are configured when you want to modify URL, data location, drivers, enable or disable connections, etc.

To configure ready-to-use interpreters:

1. Click the Interpreter that you want to view from the list displayed on the LHS. The default configured interpreter variant is displayed.
2. Modify the values in the fields as per requirement. For example, to modify a parameter's limit, connect to a different schema, PGX server, and so on.
3. Click **Update**. The modified values are updated in the Interpreter.

Manage Tasks

Tasks are created when notebooks or paragraphs are executed by the Notebook users. It is important to know the status of the execution, whether the tasks are created, rejected, canceled, etc. The Tasks page allows you to view the status of the task and associated notebooks, paragraphs, interpreters, etc. By default, all the tasks are listed on the Task page. You can view the specific task using filters such as task status, date of creation, and notebook name.

Manage Tasks

You can view the logged-in users and view, add, or modify ready-to-use permissions granted to the users, roles, or groups. You can create groups, roles, and permission templates (actions).

Note

You can only view users and their details.

See the *User Access and Permissioning Management* section in the OFS Compliance Studio Administration and Configuration Guide.

Manage Credentials

Compliance Studio provides secure and safe credential management. Examples of credentials are passwords, Oracle Wallets, or KeyStores. This section links credentials (a wallet and a password) to the jdbc interpreter variant to enable secure data access. This linking enables the jdbc interpreter to securely connect to the specified Oracle Database. You can also create new credentials based on your requirement to connect to the new interpreter variants.

For more information on Credentials, see the Link Credentials section in the OFS Compliance Studio Administration and Configuration Guide.

Note

You can link credentials only to the jdbc interpreter. The Credentials section is enabled if an Interpreter variant can accept credentials.

Configure Templates

To create, import, and set the default FCCM template in the Template Dashboard:

1. Click CS Launch Workspace to display the CS Production Workspace window.
2. Click Templates from Design Studio Options.

Compliance Studio offers different formats to view the result after a paragraph's execution. Templates enable you to define parameters to customize the result formats. You can customize the visualization of the result by defining parameters in a template and then applying that template to a notebook.

Note

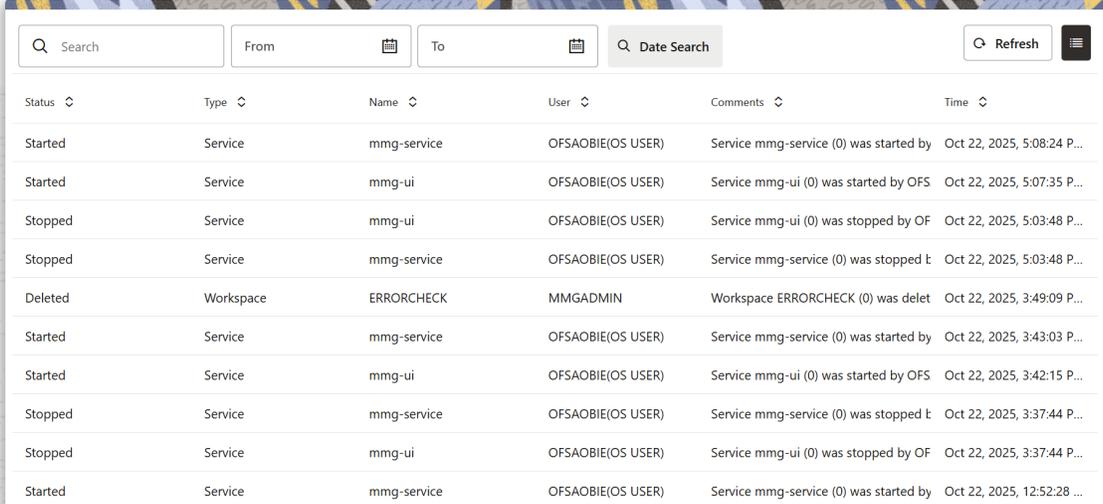
- Compliance Studio comes with a default template, but users can customize this at the template level but can also override any global template settings in each notebook paragraph.
- It is recommended to use the template that is available from out-of-the-box Compliance Studio.

For more information, see the *Interpreter Configuration and Connectivity* section in the OFS Compliance Studio Administration and Configuration Guide.

Audit Trail

The Audit Trail window provides information such as the start and stop of UI and services, add or delete of the datasource, API Registry, kafka topics, conda environments, and so on.

It supports file manager operations such as uploading files, creating folders, deleting files and folders, and previewing files. It also supports workspace related activities such as creating, editing and deleting workspaces.

Figure 8-30 The Audit Trail Page


The screenshot shows the 'Audit Trail' page with a table of audit events. The table has columns for Status, Type, Name, User, Comments, and Time. The events include started and stopped services (mmg-service, mmg-ui) and a deleted workspace (ERRORCHECK).

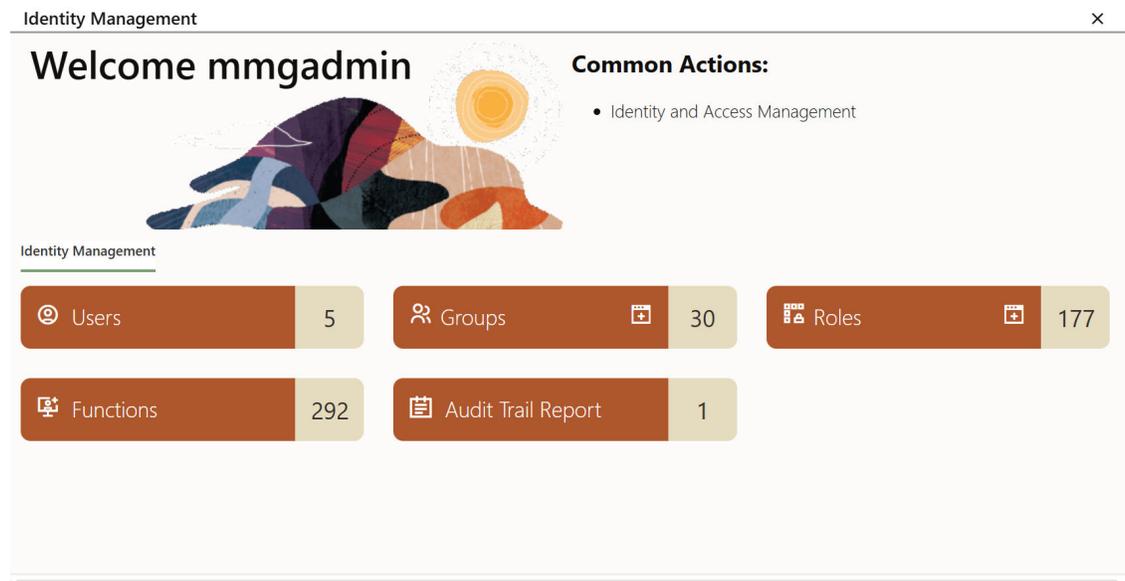
Status	Type	Name	User	Comments	Time
Started	Service	mmg-service	OFSAOBIE(OS USER)	Service mmg-service (0) was started by	Oct 22, 2025, 5:08:24 P...
Started	Service	mmg-ui	OFSAOBIE(OS USER)	Service mmg-ui (0) was started by OFS	Oct 22, 2025, 5:07:35 P...
Stopped	Service	mmg-ui	OFSAOBIE(OS USER)	Service mmg-ui (0) was stopped by OF	Oct 22, 2025, 5:03:48 P...
Stopped	Service	mmg-service	OFSAOBIE(OS USER)	Service mmg-service (0) was stopped t	Oct 22, 2025, 5:03:48 P...
Deleted	Workspace	ERRORCHECK	MMGADMIN	Workspace ERRORCHECK (0) was delet	Oct 22, 2025, 3:49:09 P...
Started	Service	mmg-service	OFSAOBIE(OS USER)	Service mmg-service (0) was started by	Oct 22, 2025, 3:43:03 P...
Started	Service	mmg-ui	OFSAOBIE(OS USER)	Service mmg-ui (0) was started by OFS	Oct 22, 2025, 3:42:15 P...
Stopped	Service	mmg-service	OFSAOBIE(OS USER)	Service mmg-service (0) was stopped t	Oct 22, 2025, 3:37:44 P...
Stopped	Service	mmg-ui	OFSAOBIE(OS USER)	Service mmg-ui (0) was stopped by OF	Oct 22, 2025, 3:37:44 P...
Started	Service	mmg-service	OFSAOBIE(OS USER)	Service mmg-service (0) was started by	Oct 22, 2025, 12:52:28 ...

Select the **Audit Timeline View** icon, adjacent to the **Refresh** icon, to view the listed sequence of actions that have been performed. Additionally, the Audit Table view (graphical representation) can also be viewed from the main **Audit Trail** page. Models can be searched by

- **Name:** Enter the name of the model. The list is automatically filtered when a value is typed in this field.
- **From:** Click the **Select Date.** icon in the **From** field to select a date.
- **To:** Click the **Select Date** icon in the **To** field to select a date. After selecting this date, click the **Date Search** icon to filter the models as per the selected dates.

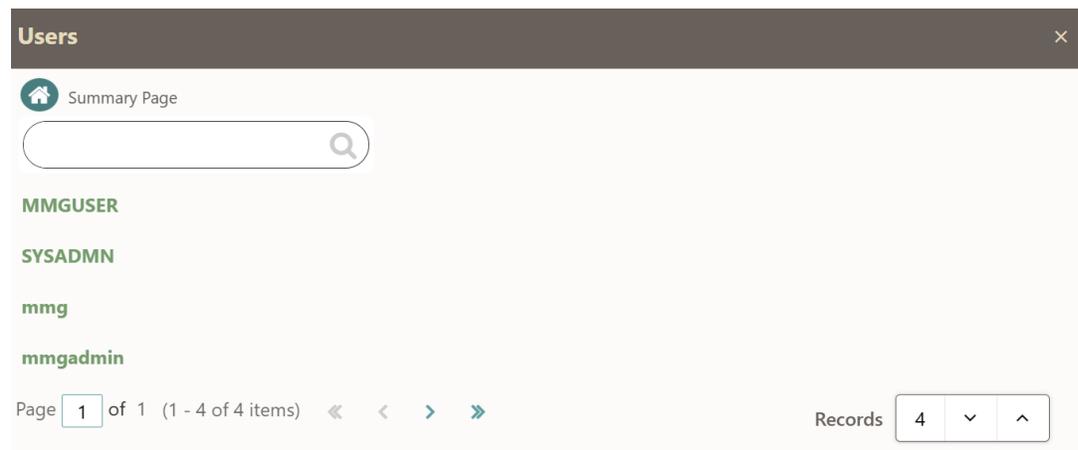
Identity Management

Click **Identity Management** to navigate to the **Identity Management** page.

Figure 8-31 The Identity Management Page

You can manage users, groups, roles, functions, and also view the audit trail report from this page.

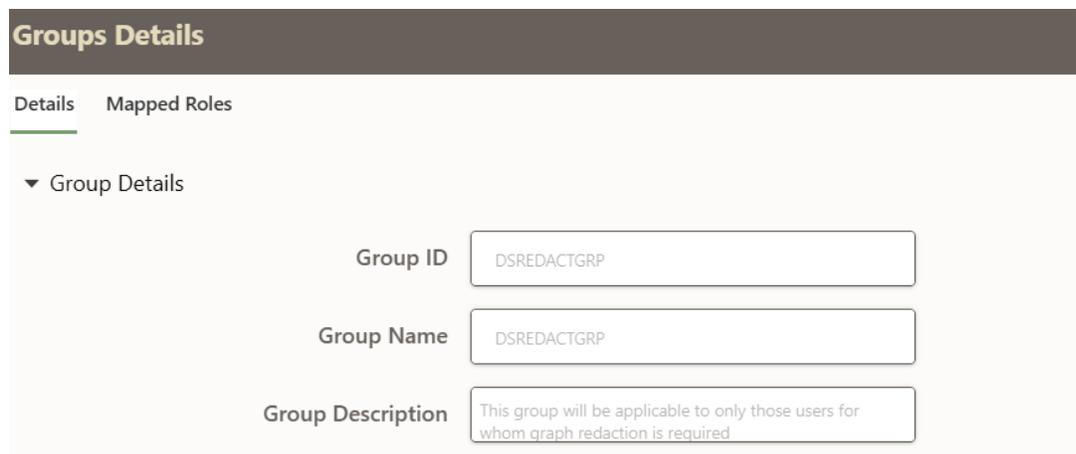
- **Users:** The users who are mapped to the application are listed on this page.

Figure 8-32 Identity Management Users

- **Groups:** The application admin users contain groups. This page displays the groups that are assigned to a user.

Figure 8-33 Identity Management Groups

Select a group to view or add details about it on the **Group Details** page. The **Details** tab is open by default.

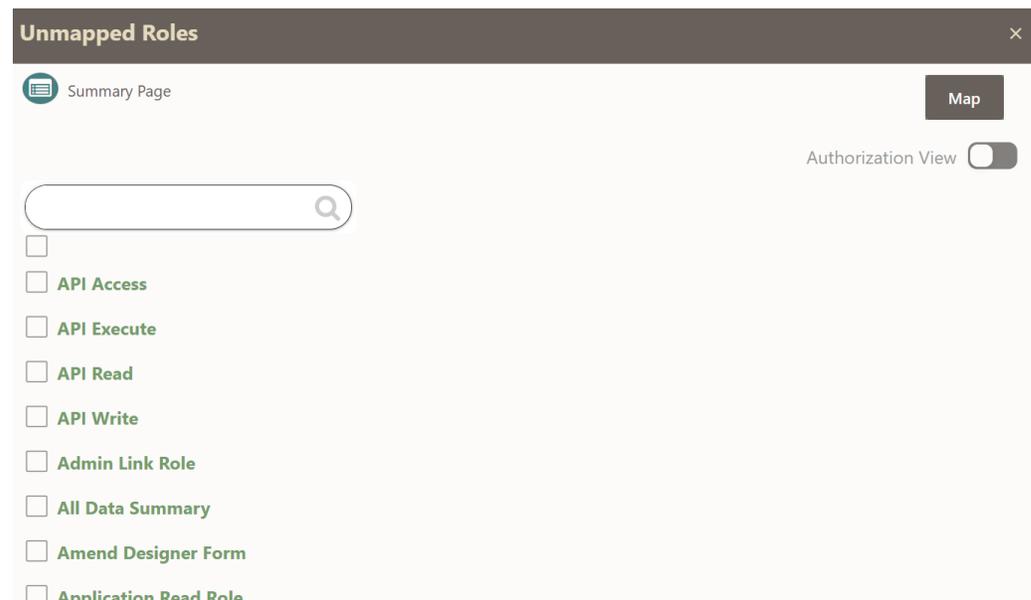
Figure 8-34 Group Details - Details

Additionally, click the **Mapped Roles** tab to view details about roles that are mapped to specific groups.

Figure 8-35 Group Details - Mapped Roles

All the groups have specific roles mapped to it.

- **New Mapping:** Click **New Mapping** to map a role to a group via the **Unmapped Roles** page. On this page, select the desired roles and click **Map**. You can also search for a specific role by entering the name of the role in the **Search** field and then clicking the search icon.

Figure 8-36 Unmapped Roles

- **Unmap:** Select the Role(s) that you want to unmap and then click **Unmap**.

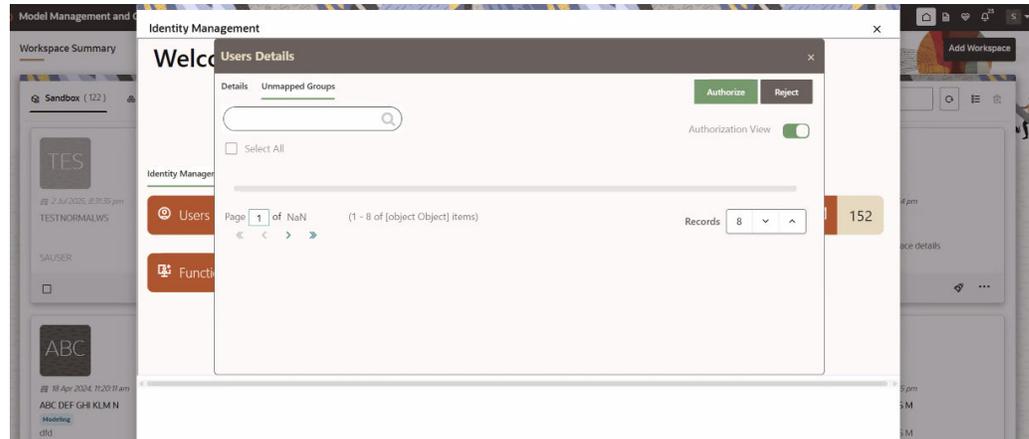
Note

If a user has logged in and then mapped the role and group with a specific user credential, then, they can unmap only other groups, and not the user group they have logged in with.

- **Authorization:** When you need to map a new role, you request for a new role or user group mapping. Select the **Authorization View** toggle. The **Authorize** and **Reject**

button appear. To authorize or approve the role or group, select **Authorize**. To reject the request, click **Reject**.

Figure 8-37 Authorization View screen



- **Roles:** The application admin users contain roles. This page displays the roles that are assigned to a user.

Figure 8-38 Roles



Select a to view or add details about it on the **Roles Details** page. The **Details** tab is open by default.

Figure 8-39 Role Details

The screenshot shows a window titled "Roles Details" with a close button (X) in the top right corner. Below the title bar, there are two tabs: "Details" (selected) and "Mapped Functions". Under the "Details" tab, there is a section titled "Role Details" with a dropdown arrow. Below this section, there are three input fields:

- Role Code:** APIACCESS
- Role Name:** API Access
- Role Description:** User Group mapped will have access to API Link and Summary

Additionally, click the **Mapped Functions** tab to view details about roles that are mapped to specific functions.

Figure 8-40 Mapped Functions

The screenshot shows a window titled "Roles Details" with a close button (X) in the top right corner. Below the title bar, there are two tabs: "Details" and "Mapped Functions" (selected). In the top right corner of the "Mapped Functions" tab, there are two buttons: "New Mapping" (green) and "Unmap" (dark grey). Below the tabs, there is a "Summary Page" section with a search bar and a search icon. Below the search bar, there is a list of mapped functions:

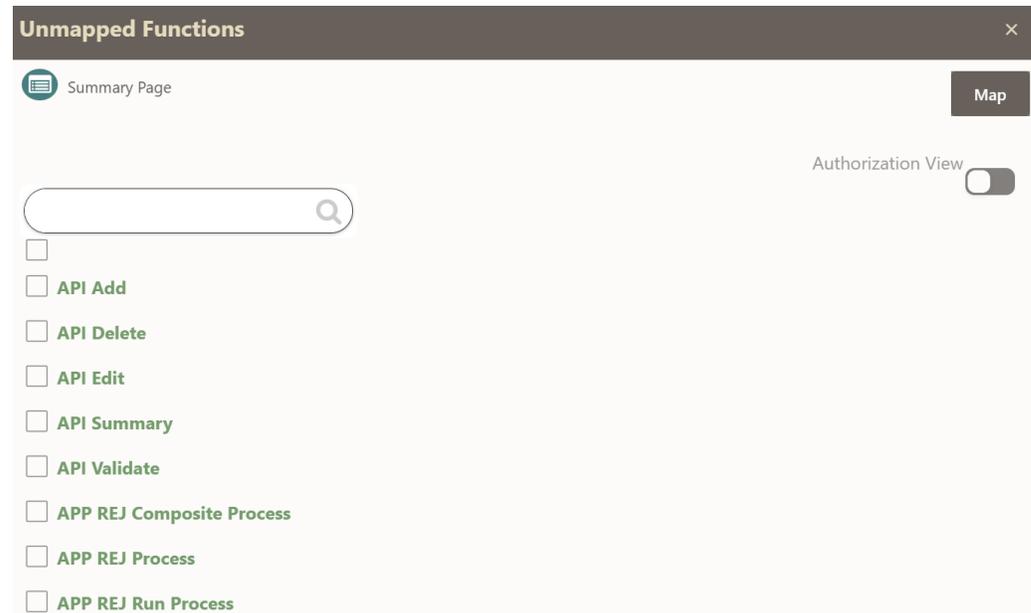
- Summary Page
- API Summary

At the bottom left, there is a pagination control: "Page 1 of 1 (1 - 1 of 1 items)" with navigation arrows. At the bottom right, there is a "Records" control showing "1" record with up and down arrows.

All the roles have specific functions mapped to it.

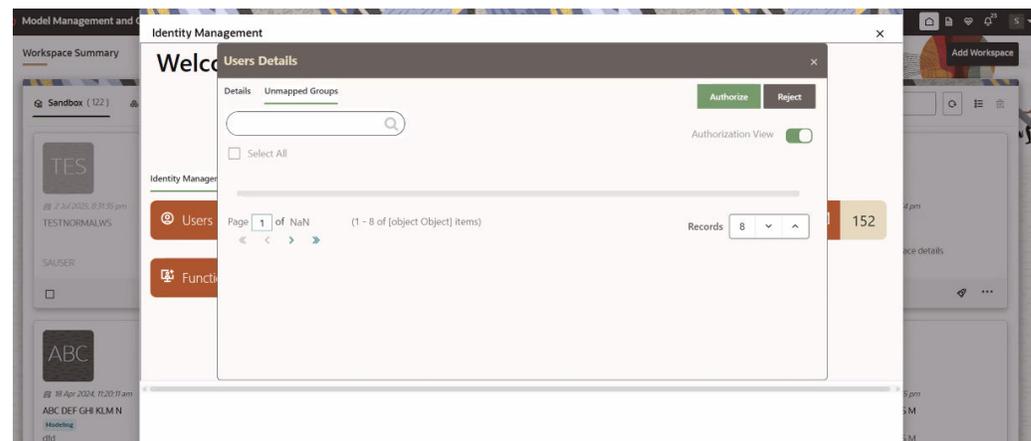
- **New Mapping:** Click **New Mapping** to map a role to a function via the **Unmapped Functions** page. On this page, select the desired functions and click **Map**. You can also search for a specific function by entering the name of the function in the **Search** field and then clicking the search icon.

Figure 8-41 Unmapped Functions



- **Unmap:** Select the function(s) that you want to unmap and then click **Unmap**.
- **Authorization:** Select the **Authorization View** toggle. The **Authorize** and **Reject** button appear. To authorize or approve the role or function, select **Authorize**. To reject the request, click **Reject**.

Figure 8-42 Authorization View



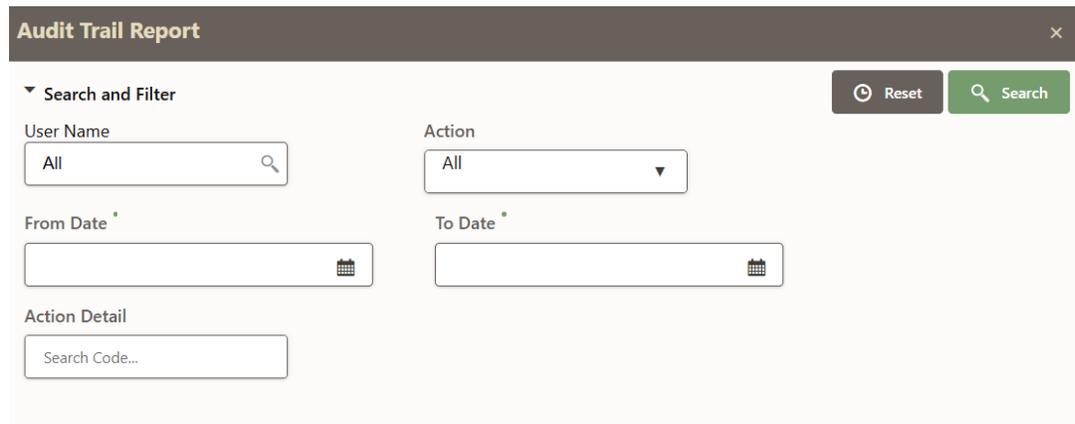
- **Functions:** View the list of functions on this page. Click a specific function to view or add details about it. You can also search for a specific function by entering the name of the function in the **Search** field and then clicking the search icon.

Figure 8-43 Functions



- **Audit Trail:** The audit trail report reflects the activities the user performs during a particular period.

Figure 8-44 Audit Trail Report



- **User Name:** Click the arrow to select a user name from the drop-down list.
- **Action:** Click the arrow to select an action from the drop-down list.
- **From Date:** Click the **Select Date** icon to select a from date.
- **To Date:** Click the **Select Date** icon to select a to date.
- **Action Detail:** Enter a value in this field.
- **Reset:** Click this button to reset the data in the audit trail report.
- **Search:** Click the search button to track a particular activity.

9

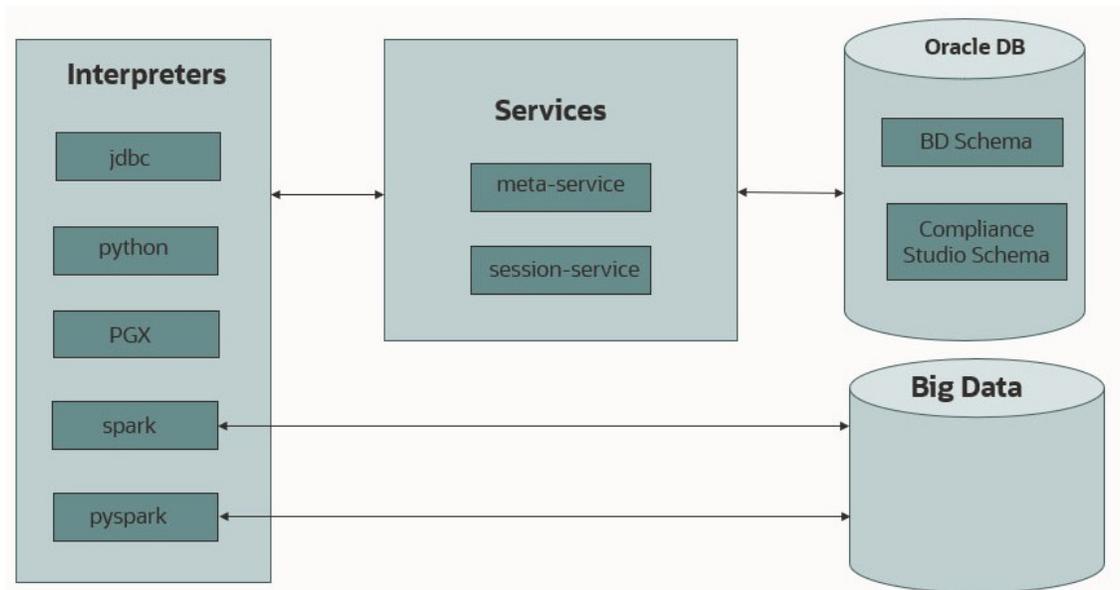
Interpreter Configuration and Connectivity

An interpreter is a program that directly executes instructions written in a programming or scripting language without requiring them previously to be compiled into a machine language program. Interpreters are plug-ins that enable users to use a specific language to process data in the backend. Examples of Interpreters are jdbc-interpreter, spark-interpreters, python-interpreters, etc. Interpreters allow you to define customized drivers, URLs, passwords, connections, SQL results to display, etc.

In OFS MMG, Interpreters are used in Notebooks to execute code in different languages. Each Interpreter has a set of adjusted and applied properties across all notebooks. For example, using the python-interpreter makes it possible to change between versions, whereas the jdbc-interpreter offers to customize the URL, schema, or credentials. In OFS MMG, you can either use a default interpreter variant or create a new variant for an interpreter. You can create more than one variant for an interpreter. The benefit of creating multiple variants for an Interpreter is to connect different versions of interpreters (Python version: 3, Python version: 2, etc.). This helps to connect a different set of users and database schema. For example, OFS MMG schema, BD schema, etc. OFS MMG provides secure and safe credential management such as Oracle Wallet (jdbc wallet), Password (jdbc password), or KeyStores to link to interpreter variants to access secured data.

The following image illustrates the examples of interpreters used in OFS MMG and database connections.

Figure 9-1 Examples of Interpreters



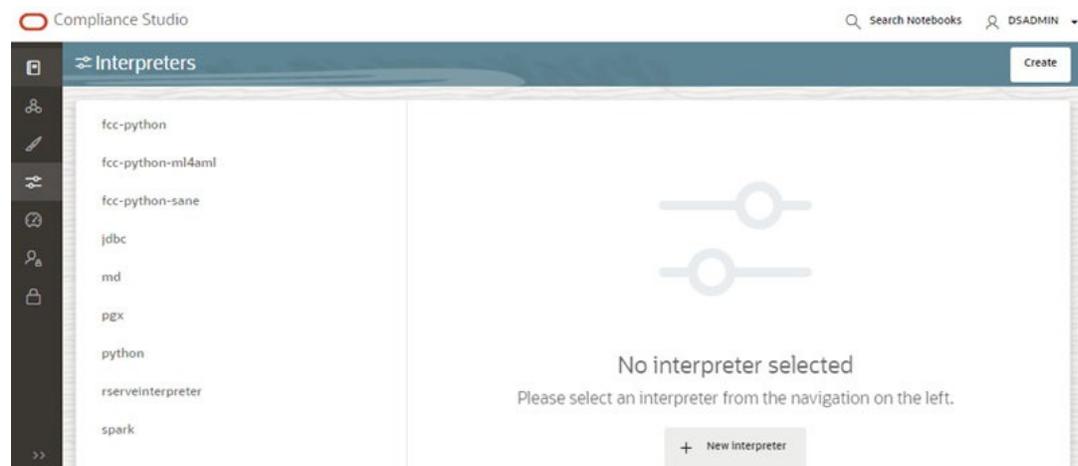
Configure Interpreters

Interpreters are configured when you want to modify URL, data location, drivers, enable or disable connections, etc.

To configure ready-to-use interpreters, follow these steps:

1. On the **Workspace Summary** page, select Launch workspace to display the **CS Production workspace** window.
2. Click the **User Profile** drop-down list and select **Data Studio Options** widget. The following options are available:
 - Interpreters
 - Tasks
 - Permissions
 - Credentials
 - Templates
3. Click **Interpreters** that you want to view from the list displayed on the LHS. The default configured interpreter variant is displayed on the RHS.

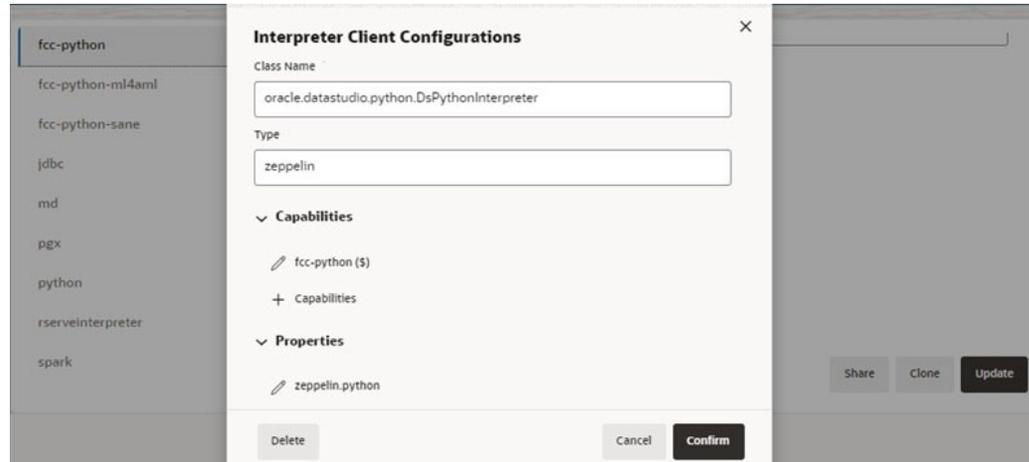
Figure 9-2 Interpreters' screen



4. Modify the values in the fields as per requirement. For example, to modify a parameter's limit, connect to a different schema, PGX server, etc.

You can modify the values in the following UI options:

- Wizard

Figure 9-3 Wizard UI options

An interpreter can group multiple interpreter clients that all run in one JVM process and can be stopped together.

For example, the spark interpreter group contains the spark and pyspark interpreter client.

Figure 9-4 Properties screen

Properties ✕

Key ↕

zeppelin.python

Environment Name

Property Name

zeppelin.python

Default Value

/scratch/fccstudio/CS_81210_81240_UP/OFS_COMPLIAI

Description

Python directory. It is set to python by default.(assume p

Type

Delete Cancel **Confirm**

Group Configuration

Initial Code

For example, when using a Spark interpreter group with spark and pyspark interpreter clients. If you define the initialization code for the spark interpreter group, the initialization code will run when the runtime environment is created, i.e., the first time a user runs a paragraph of either spark or pyspark in a notebook with OFS MMG running in NOTEBOOK session mode.

Initial Code Capability

The initial code capability defines what interpreter client to use to run the group initial code. For example, in the spark interpreter group, you would select the spark capability as the initial code capability to create a spark context for the group JVM process.

Credential Configurations

For linking any credentials to the interpreter, you have to define what credential types should be used and what credential mode to use. For example, the jdbc interpreter supports a credential type of type Password for the credential qualifier **jdbc_password** and a credential type of type Oracle Wallet for the credential qualifier **jdbc_wallet**. After defining the credential configuration, a new section for selecting the respective credential values will appear.

Interpreter Client Configuration

Interpreter properties can be configured for each interpreter client.

Figure 9-5 Interpreter Client Configuration

The screenshot shows a dialog box titled "Interpreter Client Configurations" with a close button (X) in the top right corner. The dialog contains the following fields and sections:

- Class Name:** A text input field containing "oracle.datastudio.python.DsPythonInterpreter".
- Type:** A text input field containing "zeppelin".
- Capabilities:** A section with a downward arrow icon and the text "Capabilities". It contains a list item "fcc-python (\$)" with a pencil icon, and a "+ Capabilities" button.
- Properties:** A section with a downward arrow icon and the text "Properties". It contains a list item "zeppelin.python" with a pencil icon.

At the bottom of the dialog, there are three buttons: "Delete", "Cancel", and "Confirm".

Lifecycle Configuration

Host Mode

In the Host lifecycle mode, the following properties can be configured:

- **Host:** The hostname on which the interpreter is listening. For example, localhost if the **interpreter** runs on the same machine as the server.
- **Port:** The port on which the interpreter is listening.

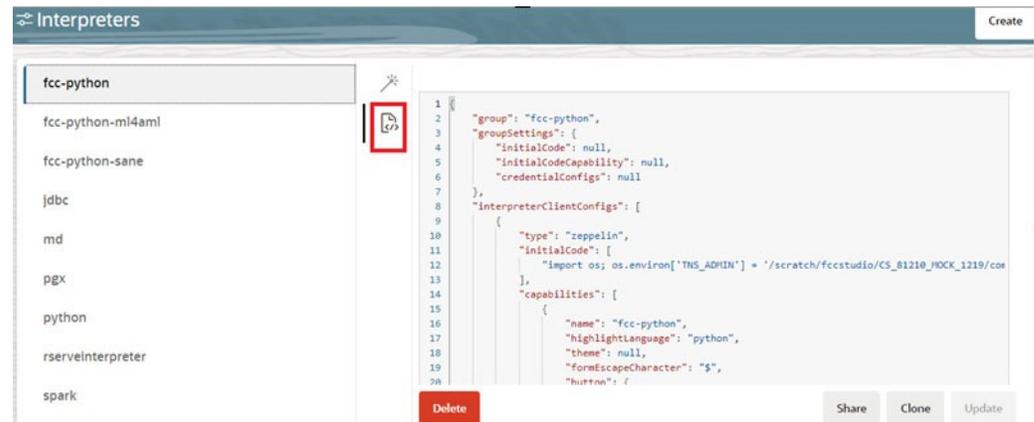
Credentials

A credential section appears if you have defined a credential configuration as part of the group settings. For each credential qualifier, an already defined credential can be

selected. If the credential mode Per User is used, each individual user has to select their own credential.

- **JSON:**
You can modify the values in the properties of the interpreter in the JSON file, as shown in the following figure.

Figure 9-6 JSON file properties



5. Click Update. The modified values are updated in the Interpreter.
6. The user can also perform Share, Clone, and Delete operations on this screen.

The following table lists the Ready-to-use interpreter in OFS MMG.

Table 9-1 Ready-to-use interpreter

Interpreters	Description
python Interpreter	<p>The python interpreter is used to write Python code in a notebook to analyze data from different sources, machine learning, artificial intelligence, etc.</p> <p>The python interpreter uses a python conda environment. OFS MMG comes with predefined conda environments as follows:</p> <ul style="list-style-type: none"> • default_<CS version> • ml4aml_<CS version> • sane_<CS version> <p>Before executing any python notebooks, you need to attach the conda environment using drop-down list.</p>

Table 9-1 (Cont.) Ready-to-use interpreter

Interpreters	Description
jdbc Interpreter	<p>The jdbc interpreter is a ready-to-use interpreter used to connect to Studio schema. This Interpreter is used to connect and write SQL queries on any schema without any restriction. In the jdbc Interpreter, you can configure schema details, link Wallet Credentials to the jdbc Interpreter, etc.</p> <p>Note:</p> <p>This feature is not recommended approach because it can only be used to connect to a single schema, and all users will have access to that, rather than access being managed per user. In future releases this interpreter will not be enabled by default but instructions will be given to enable if required.</p> <p>Limitation</p> <ul style="list-style-type: none"> Data source configuration is not dynamic; instead, it is static from the Interpreter Configuration screen. There is no restriction or secure access of data provided with this interpreter.
jdbc Interpreter	<p>Recommendation</p> <p>Users are recommended to use a python interpreter to get dynamic data source configuration; even data access permission features can also be used with this interpreter.</p>
md Interpreter	<p>The md interpreter is used to configure the markdown parser type. This Interpreter displays text based on Markdown, which is a lightweight markup language.</p> <p>The connection does not apply to this Interpreter.</p>
pgql Interpreter (part of PGX interpreter)	<p>The pgql interpreter is a ready-to-use interpreter used to connect the configured PGX server. This Interpreter is used to perform queries on the graph in OFS MMG. PGQL is a graph query language built on top of SQL, bringing graph pattern matching capabilities to existing SQL users and new users interested in graph technology but who do not have an SQL background.</p>
pgx-python (part of PGX interpreter)	<p>The pgx-python interpreter is a ready-to-use interpreter used to connect to the configured PGX server. It is a python based interpreter with a PGX python client embedded in it to query on graph present in the PGX server. By default, this Interpreter points to ml4aml Python Virtual environment.</p>
pgx-algorithm Interpreter (part of PGX interpreter)	<p>The pgx-algorithm interpreter is a ready-to-use interpreter that connects to the configured PGX server. This Interpreter is used to write an algorithm on the graph and is also used in the PGX interpreter.</p>

Table 9-1 (Cont.) Ready-to-use interpreter

Interpreters	Description
pgx-java Interpreter (part of PGX interpreter)	The pgx-java interpreter is a ready-to-use interpreter that connects to the configured PGX server. It is Java11 based interpreter with a PGX client embedded in it to query on graph present in the PGX server.
spark Interpreter	The spark interpreter connects to the big data environment by default. Users must write for connection either in the Initialization section or in the notebook's paragraph. This Interpreter is used to perform analytics on data present in the big data clusters in the Scala language. This requires additional configuration, which must be performed as a prerequisite or as post-installation with the manual change of interpreter settings. In the spark interpreter, you can configure the cluster manager to connect, print the Read Eval Print Loop (REPL) output, the total number of cores to use, etc.
pyspark Interpreter	The pyspark interpreter connects to the big data environment by default. Users must write code for connection either in the Initialization section or in the notebook's paragraph. This Interpreter is used to write the pyspark language to query and perform analytics on data present in big data. This requires additional configuration, which must be performed as a prerequisite or as post-installation with the manual change of interpreter settings. In the pyspark Interpreter, you can configure the Python binary executable to use for PySpark in both driver and workers, set true to use IPython, else set to false, etc.

python Interpreter

In OFS MMG, the python interpreter uses a python conda environment. OFS MMG comes with predefined conda environment as follows:

- **default_<CS Version>**
- **ml4aml_<CS Version>**
- **sane_<CS Version>**

%python interpreter points to a different conda environment. The following table lists the predefined conda environment.

Table 9-2 Predefined Conda Environment

Conda Environment	Description
default_<CS version>	Default python interpreter.
ml4aml_<CS version>	Python interpreter for ML4AML use cases.

Table 9-2 (Cont.) Predefined Conda Environment

Conda Environment	Description
sane_<CS version>	Python interpreter for scoring Name and Address Matching.

Note

Users can also configure the python libraries. For more information about python libraries, see the Python Libraries for Predefined Conda Environment section.

Configure a Python Interpreter

To configure a python interpreter, follow these steps:

1. On the Interpreter page LHS menu, select python. The python interpreter pane is displayed.
2. On the Interpreter Settings page, expand **Interpreter Client Configurations** and click the Edit icon for **<Class Name> (zeppelin)**. The Interpreter Client Configurations Window is displayed.
3. Enter the following information in the python interpreter variant pane as described in the following table.

Table 9-3 Python Interpreter Settings

Field	Description
zeppelin.python	Enter the Python installed path. The value points to the default Python version set for the Interpreter.
zeppelin.python.useIPython	Set to True to use IPython, else set to False .
zeppelin.python.maxResult	Enter the maximum number of results that must be displayed. By default the value is 1000 . Note: To update the default value for this property, see the Setting the Maximum Number of Results in the Python Interpreter section.
zeppelin.interpreter.output.limit	Output message from interpreter exceeding the limit will be truncated. Set the default value and the value ranges from 102400 to 10240000 bytes. Note: Increasing the Default Value from 102400 bytes to a higher value may slow down output rendering in the python paragraph. If the zeppelin.interpreter.output.limit property is unavailable, create and set a default value. For more information, see the Create Zeppelin Interpreter Output Limit in the Python Interpreter section.

Change Version in the Python Interpreter

In the python Interpreter, the Linux console uses the default python version in `/user/fcstudio/python_user/bin/python` as value. If you want to modify the python version, either you can

create an interpreter variant or modify the existing python version in the same interpreter variant.

Note

The **python2** is the default version used in the Linux console and is no longer supported. Hence, you can use any version of **python3** or any conda environment with a specific python version or a specific version of python packages.

To use a different version of Python, follow these steps:

1. Navigate to the **python** Interpreter Settings page.
2. Expand **Interpreter Client Configurations** and click the Edit icon for <Class Name> (zeppelin). The Interpreter Client Configurations Window is displayed.
3. Click `zeppelin.properties`. The Properties window is displayed.
4. Change the default Python version in the Default Value parameter to the new version. `<MMG_INSTALLATION_PATH>/deployed/python-packages/ defaultVirtualEnv/bin/ <Python Version>`.

By default, it is python3.

For example, `<MMG_INSTALLATION_PATH>/deployed/pythonpackages/ defaultVirtualEnv/bin/python3`.

5. Create a new interpreter variant and configure the version in the Default Value parameter. For information on creating a new interpreter variant, see Create an Interpreter Variant section. For example, to use Python 3.6.13, create a new python interpreter variant and enter the value as python 3.6.13.

Create Zeppelin Interpreter Output Limit in the Python Interpreter

Note

This section is applicable only when `zeppelin.interpreter.output.limit` is unavailable in the Python Interpreter.

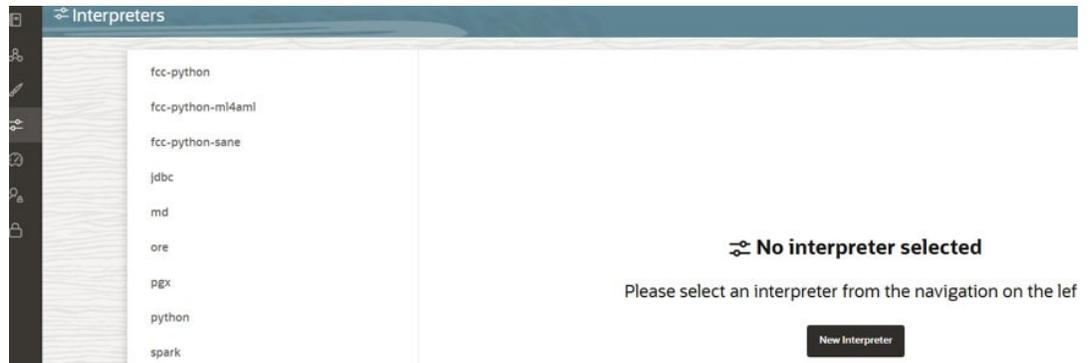
You can create the Zeppelin Interpreter Output Limit in the Python interpreter either in OFS MMG UI or in OFS MMG Server.

OFS MMG UI

To create the Zeppelin Interpreter Output Limit in Python Interpreter via OFS MMG UI, follow these steps:

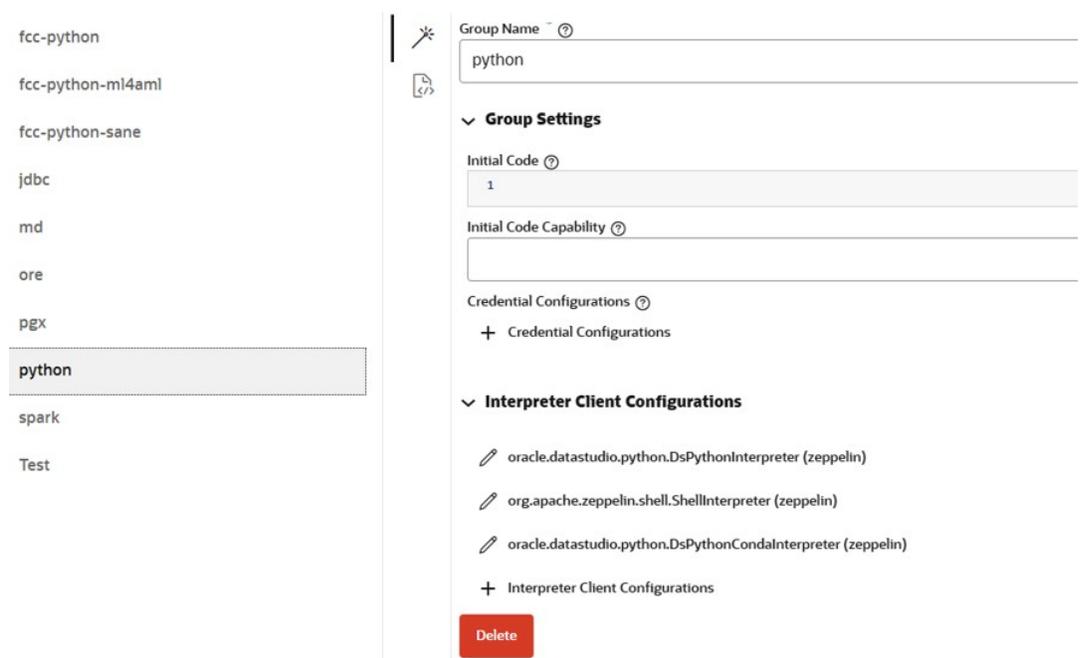
1. Log in to OFS MMG UI.
2. Click **User Profile**, select **Data Studio Options** and then click **Interpreters**. The **Interpreters** page is displayed.

Figure 9-7 Interpreters



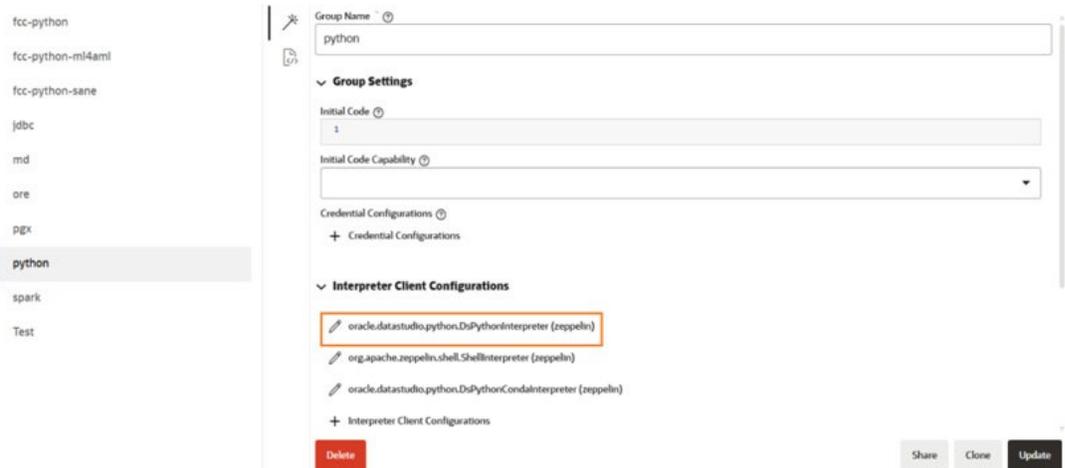
3. On the Left-Hand Side, click **python** interpreter. By default, **Wizard** pane is displayed.

Figure 9-8 Python Interpreter



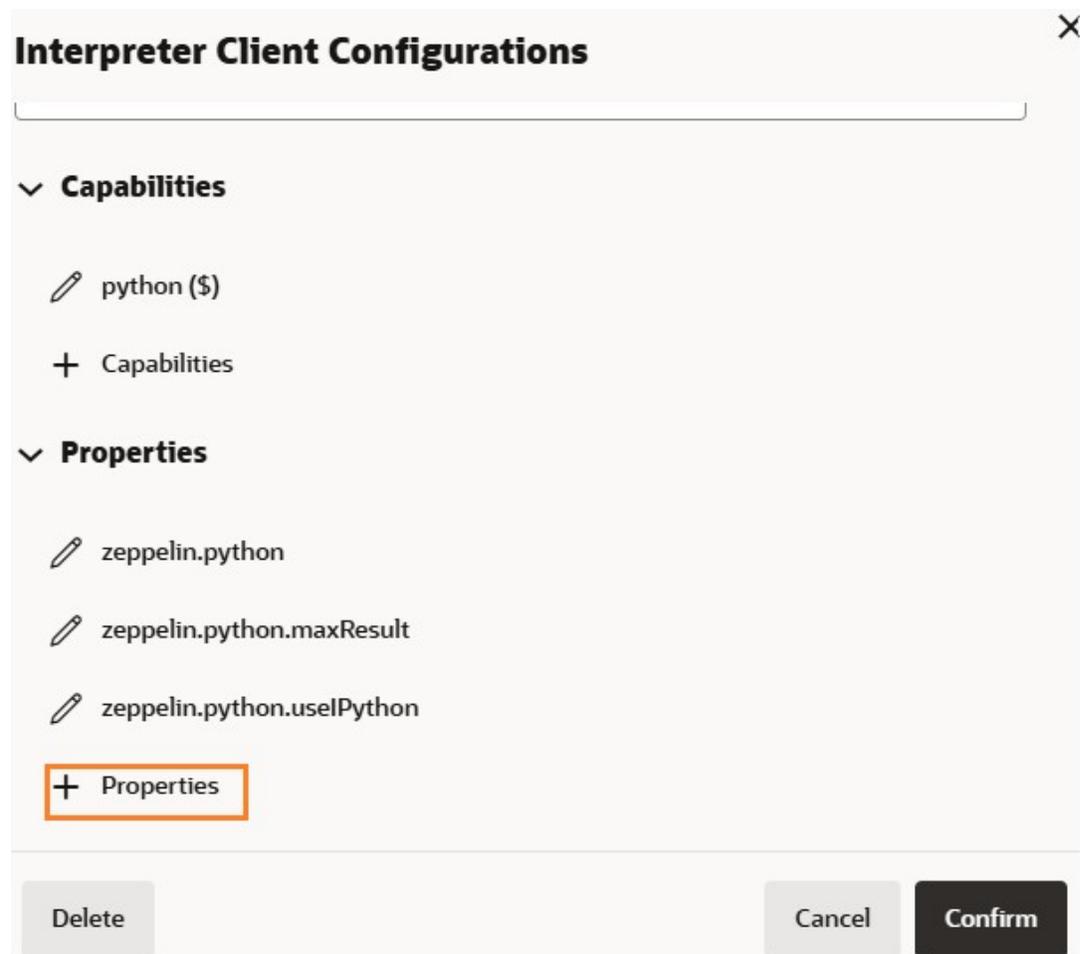
4. On the Right-Hand Side, navigate to **Interpreter Client Configurations**.

Figure 9-9 Interpreter Client Configurations



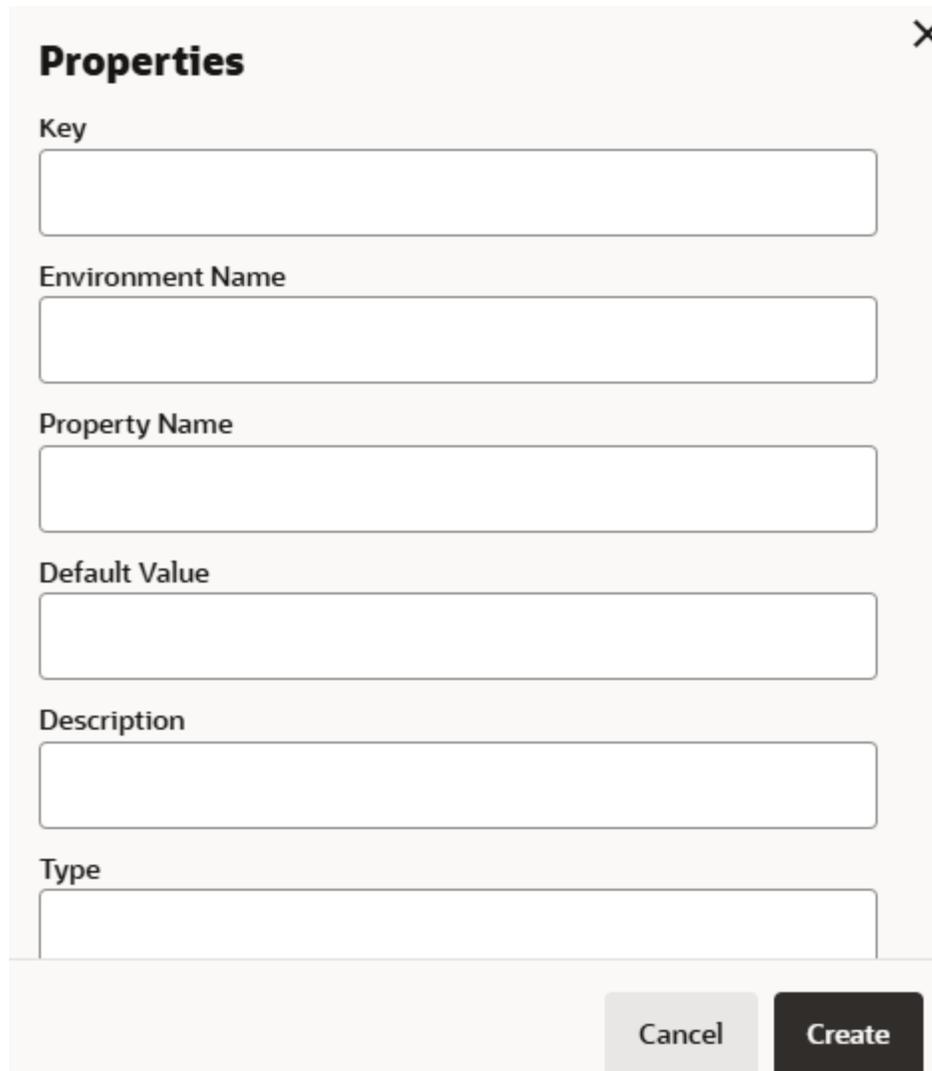
5. Click **oracle.datastudio.python.DsPythonInterpreter (zeppelin)**. The **Interpreter Client Configurations** pane is displayed.

Figure 9-10 Zeppelin Properties



6. Navigate to Properties and click **+ Properties**. The **Properties** pane is displayed.

Figure 9-11 Properties of Zeppelin Interpreter Output Limit



The screenshot shows a 'Properties' dialog box with the following fields:

- Key**: An empty text input field.
- Environment Name**: An empty text input field.
- Property Name**: An empty text input field.
- Default Value**: An empty text input field.
- Description**: An empty text input field.
- Type**: An empty text input field.

At the bottom right of the dialog, there are two buttons: a light gray 'Cancel' button and a dark gray 'Create' button.

7. Enter the **Key** and **Property Name** as `zeppelin.interpreter.output.limit`.

Note

The **Environment Name**, **Description**, and **Type** fields are optional, and you can provide the details if required.

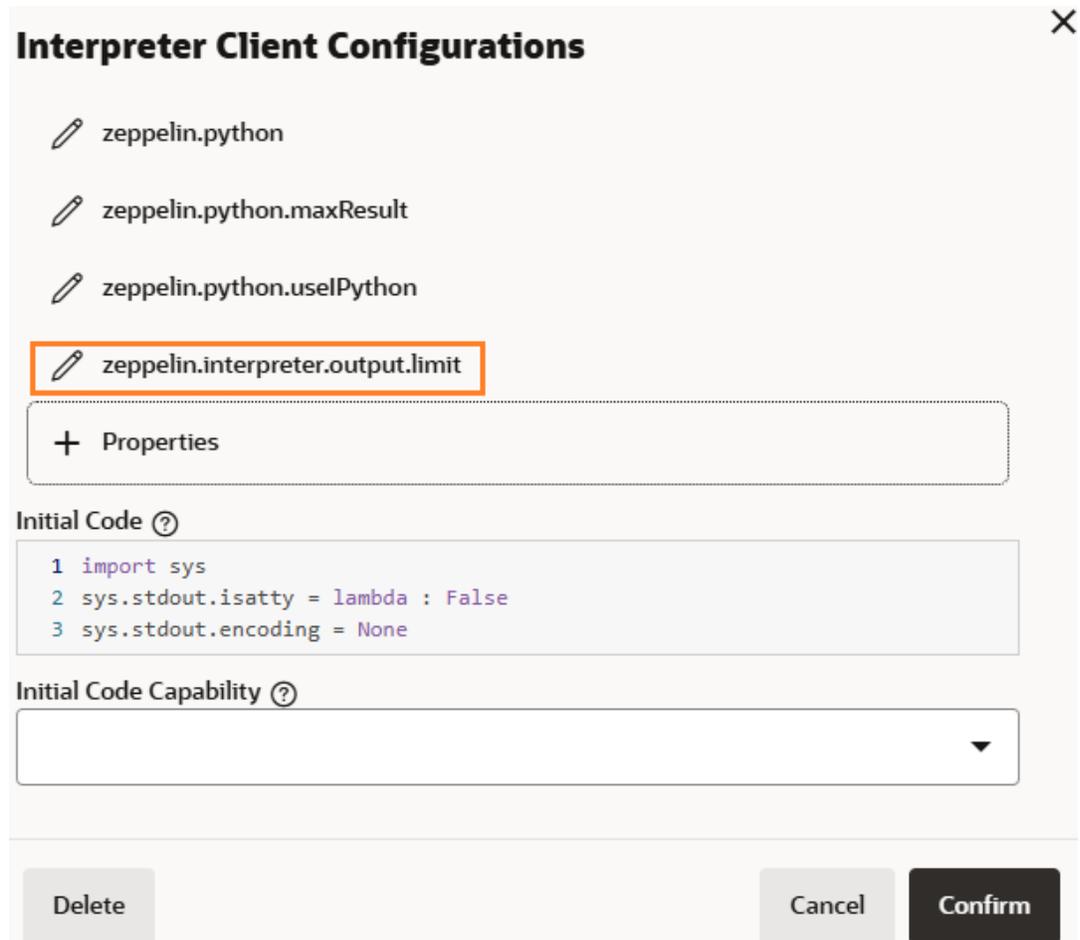
8. Update the preferred value in the **Default Value**. The value ranges from **102400** to **10240000** bytes.

Note

Increasing the Default Value from 102400 bytes to a higher value may slow down output rendering in the python paragraph.

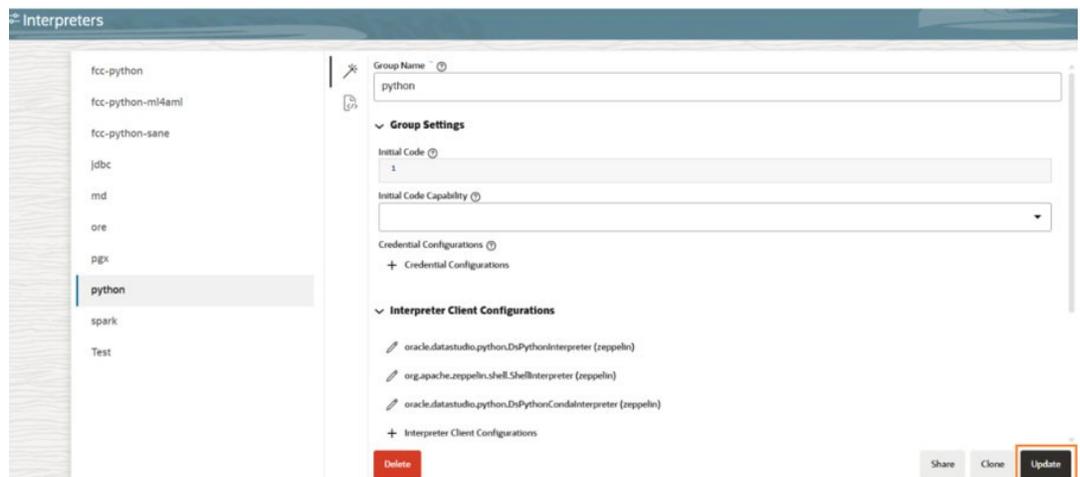
9. Click **Create**. The `zeppelin.interpreter.output.limit` property will be created and displayed in the Interpreter Client Configurations section.

Figure 9-12 Zeppelin Interpreter Output Limit



- Click **Confirm** to confirm the changes. The **Python Interpreter** page is displayed.

Figure 9-13 Update Python Interpreter



- On the Right-Hand Side, click **Update** to save the modifications. A confirmation message will indicate that the Python interpreter has been updated.

- Restart OFS MMG. The new `zeppelin.interpreter.output.limit` will be created in the python interpreter.

OFS MMG Server

To create the Zeppelin Interpreter Output Limit in the Python interpreter in OFS MMG Server, follow these steps:

- Navigate to the `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/ builtin/interpreters` directory.
- Open the `python.json` file and navigate to `className": "oracle.datastudio.python.DsPythonInterpreter`.
- Add the following lines at the end of the zeppelin properties.

```
"zeppelin.interpreter.output.limit ": {
    "envName": "",
    "propertyName": "",
    "defaultValue": "",
    "description": "",
    "type": ""
}
```

You can refer the following example when default value is 102400 bytes.

```
"zeppelin.interpreter.output.limit ": {
    "envName": "null",
    "propertyName":
"zeppelin.interpreter.output.limit",
    "defaultValue": "102400",
    "description": "Output message from interpreter
exceeding the limit will be truncated",
    "type": "number"
}
```

- Save and close the `python.json` file.
- Restart OFS MMG. The new `zeppelin.interpreter.output.limit` will be created in the python interpreter.

Setting the Maximum Number of Results in the Python Interpreter

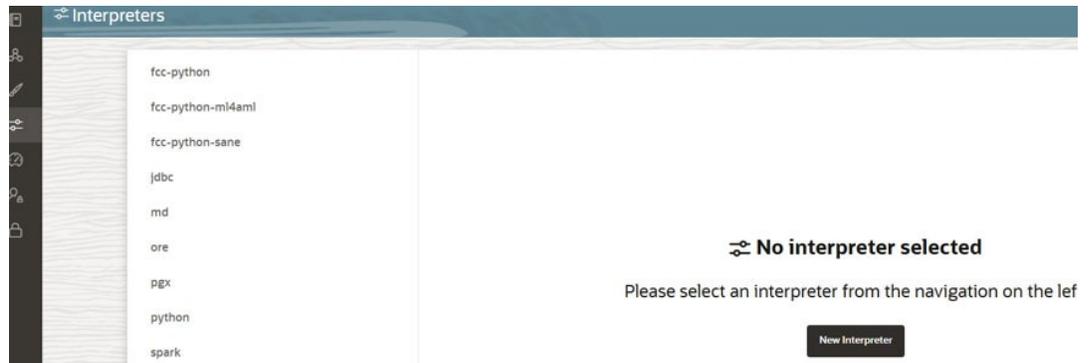
Note

This section is applicable only when modifying the default value of the `zeppelin.python.maxResult` property.

To set the maximum number of results in the Python Interpreter, follow these steps:

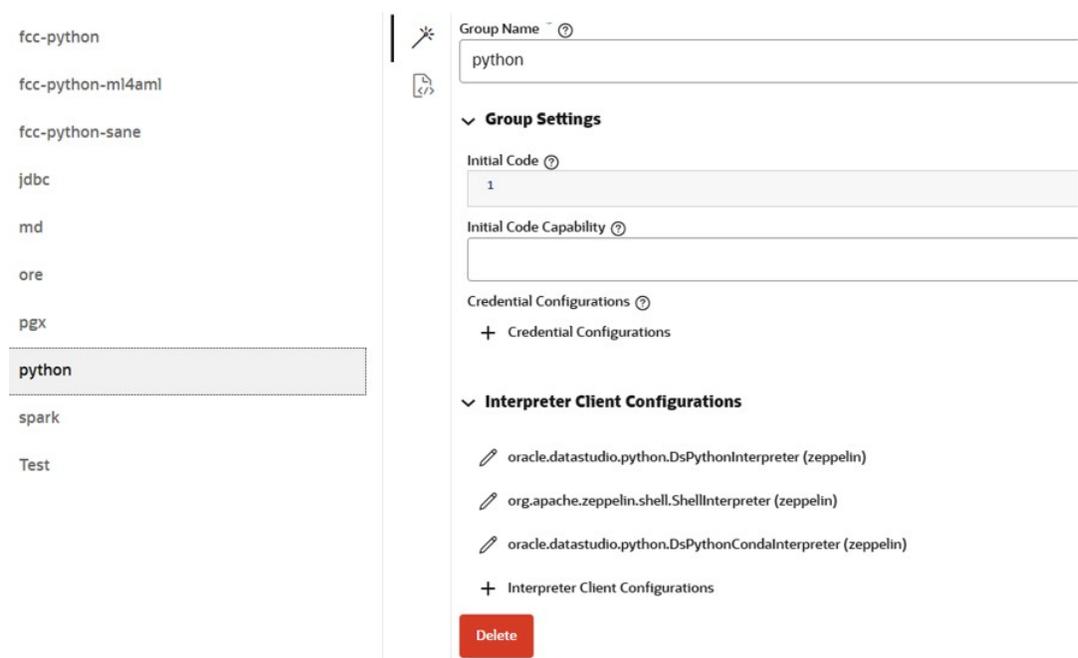
- Log in to OFS MMG UI.
- Click **User Profile**, select **Data Studio Options** and then click **Interpreters**. The **Interpreters** page is displayed.

Figure 9-14 Interpreters



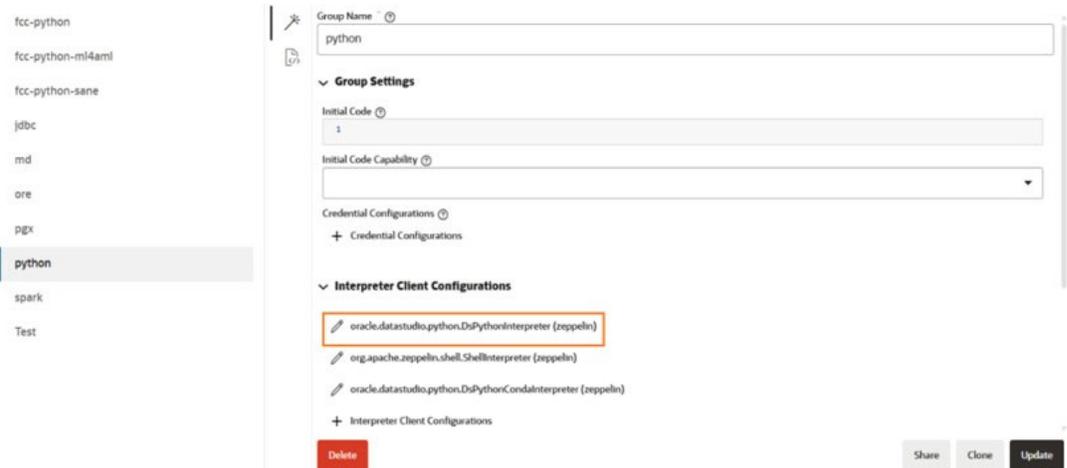
3. On the Left-Hand Side, click **python** interpreter. By default, **Wizard** pane is displayed.

Figure 9-15 Python Interpreter



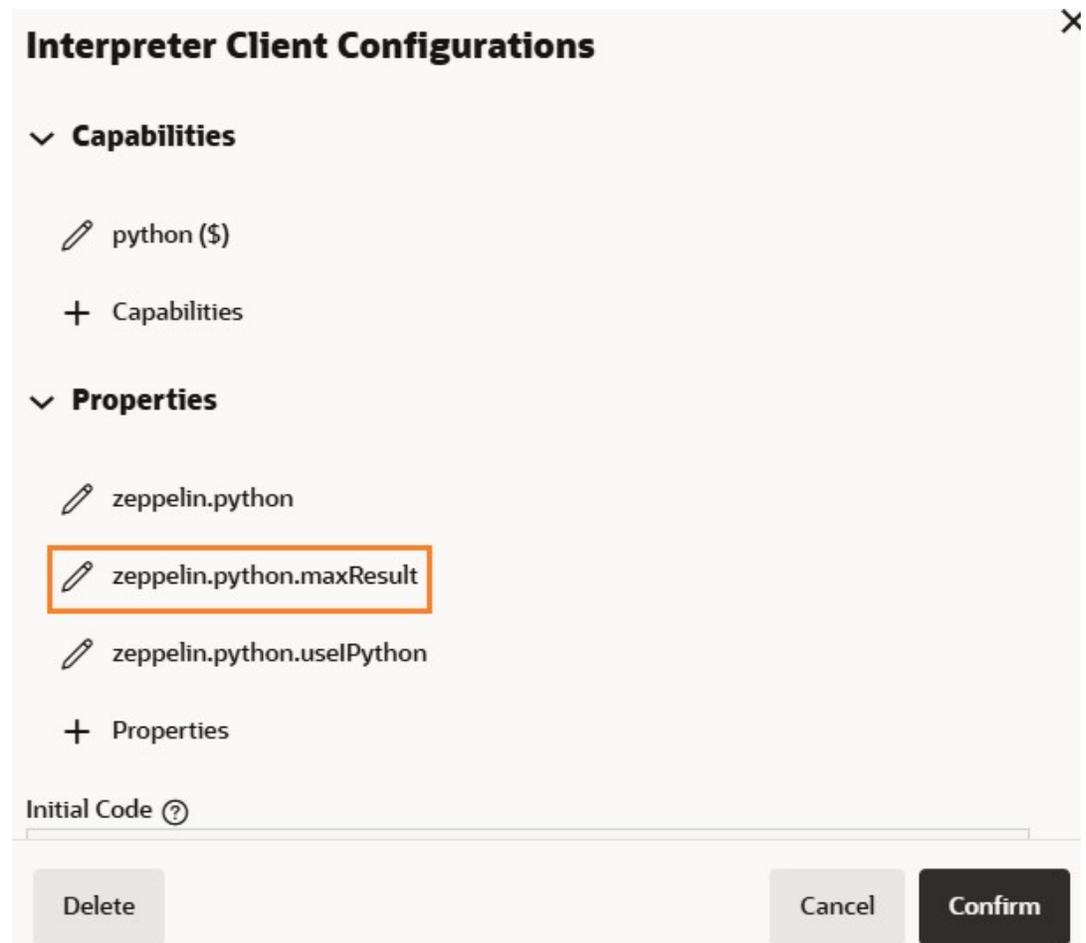
4. On the Right-Hand Side, navigate to **Interpreter Client Configurations**.

Figure 9-16 Interpreter Client Configurations



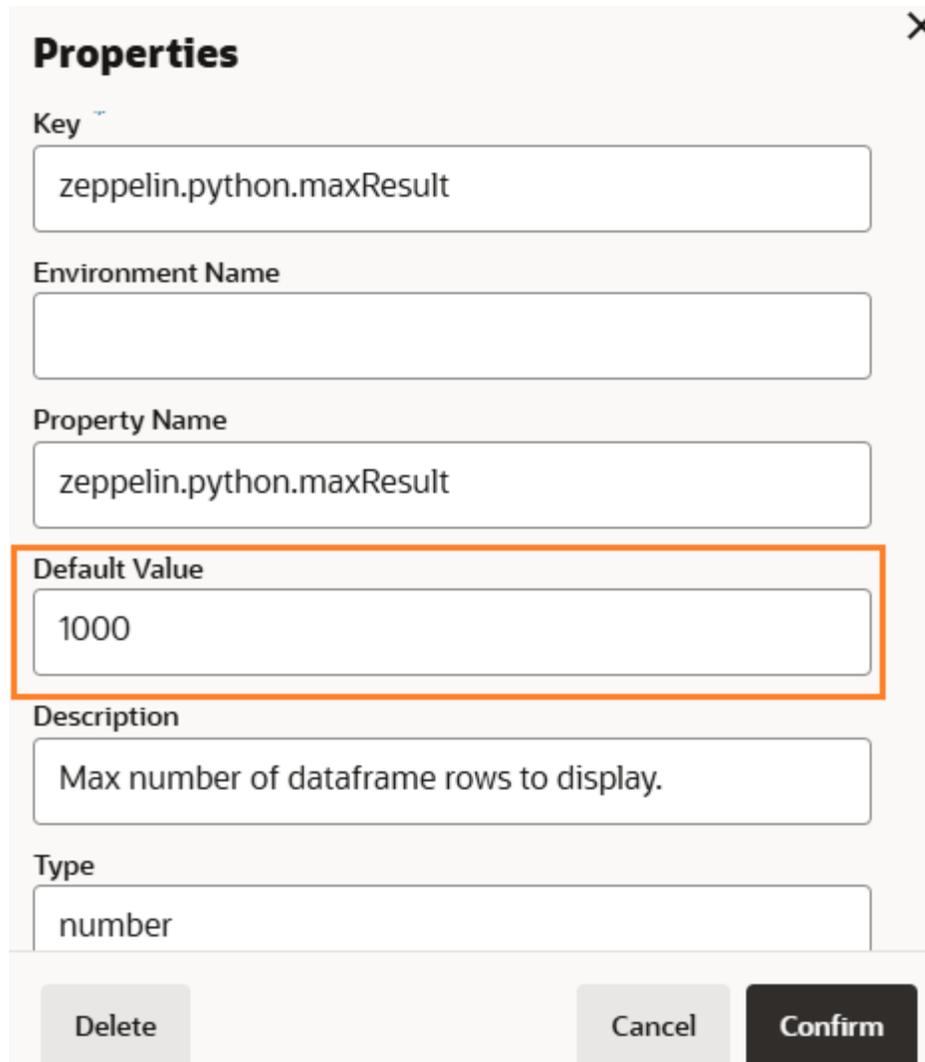
5. Click **oracle.datastudio.python.DsPythonInterpreter (zeppelin)**. The **Interpreter Client Configurations** pane is displayed.

Figure 9-17 Zeppelin Python maxResult



6. Click **zeppelin.python.maxResult**. The **Properties** pane is displayed.

Figure 9-18 Properties of zeppelin.python.maxResult



Properties ✕

Key [™]

zeppelin.python.maxResult

Environment Name

Property Name

zeppelin.python.maxResult

Default Value

1000

Description

Max number of dataframe rows to display.

Type

number

Delete Cancel Confirm

7. Update the **Default Value** based on your preference. By default the value is **1000**.
8. Click **Confirm** to confirm the changes. The **Interpreter Client Configurations** page is displayed.
9. Again, click **Confirm** to confirm the changes. The **Python Interpreter** page is displayed.
10. On the Right-Hand Side, click **Update** to save the modifications. A confirmation message will indicate that the Python interpreter has been updated.
11. Restart Compliance Studio. The default value for the maximum number of results will be updated in the Python Interpreter.

jdbc Interpreter

Note

This feature is not recommended approach because it can only be used to connect to a single schema, and all users will have access to that, rather than access being managed per user. In future releases this interpreter will not be enabled by default but instructions will be given to enable if required.

Limitation

- Data source configuration is not dynamic; instead, it is static from the Interpreter Configuration screen.
- There is no restriction or secure access of data provided with this interpreter.

Recommendation

Users are recommended to use a python interpreter to get dynamic data source configuration; even data access permission features can also be used with this interpreter.

The jdbc Interpreter is a ready-to-use interpreter used to connect Studio schema without OFSAA. This Interpreter is used to connect and write SQL queries on any schema without any restriction. The jdbc interpreter has no security attributes. It can be used to access any schema. In the jdbc interpreter, you can configure schema details, link Wallet Credentials to the jdbc Interpreter, etc.

Prerequisites

1. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ conf` directory.
2. Open the `application.yml` file and update **overwrite-builtin** property as **false**.
3. Save the changes and close the `application.yml` file.
4. Restart OFS MMG.

Configure a jdbc Interpreter

Note

This feature is not recommended approach because it can only be used to connect to a single schema, and all users will have access to that, rather than access being managed per user. In future releases this interpreter will not be enabled by default but instructions will be given to enable if required.

Limitation

- Data source configuration is not dynamic; instead, it is static from the Interpreter Configuration screen.
- There is no restriction or secure access of data provided with this interpreter.

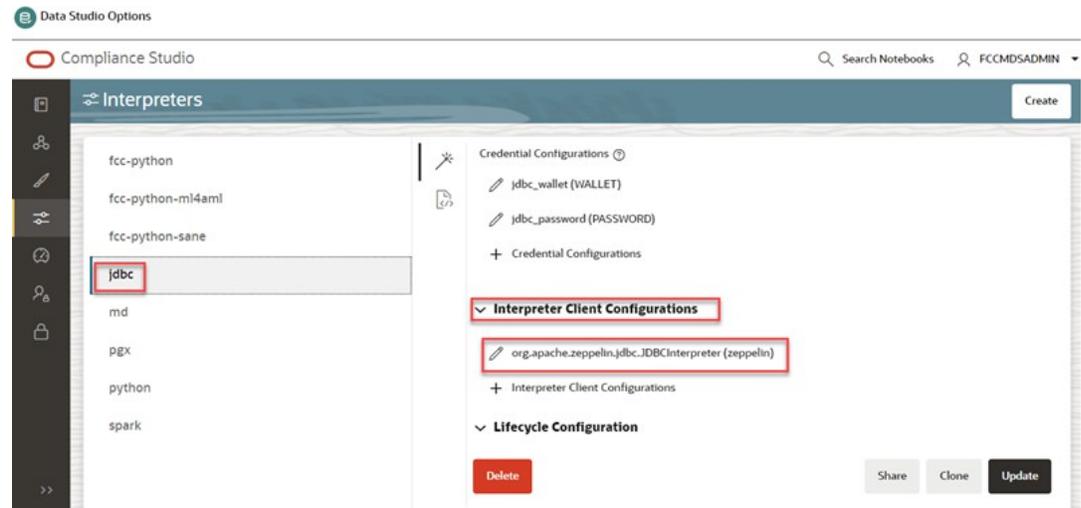
Recommendation

Users are recommended to use a python interpreter to get dynamic data source configuration; even data access permission features can also be used with this interpreter.

To configure a jdbc interpreter, follow these steps:

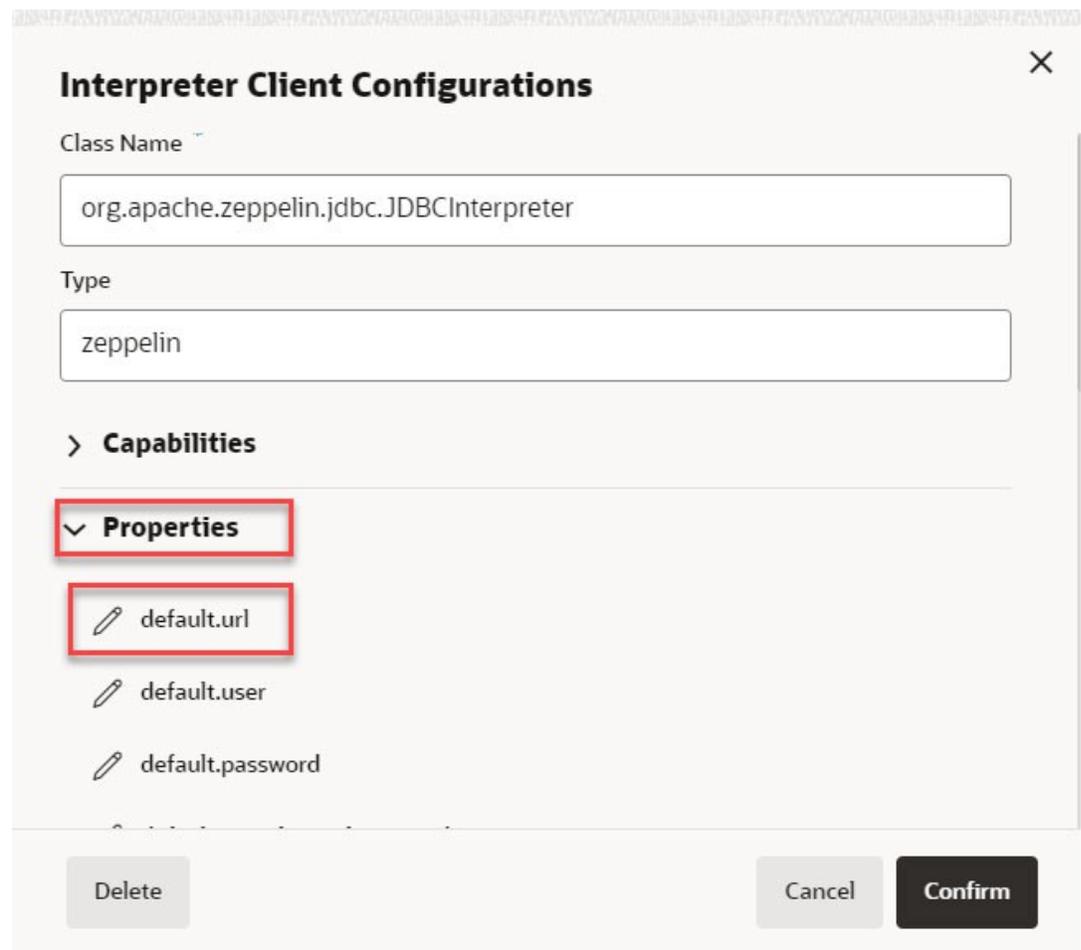
1. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ bin` directory.
2. Open `startup.sh` file, navigate to line 29 and update jdbc value as **7011**.
For example: `./"$DIR"/datastudio --port 7008 --markdown 7009 --spark 7014 --python 7012 - -jdbc 7011 --shell -1 --pgx 7022 --external`
3. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ bin` directory.
4. Open the `config.sh` file and update `DATASTUDIO_JDBC_INTERPRETER_PORT` as **7011**.
5. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/bin` directory.
6. Open the `config.sh` file and update `DATASTUDIO_JDBC_INTERPRETER_PORT` as **7011**.
7. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/builtin/interpreters/jdbc.json` directory.
8. Navigate to line 154 and update port value as **7011**.
9. Restart OFS MMG.
10. On the Interpreter page LHS menu, select jdbc. The jdbc interpreter pane is displayed.

Figure 9-19 jdbc Interpreter



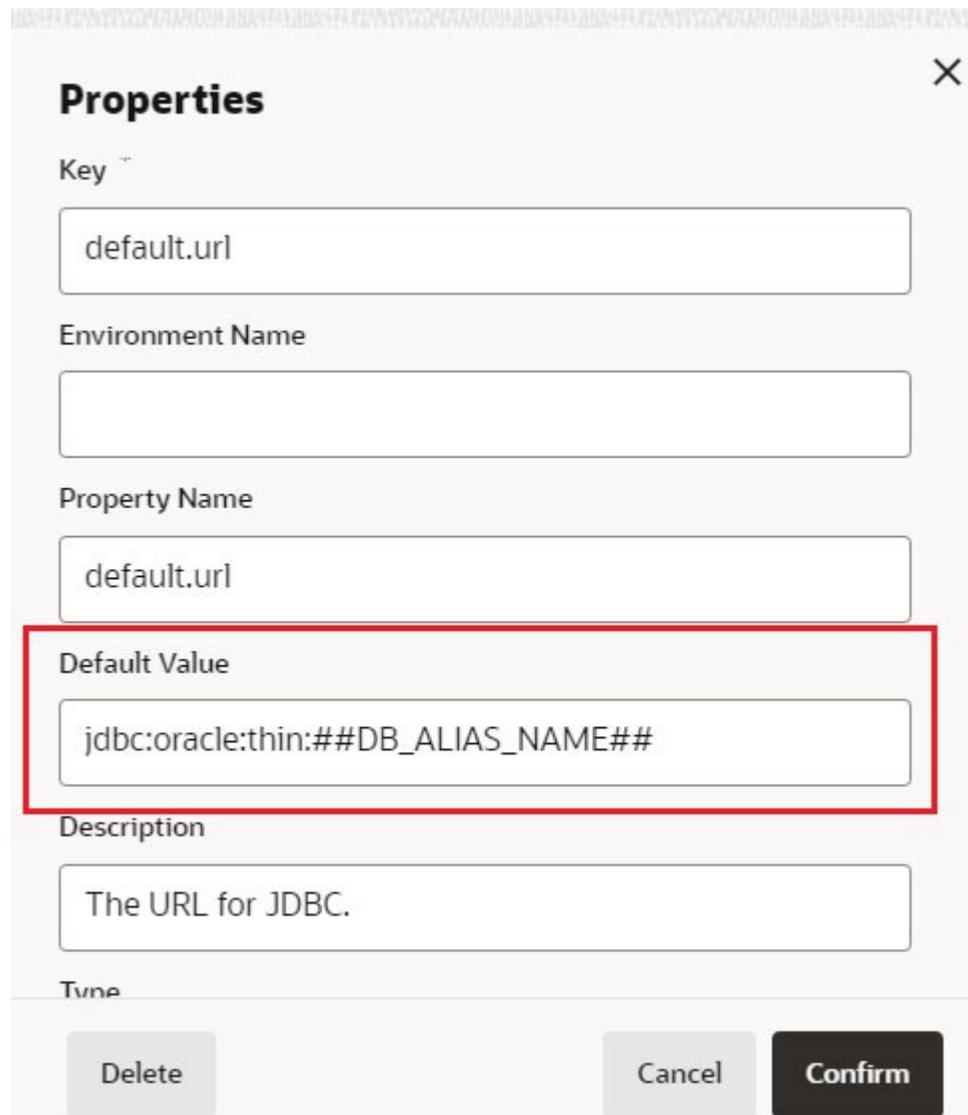
11. On Interpreter Settings page, expand **Interpreter Client Configurations** and click the Edit icon on the **<Class Name> (zeppelin)**. The Interpreter Client Configurations Window is displayed.

Figure 9-20 Interpreter Client Configurations



- Click **default.url** under the Properties. The Properties page is displayed.

Figure 9-21 Properties



The screenshot shows a 'Properties' dialog box with a close button (X) in the top right corner. The dialog contains several fields:

- Key**: A text box containing 'default.url'.
- Environment Name**: An empty text box.
- Property Name**: A text box containing 'default.url'.
- Default Value**: A text box containing 'jdbc:oracle:thin:##DB_ALIAS_NAME##'. This field is highlighted with a red rectangular border.
- Description**: A text box containing 'The URL for JDBC.'
- Type**: A label with no associated text box.

At the bottom of the dialog, there are three buttons: 'Delete' (disabled), 'Cancel' (disabled), and 'Confirm' (active).

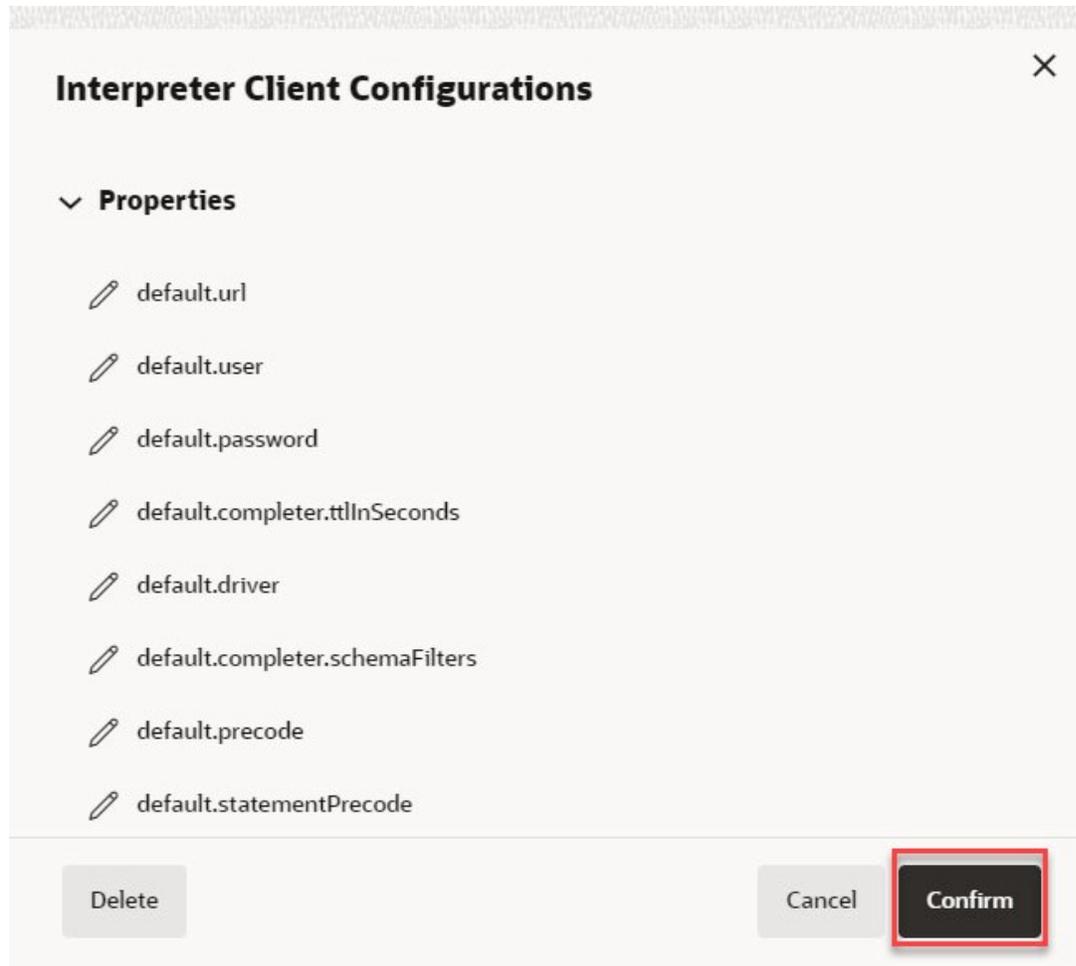
- Enter the alias name in the **Default Value** field.

The alias name is available in the `<MMG_STUDIO_INSTALLATION_PATH>/wallet/tnsnames.ora` directory.

For example, `jdbc:oracle:thin:##DB_ALIAS_NAME##`

- Click **Confirm**. The Interpreter Client Configurations page is displayed.

Figure 9-22 Interpreter Client Configurations



15. Click `default.user` property and it should be null in the **Default Value** field.
16. Click `default.password` property and it should be null in the **Default Value** field.

Note

Retain the default settings for the remaining properties in the Interpreter Client Configurations.

17. Click **Update**. The modified values are updated in the Interpreter.

Link Wallet Credentials to jdbc Interpreter

Note

This feature is not recommended approach because it can only be used to connect to a single schema, and all users will have access to that, rather than access being managed per user. In future releases this interpreter will not be enabled by default but instructions will be given to enable if required.

Limitation

- Data source configuration is not dynamic; instead, it is static from the Interpreter Configuration screen.
- There is no restriction or secure access of data provided with this interpreter.

Recommendation

Users are recommended to use a python interpreter to get dynamic data source configuration; even data access permission features can also be used with this interpreter.

OFS MMG provides secure and safe credential management. Examples of credentials are passwords, Oracle Wallets, or KeyStores. To link credentials (a wallet and a password) to the jdbc interpreter variant to enable secure data access. This linking enables the jdbc interpreter to securely connect to the specified Oracle database. For more information on linking Wallet Credentials to jdbc Interpreter, see the Link Credentials section.

Note

The Credentials section is enabled if an interpreter variant can accept credentials.

You can also create new credentials and link to jdbc Interpreter. For more information, see Create a Credential section.

md Interpreter

This Interpreter displays text based on Markdown, which is a lightweight markup language. In the md interpreter, you can configure the markdown parser type. Markdown (md) is a plain text formatting syntax designed so that it can be converted to HTML. Use this section to configure the markdown parser type.

To configure the md interpreter variant, follow these steps:

1. On the md Interpreter page LHS menu, select md. The md interpreter pane is displayed.
2. On the Interpreter Settings page, expand **Interpreter Client Configurations** and click the Edit icon for **<Class Name> (zeppelin)**. The Interpreter Client Configurations Window is displayed.
3. Enter the markdown parser type and click **Update**. To confirm the modified configuration.

PGX Interpreter

The PGX has the following interpreters:

- **pgql**: The `pgql` interpreter is a ready-to-use interpreter used to connect the configured PGX server. This Interpreter is used to perform queries on the graph in OFS MMG. PGQL is a graph query language built on top of SQL, bringing graph pattern matching capabilities to existing SQL users and new users interested in graph technology but who do not have an SQL background.
- **pgx-algorithm**: The `pgx-algorithm` is a ready-to-use interpreter used to connect to the configured PGX server. This Interpreter is used to write an algorithm on the graph and is also used in the PGX interpreter.
- **pgx-java**: The `pgx-java` interpreter is a ready-to-use interpreter used to connect to the configured PGX server. It is **Java11** based interpreter with a PGX client embedded in it to query on graph present in the PGX server.
- **pgx-python**: The `pgx-python` interpreter is a ready-to-use interpreter used to connect to the configured PGX server. It is a **python** based interpreter with a PGX python client embedded in it to query on graph present in the PGX server. By default, this Interpreter points to ml4aml Python Virtual environment.
- **java**: The `java` interpreter is a ready-to-use interpreter based on **Java11**, where users can write the java code.

To configure the `pgql` interpreter variant, follow these steps:

1. On the Interpreter page LHS menu, select `pgql`. The `pgql` interpreter pane is displayed.
2. On the Interpreter Settings page, expand **Interpreter Client Configurations** and click the Edit icon for **<Class Name> (zeppelin)**. The Interpreter Client Configurations Window is displayed.
3. Enter the following information in the `pgql` interpreter variant pane as tabulated in the following table.

Table 9-4 PGX interpreter

Field	Description
<code>graphviz.formatter.class</code>	Enter the class which implements the formatting of the visualization output. For example, <code>oracle.datastudio.graphviz.formatter.DataStudioFormatter</code>
<code>graphviz.driver.class</code>	Enter the class which implements the PGQL driver. For example, <code>oracle.pgx.graphviz.driver.PgxDriver</code>
<code>base_url</code>	Enter the base URL of the PGX. For example, <code>http://<HOSTNAME>:7007</code>
<code>zeppelin.interpreter.output.limit</code>	Enter the output message limit. Any message that exceeds the limit is truncated. For example, 102 or 400.

Table 9-4 (Cont.) PGX interpreter

Field	Description
num_cached_resultsets	Maximum number of results sets kept open on the PGX server per interpreter session. Only checked when the interpreter is used, and therefore it should only be used with expiring interpreter sessions. For example: 5
resultset_expiration_time_secs	Number of seconds after which unused results sets are closed on the PGX server. Only checked when interpreter session is used and should only be used with expiring interpreter sessions. For example: 3600
zeppelin.python.useIPython	Set to 'True' to use IPython, else set to 'False'.
zeppelin.python	Enter the Python installed path. The value points to the default Python version set for the Interpreter. Note: To use a different Python version, see Change Version in the Python Interpreter section.

Spark Interpreter

This section explains about Spark Interpreter configurations.

Spark Interpreter in Local Mode

To start spark interpreter in the local mode, follow these steps:

1. Download `spark-3.0.3-bin-hadoop2.7.tgz` from the [website](#).
2. Unzip the spark hadoop cluster's zip file in the below mentioned locations:
 - `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs`
 - `<MMG_INSTALLATION_PATH>/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs`
3. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/bin` directory.
4. Open the `startup.sh` file and add following line before the line containing `"counter=1";nohup "$DIR"/../interpreter-server/spark-interpreter-<version>/bin/spark-interpreter &>> <path_to_save_the_logs>/<log_file_name>.log &`

Figure 9-23 Snapshot of startup.sh file

```
#export PLAINR_INTERPRETER_OPTS="$PLAINR_INTERPRETER_OPTS -DAPP_BASE_NAME='plainr-i
#nohup "$DIR"/../interpreter-server/plainr-interpreter-23.4.2/bin/plainr-interprete
# To start Spark interpreter
nohup "$DIR"/../interpreter-server/spark-interpreter-23.4.0/bin/spark-interpreter &
counter=1;
while [[ $counter -lt 20 ]]
do
  dsHealth=`curl -s --insecure https://ofss-mum-1779.snbomprshared1.gbucdsint02bc
```

5. Save and close the file.
6. Open the shutdown.sh file and add following line before the line containing "SL=".

```
I7014=`ps -eaf | grep java | grep RemoteInterpreterServer | grep 7014 |
awk '{print $2}'`
if [[ "" != "$I7014" ]];
then kill -9 $I7014;
fi
```

Figure 9-24 Snapshot of shutdown.sh file

```
# To shutdown Spark interpreter
I7014=`ps -eaf | grep java | grep RemoteInterpreterServer | grep 7014 | awk '{print $2}'`
if [[ "" != "$I7014" ]];
then kill -9 $I7014;
fi
SL=`ps -eaf | grep java | grep oracle.datastudio.starter.App | awk '{print $2}'`
if [[ "" != "$SL" ]];
then kill -9 $SL;
fi
```

Note

In the above step, the port number for the spark interpreter is assumed to be 7014, the default port that comes with the installer. If a different port is used, then change the configuration accordingly.

7. Save and close the file.
8. Navigate to <MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ bin directory.
9. Open the startup.sh file, navigate to line 29 and update spark value as **7014**.

For example: `./"$DIR"/datastudio --port 7008 --markdown 7009 --spark 7014 --python 7012 --jdbc 7011 --shell -1 --pgx 7022 --external`

10. Navigate to <MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ bin directory.
11. Open the config.sh file and update the following parameters:
 - MMG_SPARK_ENABLED=true

- `SPARK_HOME=<MMG_INSTALLATION_PATH>/deployed/mmg-home/ mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs/ spark-<version>-bin-hadoop<version>`
- `HADOOP_HOME=##HADOOP_HOME##`

Note

Retain the placeholder as it is.

- `SPARK_MASTER=local`
- `SPARK_DEPLOY_MODE=`

Note

Retain the `SPARK_DEPLOY_MODE` as **blank**.

- `DATASTUDIO_SPARK_INTERPRETER_PORT=7014`

12. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/bin` directory.

13. Open the `config.sh` file and update the following parameters:

- `MMG_SPARK_ENABLED=true`

Note

By default, it is set to false. You can configure the following parameters only when **MMG_SPARK_ENABLED** is set to **true**.

- `SPARK_HOME=<MMG_INSTALLATION_PATH>/deployed/mmg-home/ mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs/ spark-<version>-bin-hadoop<version>`
- `HADOOP_HOME= ##HADOOP_HOME##`

Note

Retain the placeholder as it is.

- `SPARK_MASTER= local`
- `SPARK_DEPLOY_MODE=`

Note

Retain the `SPARK_DEPLOY_MODE` as blank.

- `DATASTUDIO_SPARK_INTERPRETER_PORT=7014`

14. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/builtin/interpreters/spark.json` directory.

15. Navigate to line 169 and update port value as **7014**.

16. Update **default value** as **local** for `spark.master` and **blank** for `spark.submit.deployMode`.

For example:

```
"spark.master": {
    "envName": "MASTER",
    "propertyName": "spark.master",
    "defaultValue": "local",
    "description": "Spark master uri. ex) spark://
masterhost:7077",
    "type": "string"
},
"spark.submit.deployMode": {
    "envName": null,
    "propertyName": "spark.submit.deployMode",
    "defaultValue": "",
    "description": "The deploy mode of Spark driver
program, either 'client' or 'cluster'",
    "type": "string"
},
```

17. Navigate to `<CMMG_INSTALLATION_PATH>/bin` directory.
18. Restart OFS MMG using the following command.

```
./MMG.sh -restart
```

19. Verify if the `spark-interpreter` has started using the following command:

```
netstat -nltp | grep 7014
```

Enable Additional Spark Interpreter

Interpreter variant does not apply to spark interpreters. Hence, you must enable an additional set of interpreters.

To enable an additional spark interpreter, see [Enable Additional Spark Interpreter](#) section in the Appendix.

Pyspark Interpreter

OFS MMG uses PySpark 2.4.0. Before you begin the configurations, check the prerequisites depending on your operation mode.

Prerequisites

The PySpark interpreter has the same prerequisites as that as the Spark Interpreter. For more information, see [Spark Interpreter](#). Also, all Spark components must be configured to use the same Python version.

Configuration

The PySpark interpreter can be configured through the Spark interpreter, with the only exception being the Python version used. By default, the Python version is set to 3 that can be changed either in the interpreter JSON files before the startup or from the Interpreters page of the OFS MMG application UI during runtime by changing the following properties:

To change the value of the `spark.pyspark.python` property before installing the OFS MMG, follow these steps:

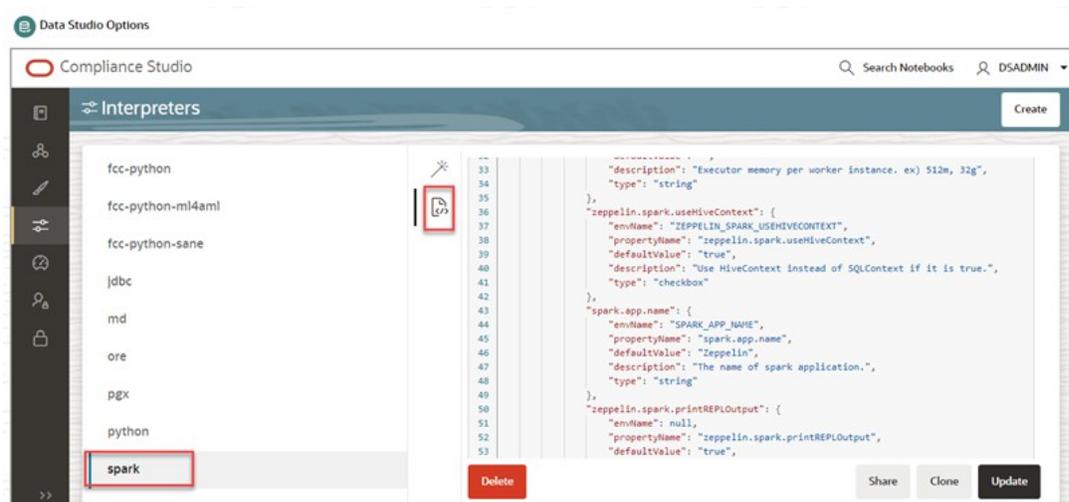
1. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/ builtin/interpreters/spark.json` directory.
2. Update the value in `spark.pyspark.python` property of the `spark.json` file.

To change the value of the `spark.pyspark.python` property after installing the OFS MMG, follow these steps:

1. Login to the MMG application.
2. Launch the **CS Production** Workspace.
3. Click the **User Profile** drop-down list and select **Data Studio Options**.
4. Click Interpreters.

By default, the Interpreters page lists all the available interpreters on the LHS.

Figure 9-25 Spark Interpreter



5. Click **spark** interpreter on the LHS and then click **Plain Configuration** on the RHS.
6. Update the value in the `spark.pyspark.python` property and click **Update**.

In the **Spark Interpreter Settings** page of the OFS MMG application UI (or `spark.json` file), change the value of the `spark.pyspark.python` property to the Python executable path that is to be used by the Spark executors.

In the **PySpark Interpreter Settings** page of the OFS MMG application UI (or `pyspark.json` file), change the value of the `zeppelin.pyspark.python` property to the Python executable path that is to be used by the Spark driver.

Use Python Virtual Environments with PySpark

To ensure that the two Python versions match, in case your components run on different machines, you must use the Python virtual environments with PySpark.

Create a Virtual Environment with Conda

Note

You can also use **virtualenv** to create your virtual environment instead of **conda**.

To create a virtual environment with Conda, follow these steps:

1. Ensure that you have conda and conda-Pack installed.

Note

To check if conda is installed, then execute the following command: "conda --version"

2. Create your virtual environment using the following command:

```
conda create -y -n <environment-name> python=<python-version>
<additional-packages>
```

Note

The <environment-name> can be chosen freely and subsequently has to be substituted in further commands.

3. Activate your virtual environment using the following command:

```
conda activate <environment-name>
```

4. Execute the following to obtain the path to your virtual environment:

```
which python
```

The obtained result is referred to as <environment-abs-path>.

5. Compress your virtual environment using the following command:

```
conda pack -n <environment-name> -o <environment-abs-path>/
<environmentname>.
tar.gz
```

Update Interpreter Properties

The interpreter properties can either be configured in the interpreter JSON files or from the Interpreters page of the OFS MMG application UI after starting the OFS MMG application.

- In the Spark Interpreter Settings page of the OFS MMG application UI (or `spark.json`), change the following:
 - Change the value of the `spark.yarn.dist.archives` property to `<environment-abspath>/<environment-name>.tar.gz#<environment-name>`
 - Change the value of the `spark.pyspark.python` property to `./<environmentname>/bin/python`

- In the **PySpark Interpreter Settings** page of the OFS MMG application UI (or `pyspark.json`), change the value of the `zeppelin.pyspark.python` parameter to `<environment-abs-path>/bin/python`.

Configure Pyspark Interpreter

Users must write for connection either in the Initialization section or in the notebook's paragraph. This interpreter is used to write the pyspark language to query and perform analytics on data present in big data. This requires additional configuration, which must be performed as a prerequisite or as postinstallation with the manual change of interpreter settings.

In the pyspark interpreter, you can configure the Python binary executable for PySpark in both driver and workers, set 'True' to use IPython, else set it to 'False'.

To configure the pyspark interpreter variant, follow these steps:

1. On the Interpreter page LHS menu, select pyspark. The pyspark interpreter pane is displayed.
2. On the Interpreter Settings page, expand **Interpreter Client Configurations** and click the Edit icon for **<Class Name> (zeppelin)**. The Interpreter Client Configurations Window is displayed.
3. Enter the following information in the pyspark interpreter variant pane as tabulated in the following table

Table 9-5 pyspark interpreter

Field	Description
<code>zeppelin.pyspark.python</code>	Enter the Python binary executable for PySpark in both drivers and workers. The default value is <code>python</code> . For example, <code>python</code>
<code>zeppelin.pyspark.useIPython</code>	Set to 'True' to use IPython, else set to 'False'.
<code>zeppelin.interpreter.output.limit</code>	Output message from interpreter exceeding the limit will be truncated

Enable Additional PySpark Interpreter

Interpreter variant does not apply to pyspark interpreters. Hence, you must enable an additional set of interpreters.

To enable an additional pyspark interpreter, see Enable Additional Spark or PySpark interpreter section in the Appendix.

R Interpreter

The R interpreter allows execution of R code within OFS MMG interactive notebooks and pipelines, supporting exploratory analytics, reproducible research, and statistical computing, while integrating seamlessly with big data tools to provide an interactive environment for advanced analytics.

You can configure the R Interpreter support either with **ORD-3.6.1** or **R 4.1.2**.

ORD-3.6.1 Installation

To install ORD-3.6.1, follow the steps:

1. Check Linux version. For example, `cat /etc/os-release`
You can install R interpreter based on the following Linux version.
 - a. To install Oracle R Distribution on Linux 7, see [Using Yum](#).
 - b. To install Oracle R Distribution on Linux 8, see [Using Yum or Dnf](#).
2. In the terminal, check installation using **R -version**.

Figure 9-26 R Interpreter

```
(base) [fbcstudio@ofss-mum-1755 conf]$ R -version
WARNING: unknown option '-version'

Oracle Distribution of R version 4.0.5  (--) -- "Shake and Throw"
Copyright (C)  The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

You are using Oracle's distribution of R. Please contact
Oracle Support for any problems you encounter with this
distribution.
```

3. To install other packages, execute the following.
 - a. `R-e "install.packages('Rserve', repos='https://www.rforge.net/')"`
 - b. `R-e "install.packages(c('knitr', 'ggplot2', 'backports'), repos='https://mirror.las.iastate.edu/CRAN/')"`

R 4.1.2 Installation

Note

This setup might update some of the older root level files and using Non-Oracle Yum Repository for getting R rpm files.

To install R 4.1.2, follow these steps:

1. In the terminal, execute the following.
 - a. `curl- O https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
 - b. `yum install epel-release-latest-7.noarch.rpm`

- c. `curl -O https://cdn.rstudio.com/r/centos-7/pkgs/R-
{R_VERSION}-1-1.x86_64.rpm`
 - d. `sudo yum install R-
{R_VERSION}-1-1.x86_64.rpm`
 - e. `sudo ln -s /opt/R/
{R_VERSION}/bin/R /usr/bin/R`
2. Check installation using **R -version**.
 3. To install other packages, execute the following.
 - a. `R -e "install.packages('Rserve', repos='https://www.rforge.net/')`
 - b. `R -e "install.packages(c('knitr', 'ggplot2', 'backports'), repos='https://
mirror.las.iastate.edu/CRAN/')`

Configuration

This section describes about how to configure R interpreter.

To configure R interpreter, follow these steps:

1. Navigate to the `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf` directory.
2. To configure Rserve, execute the following.

```
nano /scratch/software/R/Rserve.conf (sample file)
>
auth required
plaintext disabled
pwdfile /scratch/software/R/creds/Rserve.pwd
remote enable
switch.qap.tls enable
tls.port 6311
qap disable
interactive no
rsa.key /scratch/software/R/creds/server.key
tls.key /scratch/software/R/creds/server.key
tls.cert /scratch/software/R/creds/server.crt
```

3. Create a password file (Rserve.pwd) using the following.

```
>
oml $5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8
```

The file contains one line per user, where the first part is the username, and the second part is the password. The password can either be plain text or a MD5/SHA1 hash. In this example the password ``password`` is hashed with SHA1. If you use hashed passwords, the password string needs to start with a ``$`` sign.

4. To generate the SSL key, execute the following.

```
openssl genrsa -out server.key 2048
openssl req -new -key server.key -out server.csr # password 1234
openssl x509 -req -days 365 -in server.csr -signkey server.key -out
server.crt
```

5. To generate a keystore file, execute the following.

```
keytool -import -alias <keystore-alias> -file <path-to-server.crt>/
server.crt -keystore <output-path-to-keystore/rinterpreterkeystore -
storepass <keystore-secret> -noprompt
```

For example,

```
keytool -import -alias <rserve -file> <MMG_INSTALLATION_PATH>/deployed/mmg-
home/mmg-studio/conf/Rserver.crt -keystore <MMG_INSTALLATION_PATH>/
deployed/mmg-home/mmg-studio/conf/rinterpreterkeystore -storepass changeit
-noprompt
```

After execution, the following files are generated in the <MMG_INSTALLATION_PATH>/
deployed/mmg-home/mmg-studio/conf directory.

- rinterpreterkeystore
- Rserve.conf
- Rserve.pwd
- Rserver.crt
- Rserver.csr
- Rserver.key

6. Open the Rserve.conf file and update the following:

Table 9-6 Rserve.conf File

Parameter	Value
rsa.key	<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/ conf/Rserver.key
tls.key	<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/ conf/Rserver.key
tls.cert	<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/ conf/Rserver.crt
maxbufsize	1000000004 in bytes

7. Navigate to the <MMG_INSTALLATION_PATH>/bin directory.
8. Open config.sh file and update the following parameters:
 - a. **R_ENABLED=true**
 - b. **RS_CONF_PATH=<MMG_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/
Rserver.conf**
 - c. **RS_KEYSTORE=<MMG_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/
rinterpreterkeystore**

- d. **RS_KS_SECRET=changeit**
9. Run OFS MMG. To run, execute the following command.


```
./MMG.sh --update
```
10. Restart OFS MMG. The R interpreter will be visible in the UI and you can execute the R paragraph.
11. Navigate to the `<MMG_INSTALLATION_PATH>/deployed/mmg-home/bin` directory.
12. Open `config.sh` file and validate the following parameters:
 - a. **R_ENABLED=true**
 - b. **RS_CONF_PATH=<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf/Rserver.conf**
 - c. **RS_KEYSTORE=<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf/rinterpreterkeystore**
 - d. **RS_KS_SECRET=changeit**
13. Navigate to the `<MMG_INSTALLATION_PATH>/mmg-home/bin` directory.
14. Open `config.sh` file and validate the following parameters:
 - a. **R_ENABLED=true**
 - b. **RS_CONF_PATH=<MMG_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/Rserver.conf**
 - c. **RS_KEYSTORE=<MMG_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/rinterpreterkeystore**
 - d. **RS_KS_SECRET=changeit**

If users get an SSL error while executing the R interpreter, follow these steps:

1. Log in to the Linux server as root and change to the `/etc/yum.repos.d` directory.
2. Download [Rserve_1.8-10.tar.gz](#).
3. Install `openssl-devel` - `yum install openssl-devel`
4. To install, execute the following.

```
R CMD INSTALL --configure-args="--with-ssl-headers=/usr/include --with-ssl-libraries=/usr/lib64" Rserve_1.8-10.tar.gz
```

The following steps can be executed from R session to check that `rserve` configuration is working before starting the services.

5. Open the **Putty** and enter **R -version**.
6. Execute the following command.
 - a. `library(Rserve)`
 - b. `Rserve(args="--RS-conf <MMG_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf/Rserve.conf --no-save")`
7. After verifying, stop `Rserve` and ensure the port is free.
8. Navigate to the `<MMG_INSTALLATION_PATH>/bin` directory.

- Restart OFS MMG.
- Navigate to the interpreter notebook and run the R paragraph to validate the output.

Create a Credential

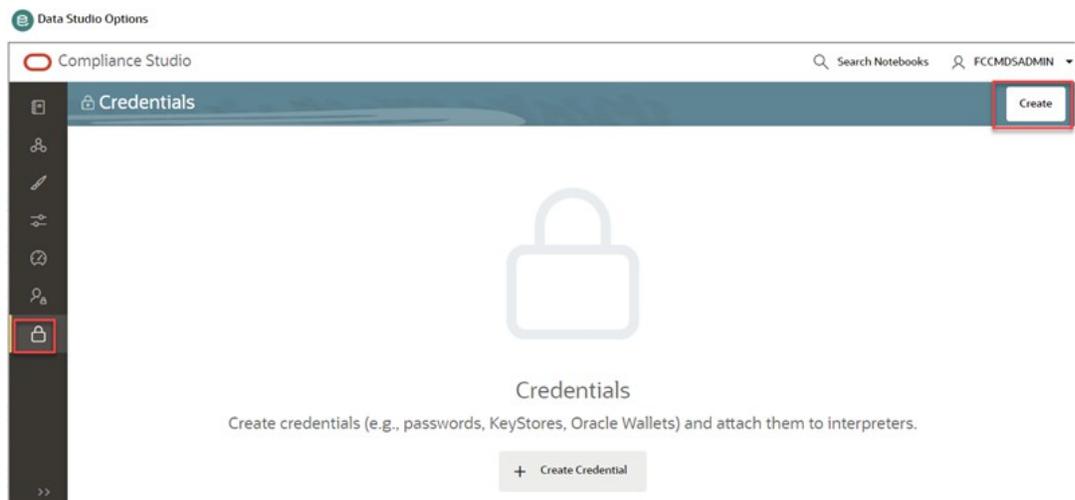
New credentials are created when database details are changed or updated. For example, change in Transparent Network Substrate (TNS) due to hostname change or compulsory periodic update of schema passwords.

Oracle Wallet provides a simple and easy method to manage database credentials across multiple domains. It allows you to update database credentials by updating the Wallet instead of having to change individual data store definitions.

To create a new password credential for the wallet, follow these steps:

- On the Compliance Studio workspace LHS Menu, click **Credentials**. The Credentials page is displayed.

Figure 9-27 Credentials Page



- Click **Create**. The **New Credential** dialog box is displayed.

Figure 9-28 New Credential for Password

3. Enter the following information in the New credential dialog as tabulated in the following table.

Table 9-7 Create Credential dialog

Field	Description
Name	Enter the name for the password credential.
Type	From the drop-down list, select the Password type.
Password	Enter the wallet password for the password credential.
Accessible via APIs in Paragraphs	Move this toggle switch to right to enable this option.

4. Click **Create**. The password is created for the wallet and displayed on the Credentials page.

To create a wallet credential, follow these steps:

1. Click **Create**. The **New Credential** dialog box is displayed.

Figure 9-29 New Credential for Wallet

The screenshot shows a dialog box titled "New credential" with a close button (X) in the top right corner. The dialog contains three input fields: "Name", "Type" (a dropdown menu with "Oracle Wallet" selected), and "File". Below these fields is a dashed box containing the text "Drag and Drop" and "Select a file or drop one here.". At the bottom are two buttons: "Cancel" and "Create".

2. Enter the following information in the New credential dialog as tabulated in the following table.

Table 9-8 Create Credential dialog box

Field	Description
Name	Enter the name for the wallet credential.
Type	From the drop-down list, select the Oracle Wallet type.

Table 9-8 (Cont.) Create Credential dialog box

Field	Description
File	<p>Upload the wallet zip file that includes the following files:</p> <ul style="list-style-type: none"> • tnsnames.ora • ewallet.p12 • cwallet.sso <p>These files are available in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/wallet directory.</p> <p>Note:</p> <ul style="list-style-type: none"> • The wallet file must be in .zip format. • The maximum file size allowed for the credential file is 128Kb.

3. Click **Create**. The wallet credential is created and displayed on the Credentials page.

Link Credentials

Compliance Studio provides secure and safe credential management. Examples for credentials are passwords, Oracle Wallets, or KeyStores. To link credentials (a wallet and a password) to the jdbc interpreter variant to enable secure data access. This linking enables the jdbc interpreter to securely connect to the specified Oracle Database. You can also create new credentials to connect to the new interpreter variants based on your requirement. For more information, see [Create a Credential](#) section.

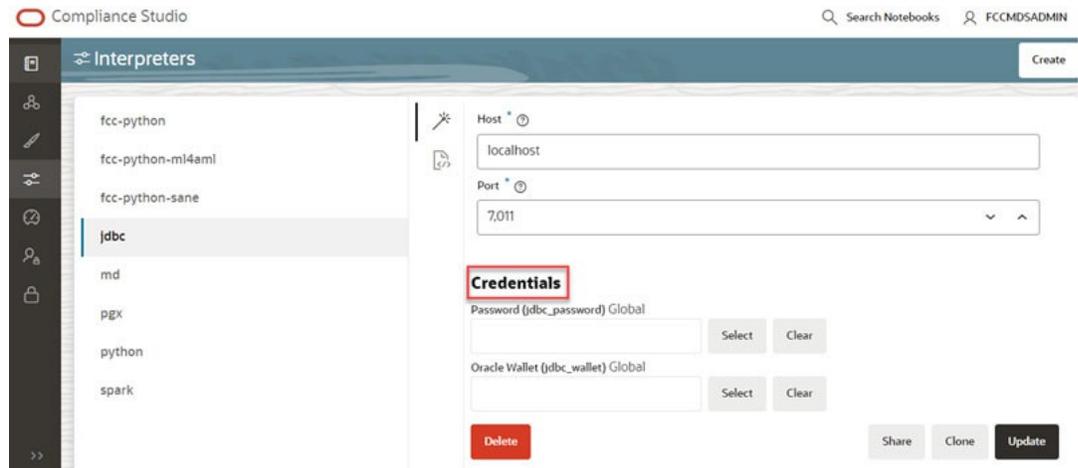
Note

You can link credentials only for jdbc interpreters. The Credential section is enabled if an Interpreter variant can accept credentials.

To link ready-to-use credentials to the required interpreters, follow these steps:

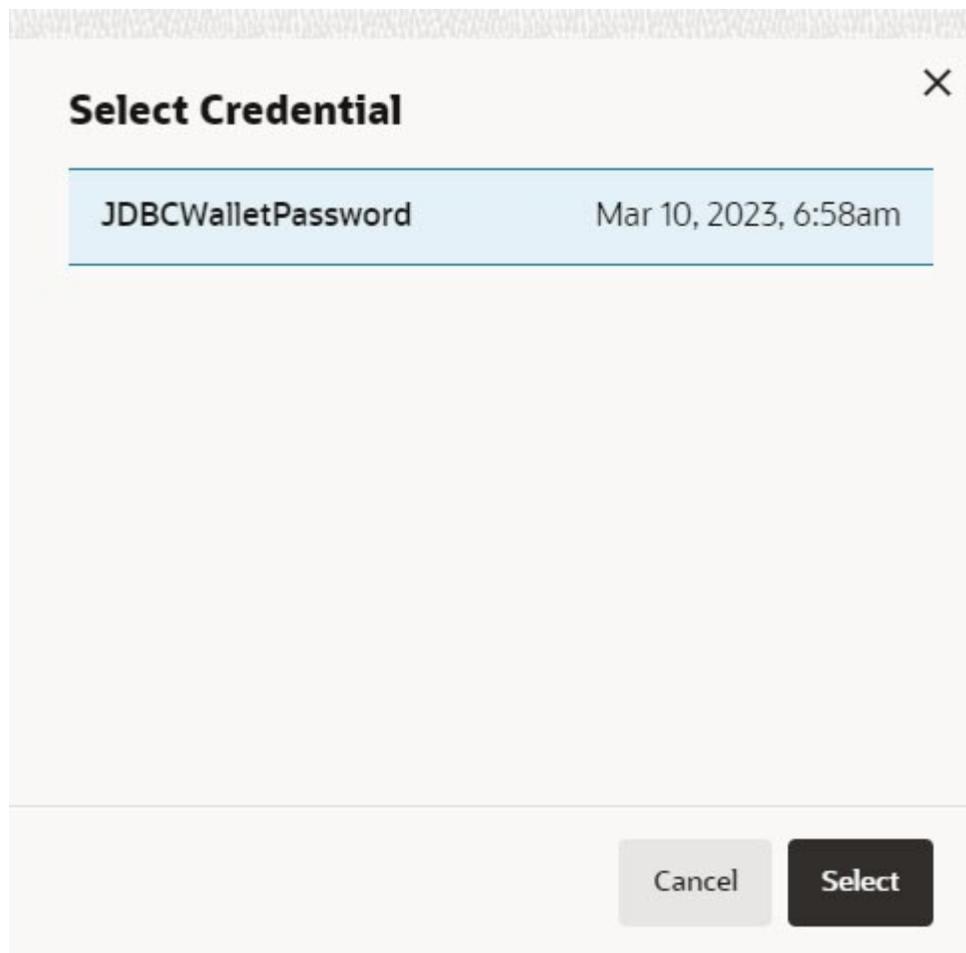
1. On the Interpreters page, select the required interpreters. For example, jdbc.
2. Navigate to the Credentials section.

Figure 9-30 Credentials



3. Click **Select** to select the Password (jdbc password) that you want to link to the Interpreter variant. The Select Credential dialog is displayed.

Figure 9-31 Select Credential



4. Select the required Password (jdbc_password) and click **Select**.

5. Click **Select** on the Credentials section to select the Oracle Wallet (jdbc_wallet) that you want to link to the Interpreter variant. The Select Credential dialog is displayed.
6. Select the required Oracle Wallet (jdbc_wallet) and click **Select**.
7. Click **Update** on the Credentials section to save the changes.
The required password and Oracle Wallet are linked to the jdbc Interpreter.
8. Restart Compliance Studio.

Create an Interpreter Group

In Compliance Studio, you can either use a default interpreter group or create a new group for an interpreter. You can create more than one group for an interpreter. Multiple groups for an interpreter are created to connect different versions of interpreters (Python version: 3, Python version: 2) and connect a different set of users and database schema. For example, Compliance Studio schema, BD schema, etc.

To create a new interpreter group, follow these steps:

1. On the Interpreters page, click the required interpreters from the LHS list. For example, jdbc interpreter.
2. The default interpreter group is displayed on the RHS.
3. On the default interpreter, click **Clone** button to create a new group. The Create Interpreter Group dialog box is displayed.
4. Enter the Name for the new interpreter group. Click **Create**. A new group is created with a name, **<Interpreter Type>.<Group Name>**.
5. Provide the new schema details, such as the default.url, default.user, and default.password.

Clone an Interpreter

You can either use a default interpreter group or create a new group for an interpreter. You can create more than one group for an interpreter.

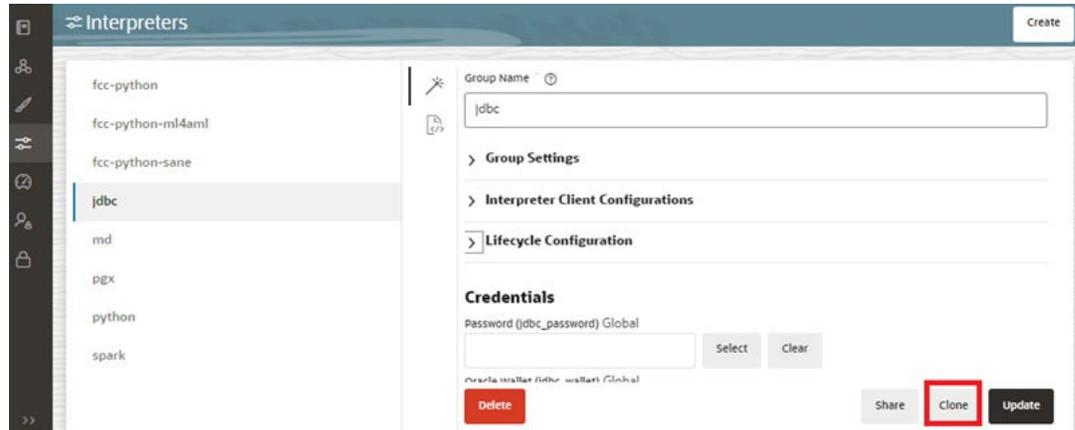
Multiple groups for an interpreter are created to connect different versions of interpreters (Python version: 3, Python version: 2) and connect a different set of users and database schema. For example, Compliance Studio schema, BD schema, etc.

1. Log in to the Compliance Studio application.
2. Launch the **CS Production** Workspace.
3. Click the **User Profile** drop-down list and select **Data Studio Options**.
4. Click **Interpreters**.

By default, the Interpreters page lists all the available interpreters.

5. Click **jdbc** interpreter on the LHS. The default configured interpreter variant is displayed on the RHS:

Figure 9-32 jdbc interpreter screens



6. Click **Clone** on the RHS. The pop-up window displayed for the group name.
7. Enter the group name in the **Group Name** text box and click **Create**. The new group is created and displayed on LHS.
8. Enter **Name** for the new interpreter group.
9. Click **Create**. A new group is created with a name, **<Interpreter Type>.<Group Name>**.
10. Provide the new schema details, such as the default.url, default.user, and default.password.

You can modify the values in the interpreter properties in the JSON file or Wizard view.

10

Data Management

The Data Management module allows users to efficiently manage data stores within a workspace. It provides key features for importing both data models and data from Source Schemas. With features like Data Modelling and Data Population, users can replicate database structures, create data ingestion jobs, and schedule or manually run those jobs to bring in new data. This module simplifies schema integration, supports data migration, and ensures consistent data flow across environments.

Data Management also has a feature where user can see the structure of the database object.

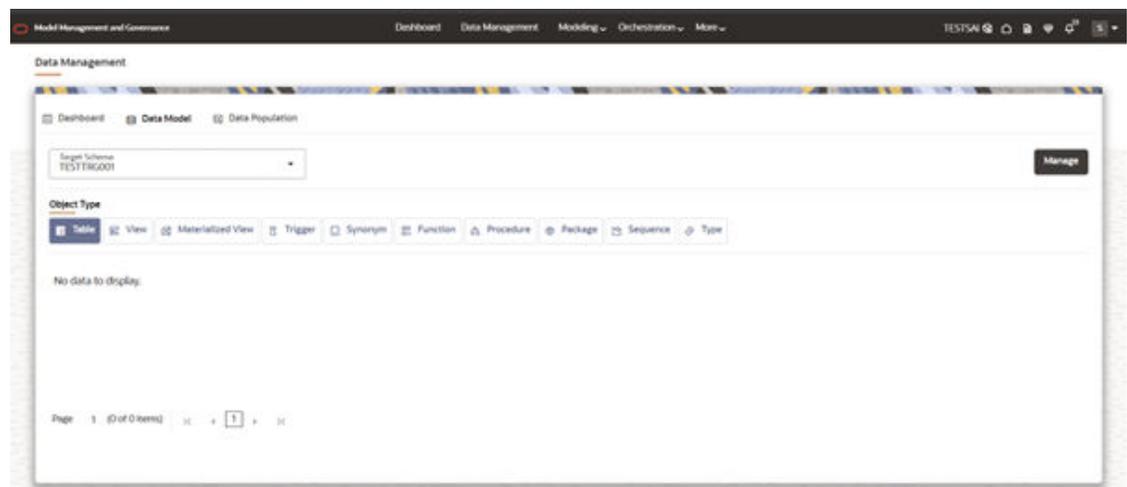
Note

Currently, Hive Data Stores does not support Data Management.

Data Modeling

Data Modeling refers to the process of replicating database objects from a source schema to a target schema. During this process, the structural definitions and metadata of the database objects are transferred to the target environment. This replication ensures that the logical design and structural components of the database are accurately maintained across schemas, supporting development activities for modelers. Database Objects are tables, views, functions, triggers, synonyms, Materialized views, procedures, packages, Sequences, and types.

Figure 10-1 Data Modeling tab



- **Database Objects List menu**
 - The interface displays a list of database object types. Users can select any object type to view the corresponding objects available within the schema.

- **Target Schemas Dropdown**
 - The user can select the target schema from a dropdown menu. Based on the selected target schema, the system will display the corresponding list of database objects—filtered by the selected object type—in the left panel.
- **Database Objects List**
 - A table displays the list of database objects based on the selected object type and schema. This table includes the source schema name, object name, and a **View** option.
 - * The **View** option allows users to view the DDL (Data Definition Language) of the selected database object, along with a **Generic View** (applicable only for tables and views).

Note

The **Generic View** presents a simplified tabular representation containing details such as column name, data type, nullability, and default value.

- **Figure 10-2 Data Definition Language View**

testsrc1 - EMPLOYEE

GENERAL VIEW DDL

Physical Name ↕	Logical Name ↕	Data Type ↕	Nullable ↕	Default Value ↕
NAME	NAME	VARCHAR2	NO	NULL
AGE	AGE	NUMBER	YES	NULL
SALARY	SALARY	NUMBER	YES	NULL

– **Figure 10-3 Generic View**

Physical Name	Logical Name	Data Type	Nullable	Default Value
NAME	NAME	VARCHAR2	NO	NULL
AGE	AGE	NUMBER	YES	NULL
SALARY	SALARY	NUMBER	YES	NULL

- **Tree View and Add Item buttons**

- **Tree View:**

1. Clicking this option opens a new tab within the Data Management UI, titled [TargetSchemaName] - Modeling.
2. This view provides a hierarchical structure to assist users in creating and organizing new data models.

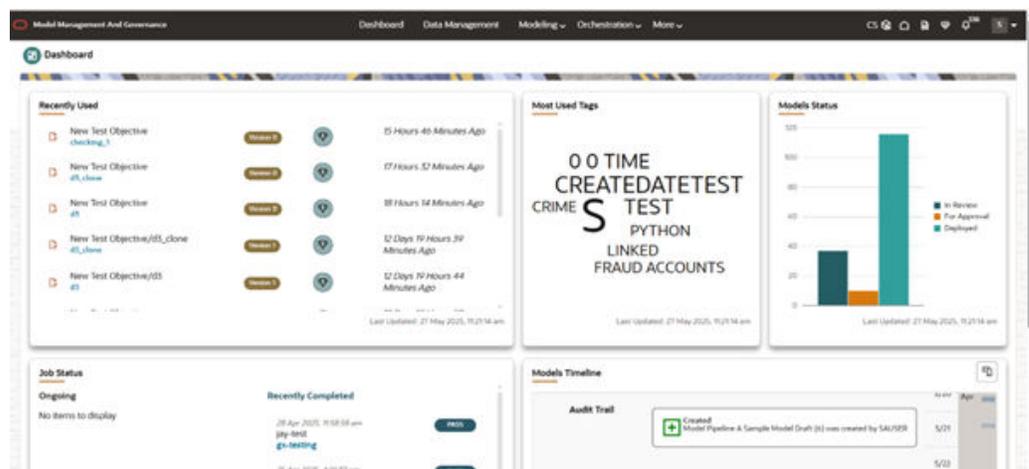
- **Add Item:**

1. This button opens a pop-up window that facilitates data modeling for the selected database object type (as chosen from the left panel).

- **Steps/Process of Data Modeling**

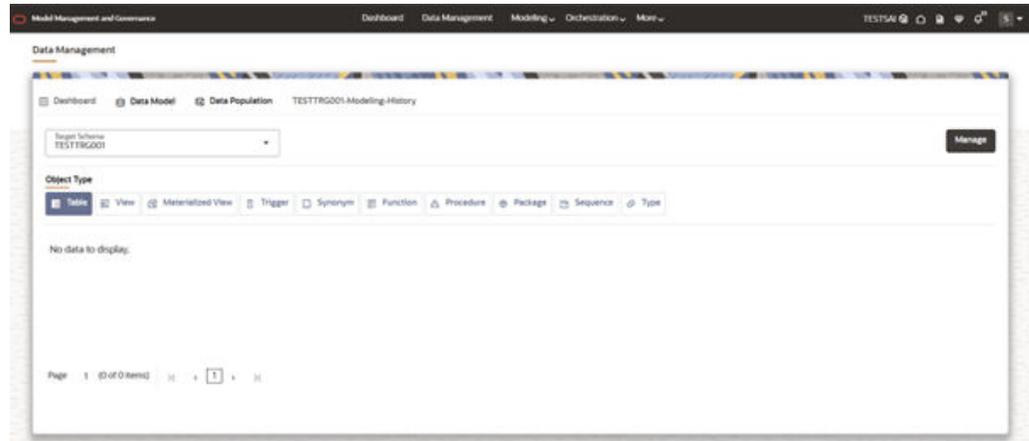
1. Launch any workspace where you wanted to do **Data Modeling** and go to **Data Management** tab from main menu.

Figure 10-4 Data Management tab from main menu



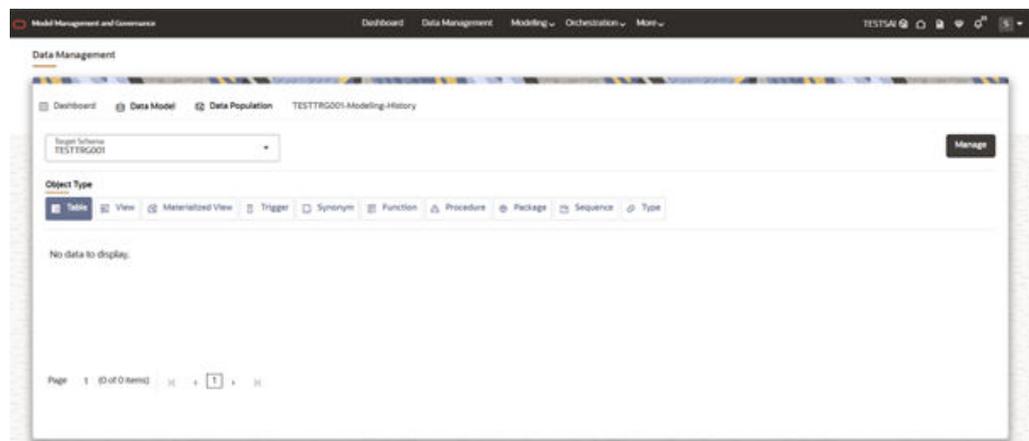
2. Click **Data Model** sub-tab.

Figure 10-5 Data Model sub-tab



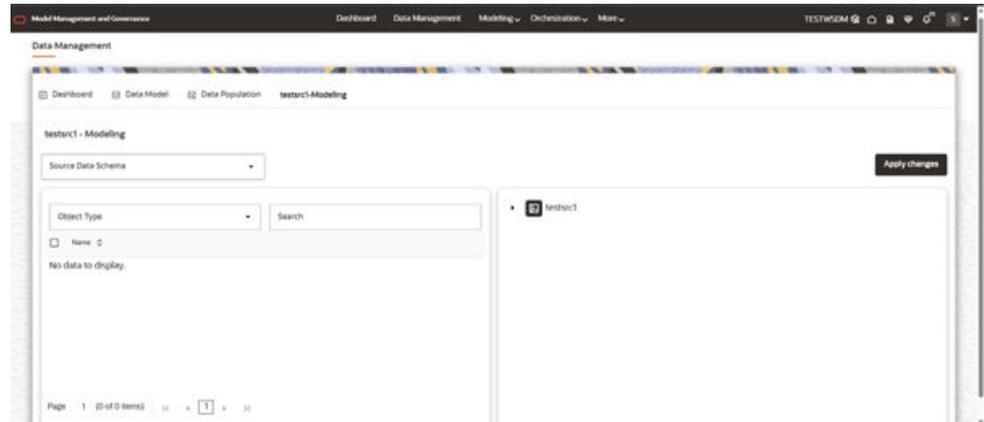
3. This tab contains the **Table Schema** drop-down list, **Object Types** and the list of schemas.
4. Click **Manage** to open a new tab within the MMG- Data Management UI.

Figure 10-6 Manage button tab



5. When a new tab opens, it includes the following components:
 - **Source Schema Drop down:**
 - a. Allows the user to select the desired source schema for Data Modeling. Users can switch between multiple Source Schemas and select the required database objects from each—all in a single Modeling session.
 - **Left panel:**
 - a. Contains an object type drop down and a search bar for filtering.
 - b. Displays a table listing the available database objects based on the selected type and schema.
 - **Right panel:**
 - a. If the user is initiating Data Modeling for the first time, this panel will be empty.
 - b. As the user selects database objects from the Left panel, they will be added to and displayed in the Right panel for further configuration.

Figure 10-7 Left panel and Right panel

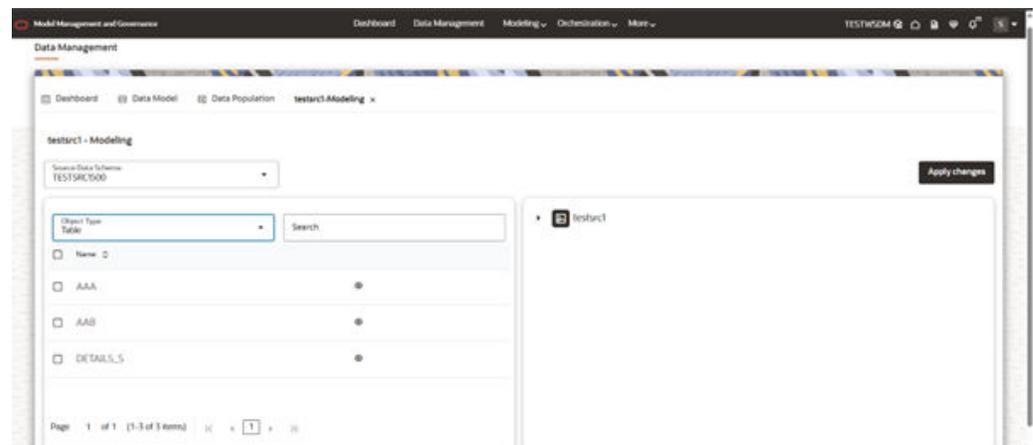


6. Now the user can choose source schema from where user wants to bring in the database objects.
7. After selecting the Source Schema, the user can choose the desired database object type from the left panel. The table in the left panel will then display all available database objects that match the selected type and schema.

i Note

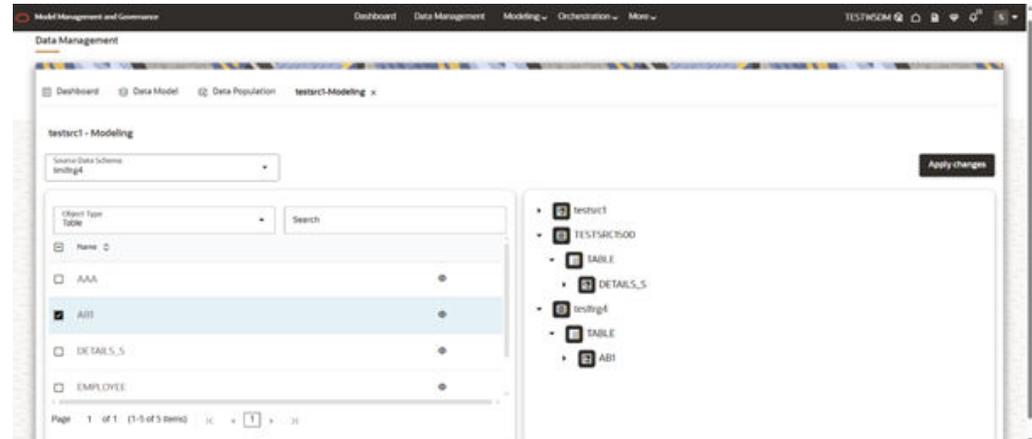
Each database object listed in the table includes an eye icon. Clicking this icon opens a pop-up that provides both the **DDL View** and the **Generic View** of the selected object.

Figure 10-8 DDL and Generic View of the selected object



8. User can choose the required and the selected ones automatically comes into the right panel in following structure.
 - a. **Source Schema-Name1:** DB object type 1, DB Object 1 2, DB Object 2.
 - b. **Source Schema-Name2:** DB object type 1, DB Object 1 2, DB Object 2.

Figure 10-9 Source Schema Name



- Once all the required database objects are selected, the user can click the **Apply Changes** button to initiate the Data Modeling process. On triggering, a confirmation pop-up will be displayed, indicating that the Data Modeling process has started. This tab will automatically close after a few seconds.

Figure 10-10 Apply Changes button

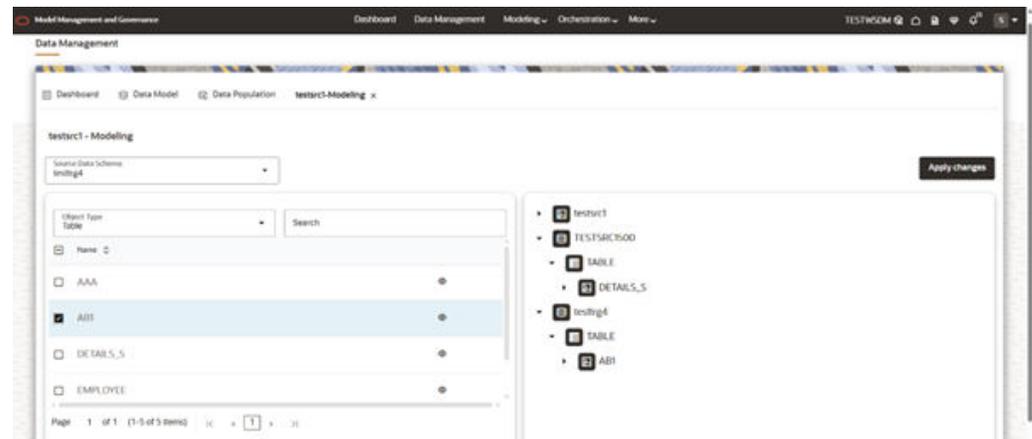
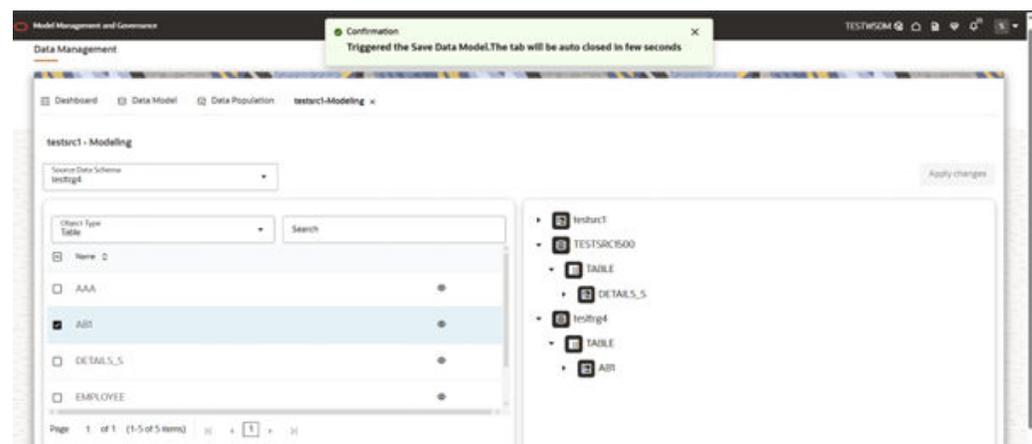


Figure 10-11 Confirmation message



Note

The user can choose database objects from multiple Source Schemas at a time. The user can also choose multiple database object types at a single time.

Note

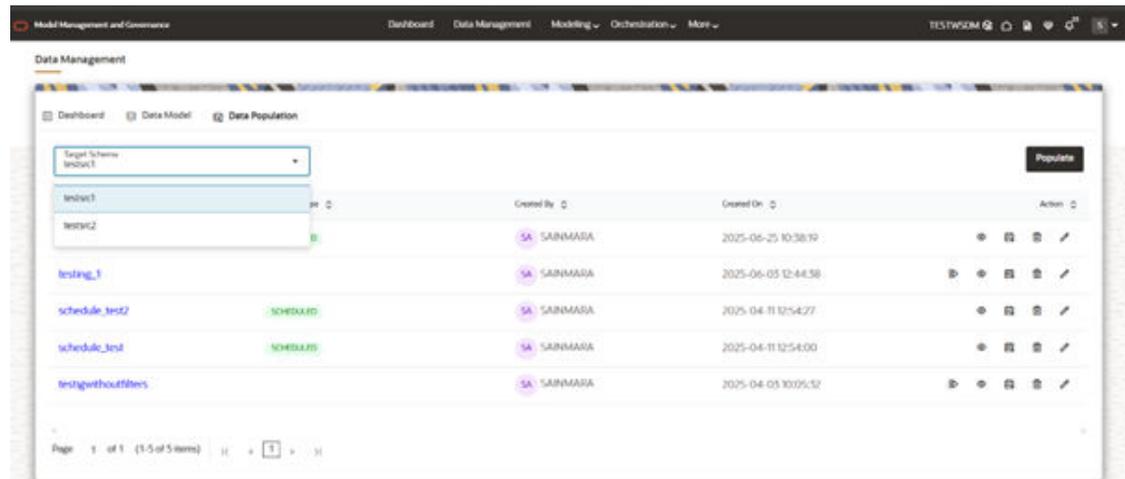
For a selected Target Schema, if any previous Data Modeling is in progress, the user is not allowed to do another Data Modeling.

Data Population

Data Population is about getting the data of tables from Source Schemas to Target Schema by applying the filters at table level and global level. Data is very important aspect of Modeling and we need data to be up to date, so keeping this point we have came up with scheduling idea for Data Population which periodically refreshes the data in the tables.

Data Population tab

Figure 10-12 Target Schema selection and populate button



Target Schemas dropdown

- It will show list of Target Schemas mapped to current workspace.
- We can select any Target Schema, by default 1st one is selected (The one which is selected firstly in workspace creation step 2).
- By choosing one schema, it will show the list of data populations that are created as a table.

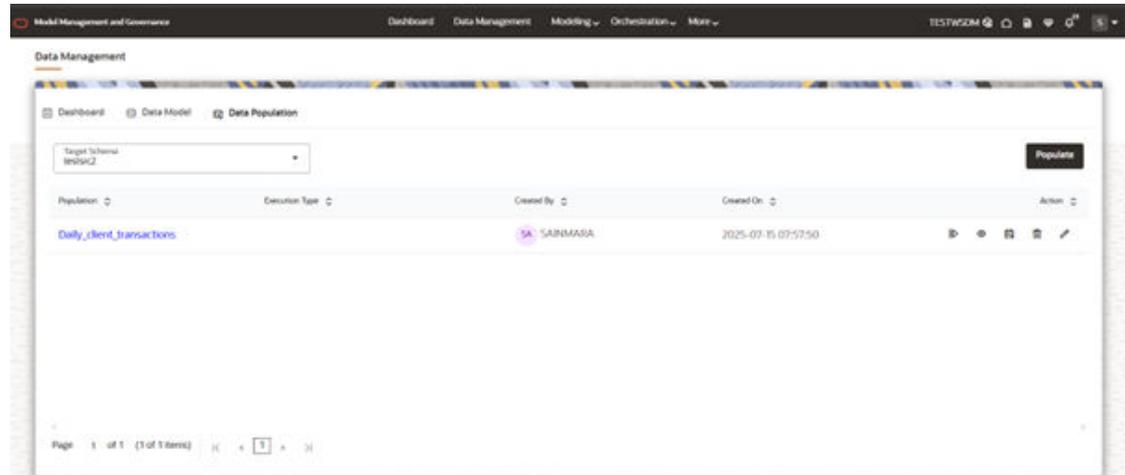
Populate button

- This button helps to create a new Data Population.
- By clicking on this, user will be taken through few steps for creating a Data Population.

Data Population List

- This will show the list of populations that created for the selected Target Schema (From dropdown).

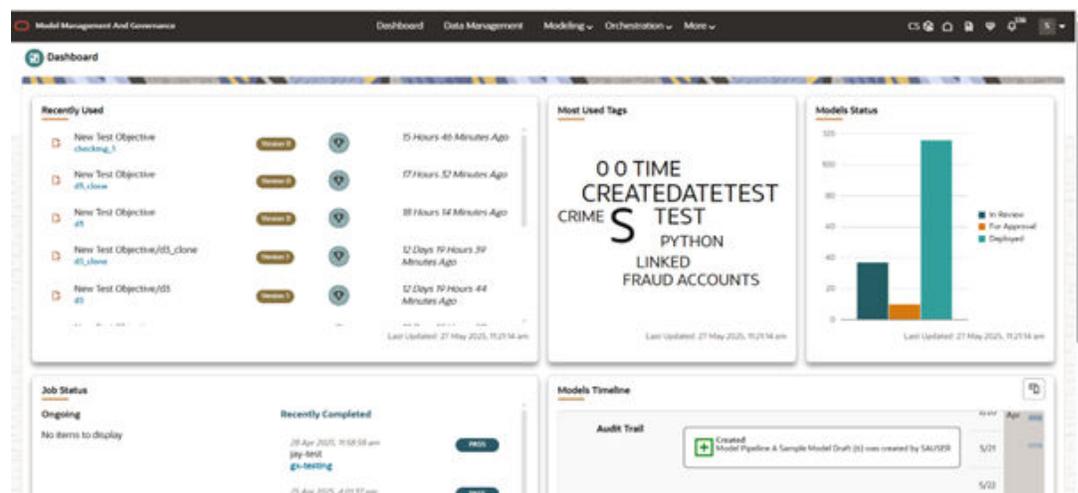
Figure 10-13 Data Population List



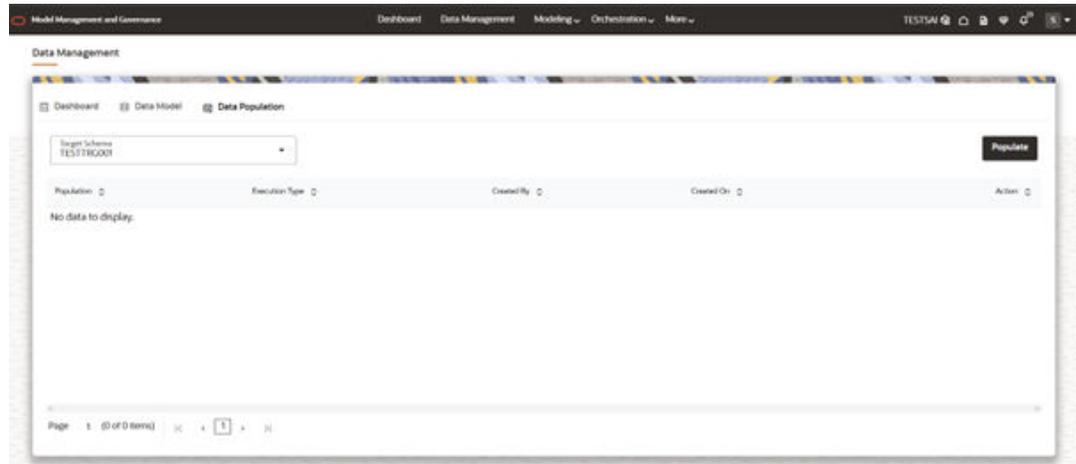
Steps of Data Population

1. Launch any workspace where you wanted to do Data Population and go to Data Management tab from main menu.

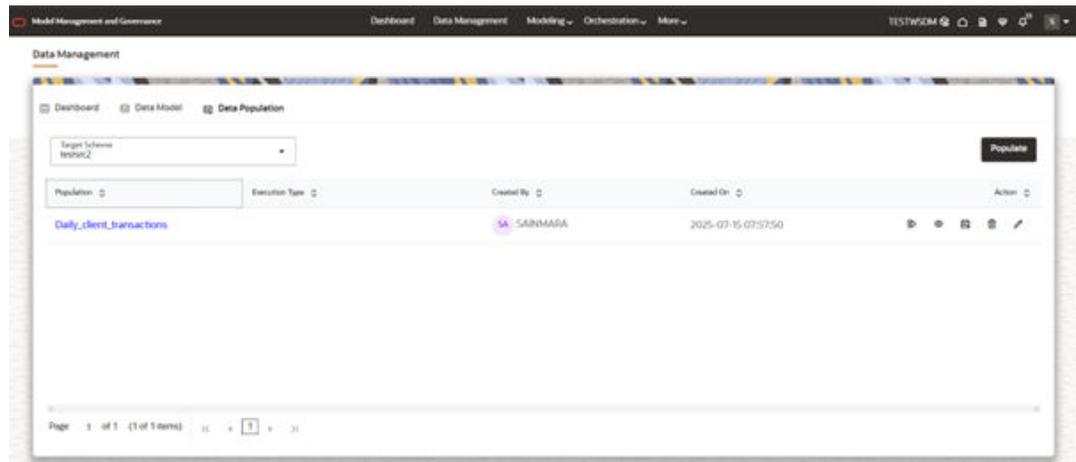
Figure 10-14 Data Population Dashboard



2. Click on Data Population sub-tab.

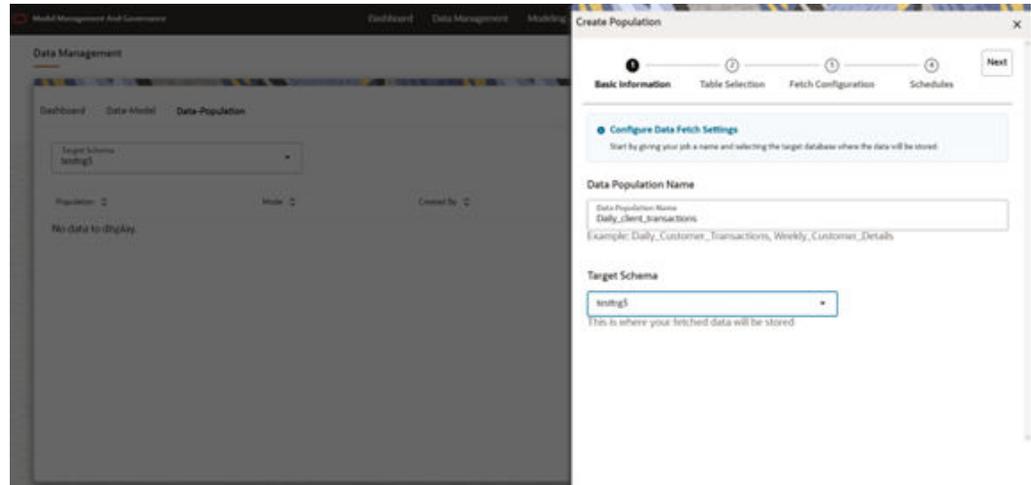
Figure 10-15 Data Population Sub-tab

3. Now the user can see the list of Data Population which are created. When we launch for first time, it will be empty.
4. Click on Populate button, which will take you through the step-by-step process to create a new Data Population.

Figure 10-16 New Data Population

5. Now a drawer will open on the right side where it consists of the following stages:
 - Basic Information
 - Table Selection
 - Fetch Configuration
 - Schedule
6. **Basic Information**
 - Here user need to provide the basic information about Data Population Name and can choose the Target Schema for which user wants to create Data Population.
For example: Data Population Name → Daily_client_transactions, Target Schema → Testtrg3

Figure 10-17 Basic Information



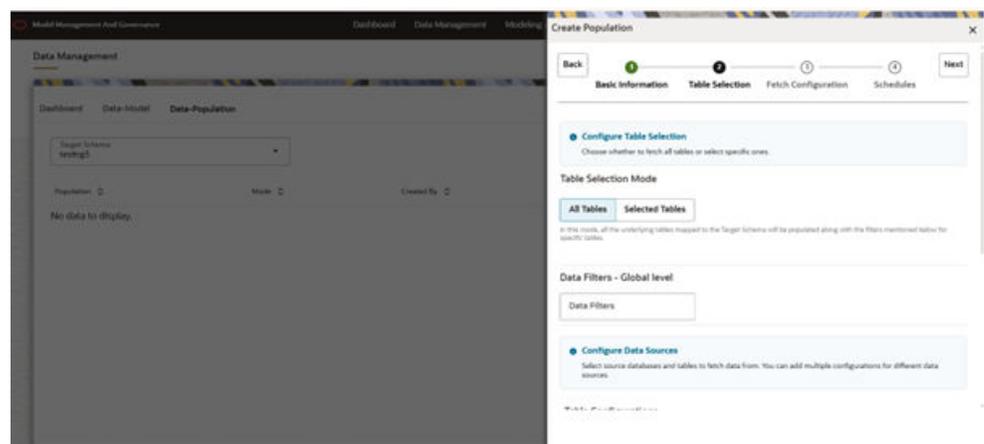
7. Table Selection

- This step includes tables selection mode and table filters to be seeded.
- **Table Selection**
 - Table selection mode is about to refresh All tables data or Selected tables data.
 - We have two modes in this: All tables and Selected tables, where All tables fetches data for all the tables from Source Schema to Target Schema, by applying the filters. Selected tables are the ones that have been selected as part of Table Filter Configuration, for fetching data.

Note

All tables mean the tables that are created by using MMG in Target Schema.

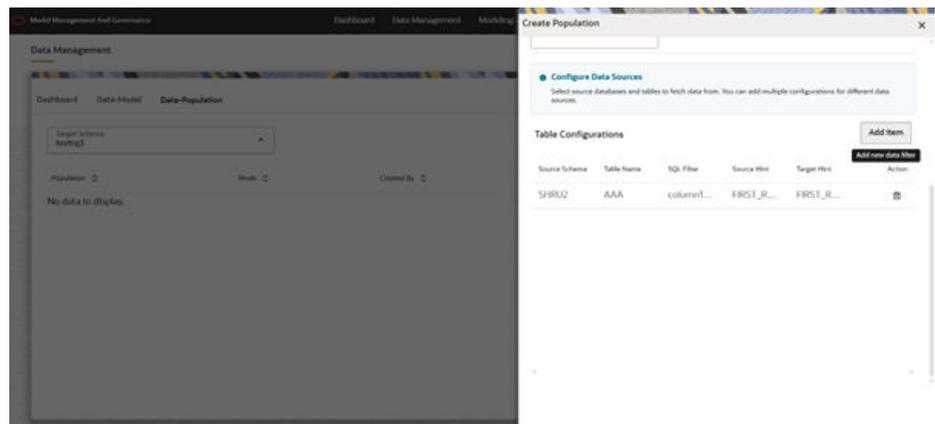
Figure 10-18 Table selection and global filter



- **Table Filters**
 - Table filters are applied to the tables when fetching the data from Source table. We have Global Filters and Table level Filters:

- **Global Filter:** This filter will be applied to all the tables that are part of data population. If the individual table have any table level filters then this is not applied to the specific table.
- **Table Level Filters:** This filter can be applied to individual table level. Click on Add item. This will add an empty row in the table where we need to fill the details of Source schema, table name, SQL Filter, Source Hint, Target hint.
 - a. **Source Schema:** This is dropdown where it lists the Source Schemas, which are already mapped to this Target Schema.
 - b. **Tables:** This is dropdown, where we select a table from list of tables, this list of tables is the tables list that are selected when doing data modeling or workspace creation step.
 - c. We can select more than one table.
 - d. **SQL Filter:** This is the filter that will be applied to the tables that are selected and used while fetching the data.
 - e. **Source Hint:** You can provide database Hints at table-level, Source hint is applied in the source schema while fetching the Data from Source.
 - f. **Target Hint:** You can provide database hints at table-level, Target hint is applied in the Target Schema while doing the inserting Data to Target.

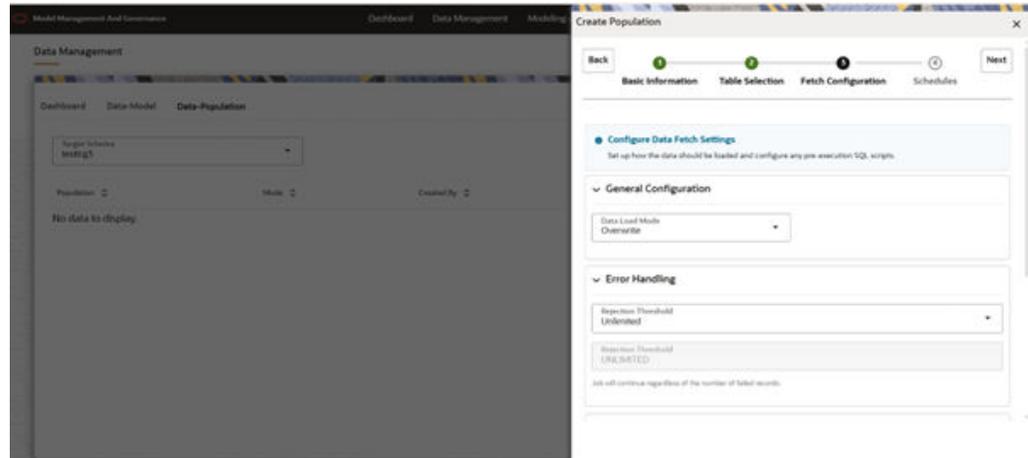
Figure 10-19 Table Level Filters



8. Fetch Configurations

- This has mainly three configurations:

Figure 10-20 General Configuraitons and Error Handling



a. General Configurations

- It has data load mode dropdown, where we have two options: Overwrite and Append.
- Overwrite mode: In this mode, the underlying tables are truncated (overwritten on existing data) followed by an insert operation.
- Append mode: In this mode, the underlying tables will be populated (added to the existing data) in the append mode.

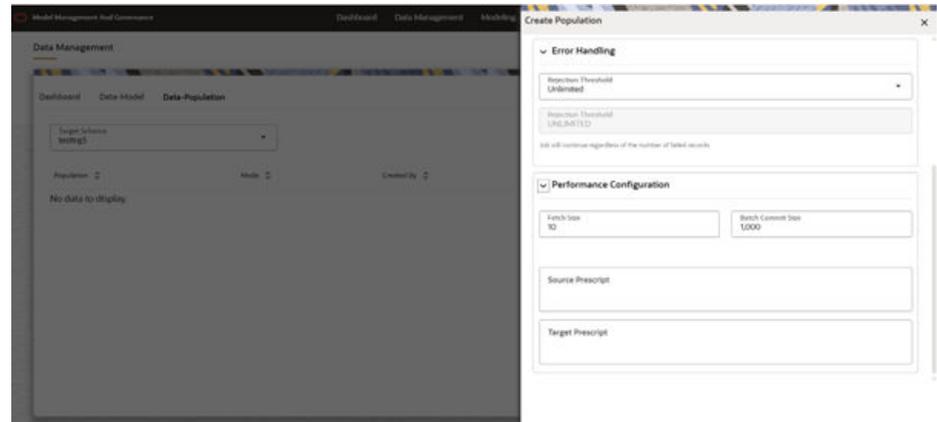
b. Error Handling

- It has Rejection Threshold configuration, whereby default it is set to Unlimited. So Here, all the errors will be ignored during the data population.
- We can also set a custom rejection threshold, for this we need to change the dropdown value to Custom Rejection Threshold, which in turns enables the input box for giving custom value. You can provide the maximum number of inserts that can fail while loading data to a given table. In case of threshold breach, all the inserts into the table will be rolled back. However, it will continue populating to the next table.

c. Performance Configuration

- This consists of Fetch Size, Batch Commit Size, Source prescript, Target prescript.
- **Fetch Size:** Enter the Fetch size of JDBC properties for data upload. By default it is set to 10. Therefore, every time it fetches 10 records from the available records.
- **Batch Commit Size:** Enter the Batch Commit size of JDBC properties for data upload. By default it is set to 1000.
- **Source Prescript:** Enter the source prescript of JDBC properties for data upload.
- **Target Prescript:** Enter the target prescript of JDBC properties for data upload.

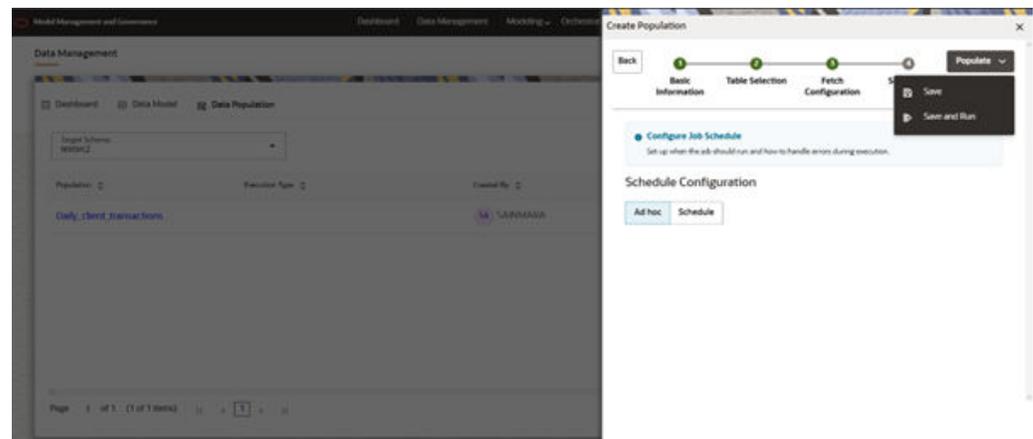
Figure 10-21 Performance Configuration



9. Schedules

- a. We have two options Ad hoc and Schedule, when Ad hoc is selected Populate button at the right acts as drop down and gives two options **SAVE** and **SAVEANDRUN**.
- b. **Save Mode:** In this mode, after filling all the details we will save this configuration. This population can be run manually from the data population lists screen.
- c. **Save And Run Mode:** In this mode, after filling all the details we will save this configuration and run the population once. This population can be run manually from the data population lists screen.

Figure 10-22 Save and Run option



- d. If the selected mode is SCHEDULE, then we need to choose whether we need to bring data Daily/Weekly/Monthly/CRON.
 - i. Based on mode we need to fill in few details.
 - ii. **DAILY:** We need to configure Start Date, End Date and Time when schedule should run daily.
 - iii. **WEEKLY:** We need to configure Start Date, End Date, Time and Days of week when schedule should run.
 - iv. **MONTHLY:** We need to configure Start Date, End Date, Time and Months of year and day of the month when schedule should run.

- v. **ONCE**: We need to configure Start Date and time when schedule should run.
- vi. **CRON**: We need to give a CRON Expression.
- e. Once after giving the information then we need to click on populate.

Figure 10-23 Schedule Type options

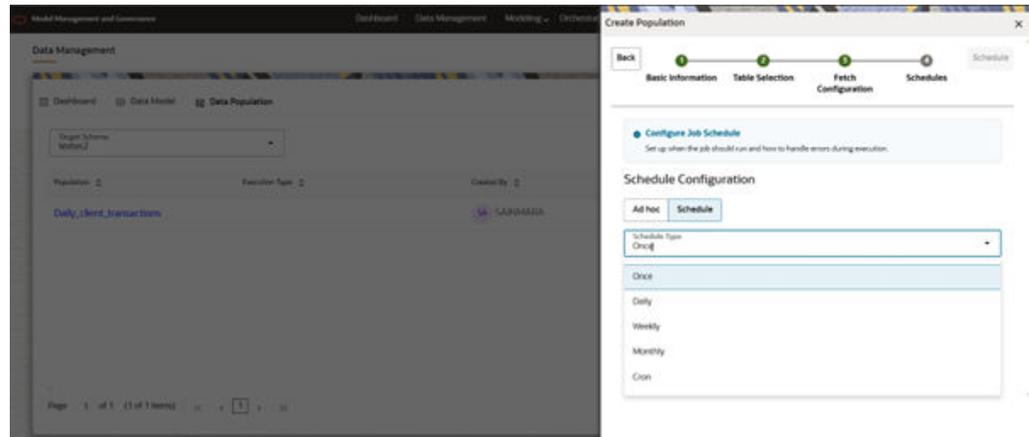


Figure 10-24 Schedule Daily options

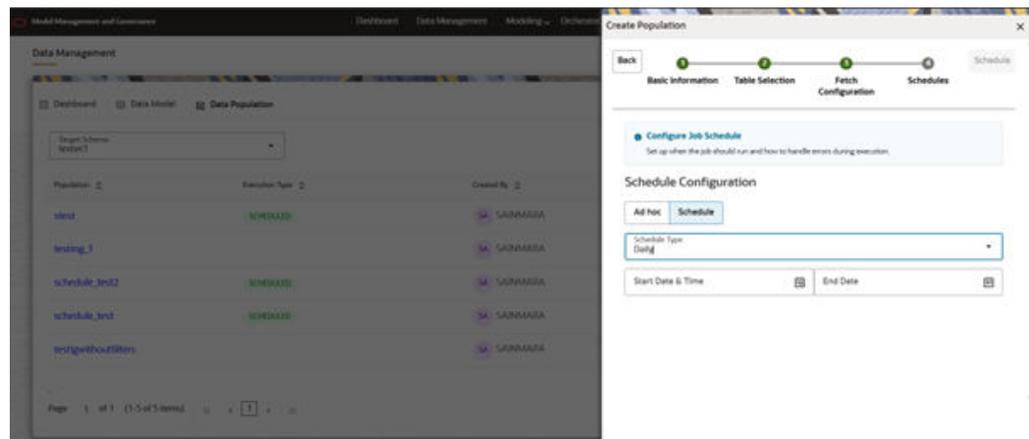


Figure 10-25 Schedule Weekly options

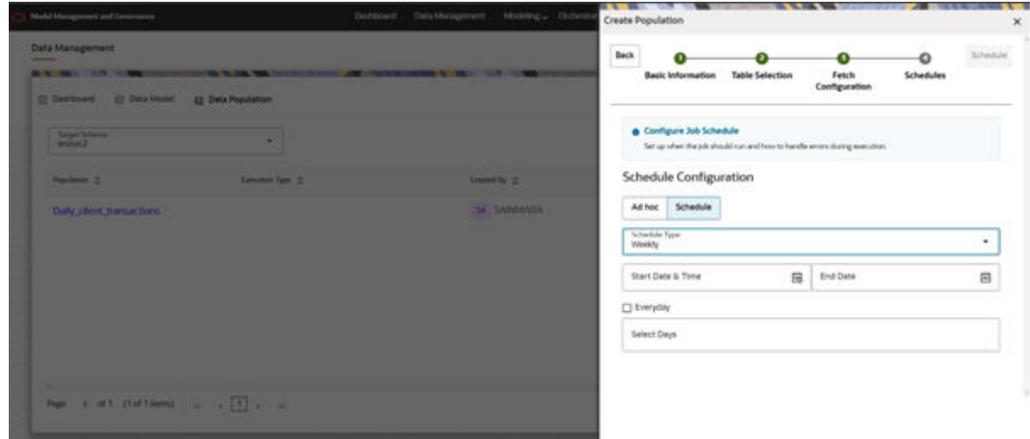


Figure 10-26 Schedule Monthly options

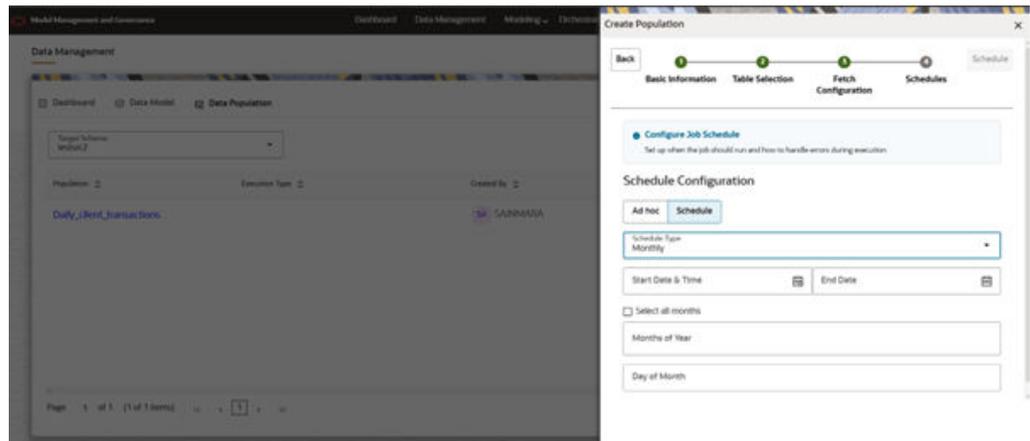


Figure 10-27 Schedule Cron Expression

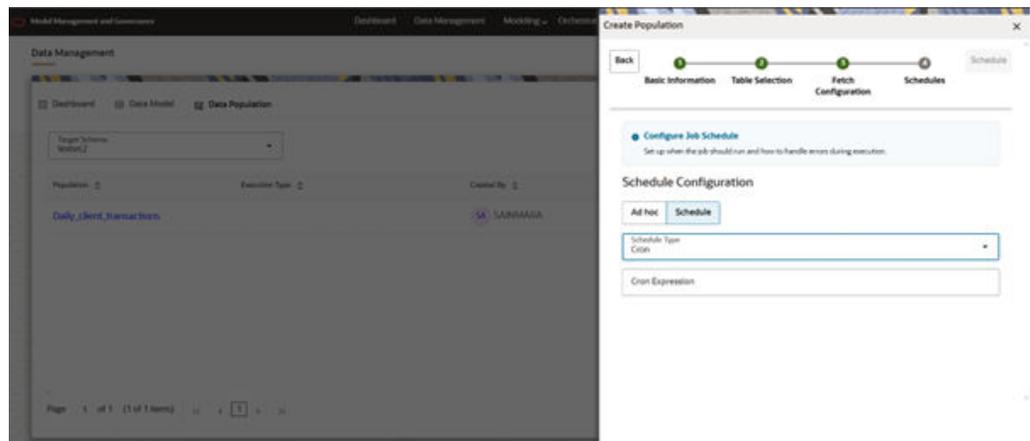
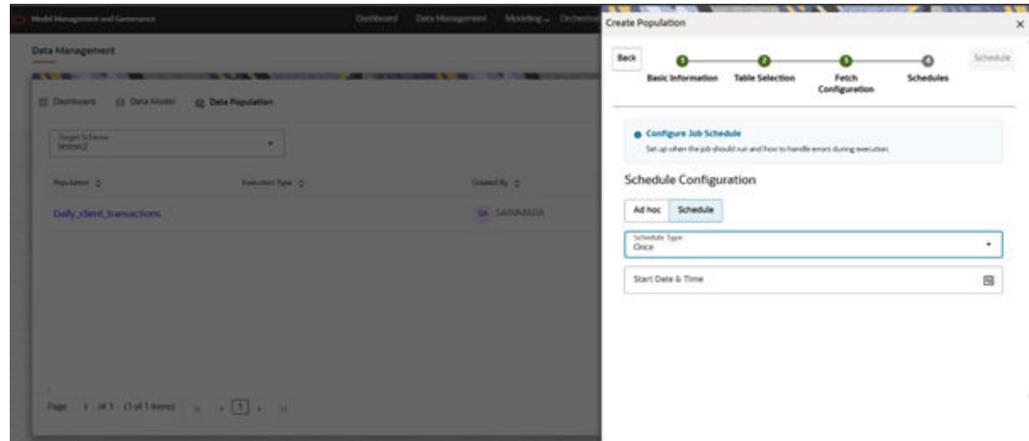


Figure 10-28 Schedule Once options



Data Populations List Action buttons

There are Actions associated to every Data Population based on mode (**SCHEDULE**, **SAVE** OR **SAVEANDRUN**).

Common Actions

1. **DELETE**: When user click on this, it deletes the Data Population.
2. **EDIT**: This option will open a drawer and enables users to edit the data population information which consists of Table level filters, Global filters, Table selection mode, General configuration, Error Handling, Performance Configuration.
3. **VIEW**: When the user clicks on this, it opens a new tab and show the information related to the population.

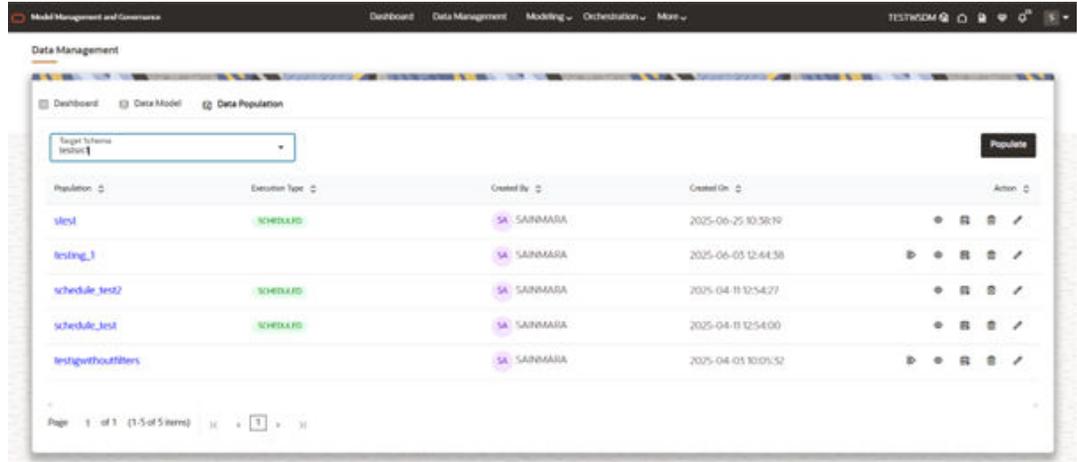
SAVE or SAVEANDRUN Mode Actions

1. **RUN**: We can Run the data population which is already configured. When we click on this the data population is triggered.
2. **SCHEDULE**: We can schedule a data population which is created in this mode. We convert the manually triggered data population to Scheduled one.

SCHEDULE Mode Actions

1. **RESCHEDULE**: This option enables the user to re-schedule already existing scheduled data population.

Figure 10-29 Data Population List Actions



11

Modeling

Modeling covers the following:

- **Datasets:** Explore data, engineer features, and monitor data drifts.
- **Model Libraries:** Manage python libraries to create and use models.
- **Model Techniques:** Manage techniques and algorithms from registered model libraries.
- **Model Catalog:** A centralized hub for all your machine learning models.
- **Pipelines:** Design, deploy and govern end-to-end machine learning pipelines for seamless model development.
- **Graphs:** Analyze complex relationships within your data through interactive graph representations.

Dataset

You can define a metadata about how you want to create a dataset and also take a snapshot of real time data and store it, which you can later use in the pipeline.

This is similar to T2Ts, where you can select data from Data Source, such as file, table or another dataset and so on. The data can be imported from one column of one table, and another column from another table, or any file. After extracting the data from tables or files, provide the name to Target Dataset.

The Dataset is a trail-based UI that allows you to configure the Dataset details.

Access the Dataset Summary

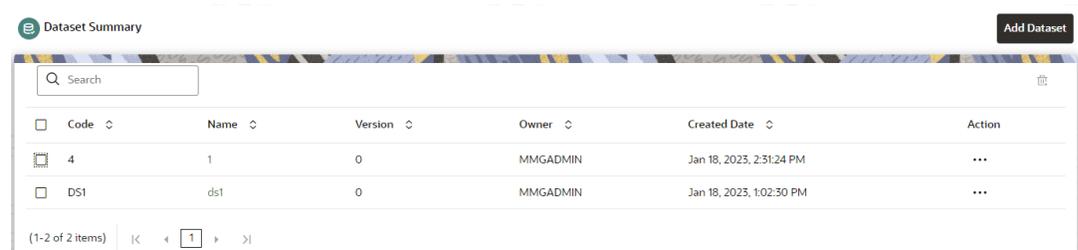
The Dataset Summary page gives access to the various Dataset functions such as create, view, and delete.

To access the Dataset Summary page, follow these steps:

1. Click **Launch Workspace** next to corresponding Workspace to Launch Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. In the LHS menu, click **Dataset** to display the **Dataset Summary** window.

This window displays the dataset records in a table.

Figure 11-1 Dataset Summary page



The screenshot shows the 'Dataset Summary' page with a search bar and an 'Add Dataset' button. Below is a table with the following data:

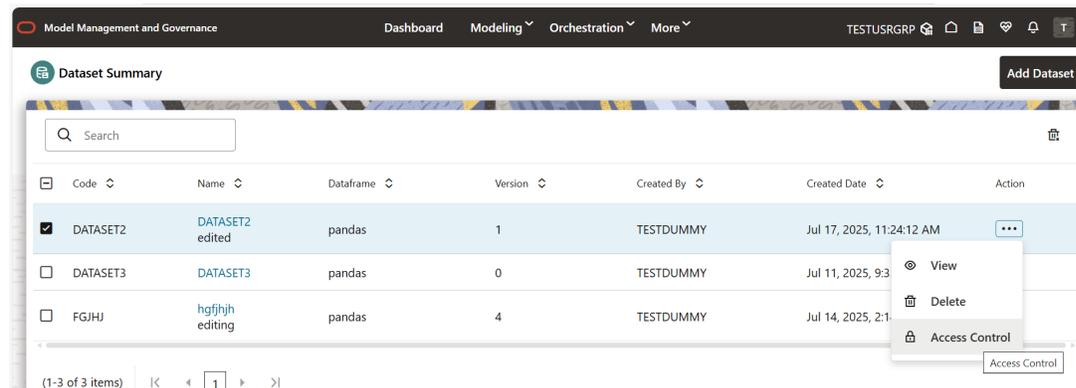
Code	Name	Version	Owner	Created Date	Action
4	1	0	MMGADMIN	Jan 18, 2025, 2:31:24 PM	...
DS1	ds1	0	MMGADMIN	Jan 18, 2025, 1:02:30 PM	...

The following table provides the descriptions for the fields and icons on the Dataset Summary page.

Table 11-1 Fields and icons on the Dataset Summary page

Field or Icon	Description
Search	The field to search for Dataset. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.
Code	The code of the dataset.
Name	The name of the dataset.
Description	The description for the Workspace
Version	Version of dataset
Created Date	The date on which the Dataset was created.
Owner	The owner of the dataset.
Add Dataset	Click Add Dataset to create a new Dataset.
Delete	Click Delete to delete multiple Datasets.
Action	Click the three dots to perform View/Delete/ Access Control functions on selected dataset.

Figure 11-2 Access Control Menu option under Action



Access Control Pop has two Access Level Radio button: Restricted Mode and Public Mode.

Figure 11-3 Restricted Mode

Access Control ✕

Dataset Name Created By ? Select Access level
 DATASET2 TESTDUMMY Public Restricted

Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public.

User Access

User Group Access

Figure 11-4 Public Mode

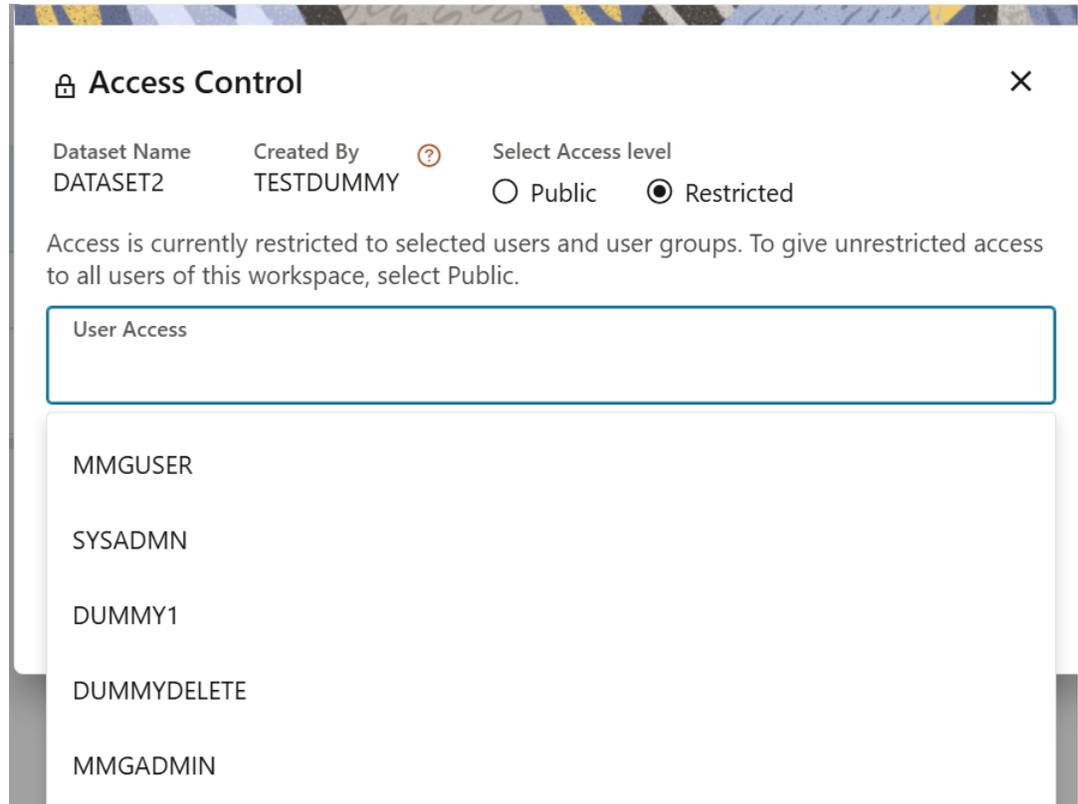
Access Control ✕

Dataset Name Created By ? Select Access level
 DATASET2 TESTDUMMY Public Restricted

Access is currently provided to all users of the current workspace. To restrict user access, select Restricted.

The Access Control pop-up have the below details:

- Dataset Name
- Created By
- Info
- Select Access level with two radio button options: Public and Restricted
- User Access and User Group Access should list all the users present in the Identity Management
- Cancel button
- Save button

Figure 11-5 Access Control User Dropdown

The screenshot shows a dialog box titled "Access Control" with a close button (X) in the top right corner. The dialog contains the following information:

Dataset Name	Created By	?	Select Access level
DATASET2	TESTDUMMY		<input type="radio"/> Public <input checked="" type="radio"/> Restricted

Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public.

User Access

- MMGUSER
- SYSADMN
- DUMMY1
- DUMMYDELETE
- MMGADMIN

The Access Control User Groups dropdown fetches all the User Groups present in the Identity Management.

Figure 11-6 Access Control User Groups dropdown

AUTOGRP

DMTEST

DSREDACTGRP

DUMMY1

DCUSRGRP

DMADMIN

DSUSRGRP

DOCADMIN

FILEADMINUSER

User Group Access

Cancel Save

Create a Dataset

The Dataset creation requires entry of the source of dataset and validation. These datasets are required for schema creation.

Creating a Dataset

To create a Dataset:

- On the Dataset Summary page, click **Add Dataset**.

Figure 11-7 Dataset Creation window

1 Define Pipeline Characteristics

2 Source Selection

3 Dataset Creation

4 Dataset Transformation

Code Required

Dataset Name Required

Description

Dataframe
 Pandas Modin Spark

Python Runtime
 default

Provide the following information to create a Dataset:

- Define Pipeline Characteristics
- Source Selection
- Dataset Creation
- Dataset Transformation

Define Pipeline Characteristics

Enter basic configuration details in this window.

To configure the basic details for the dataset:

1. Enter the required details in the **Define Pipeline Characteristics** window as shown in the following table.

Table 11-2 Details for Basic Details pane

Field	Description
Code	Enter the identification code of the dataset. This field is limited to 30 alphanumeric characters.
Dataset Name	Enter the name of dataset. This field is limited to 30 alphanumeric characters. Space not exceeding 30 characters. You cannot keep this field blank.
Description	Enter the purpose of the creation of the dataset. This field is limited to 150 alphanumeric characters. Space not exceeding 150 characters.

2. Select the data library from the options: **Pandas**, **Modin**, or **Spark** and select **Python Runtime** from the drop-down and click **Close**.
 - **Pandas**: An open-source data manipulation library for Python. It provides data structures such as Series (1-dimensional) and DataFrame (2-dimensional) that allow for easy manipulation and analysis of data. It also provides tools for reading and writing data to various file formats, including CSV, Excel, and SQL databases. **Pandas** is the default selection.
 - **Modin**: An open-source library that allows for faster operations on DataFrames using distributed computing which can lead to significant speed improvements, particularly for large datasets or computationally expensive operations.
 - **Spark** : Pyspark option for scaling dataset.
If Spark library is selected, the Python Runtime drop-down option is not displayed.
3. Click **Next** to go to the next step.

Source Selection

This section allows you to define the source of data. Here, you can choose the data structures from an existing datasources. Click on the icon next to Data Source field to view the tables/ data source information associated with the data source.

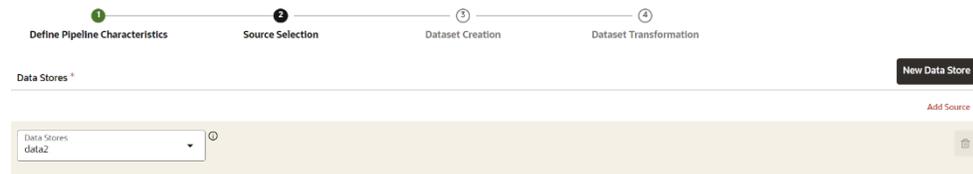
To configure the Source details for the dataset, follow these steps:

1. Select the required **Data Store** from the drop-down.

OR

Click **New Data Store** to create a data source.

Figure 11-8 Source Selection Details screen



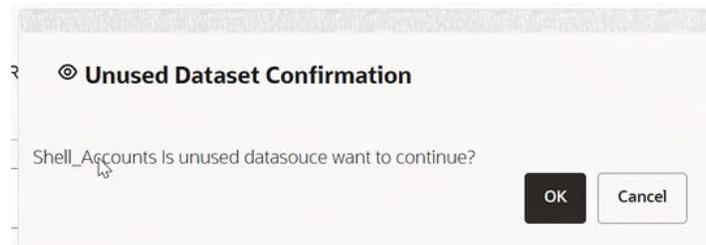
Note

The MISDATE feature is displayed only when the Data Source field is selected as a **File Type**.
If enabled, you need to add the Physical and Feature name for the Data Source.

Table 11-3 Details for Basic Details pane

Field	Description
Data Stores	The Data Stores drop-down list shows all the data sources and the data dorns those are tagged with the corresponding workspace. In addition, this lists all the available unused data source, which can be used. You can search the data sources by clicking on Data Store information icon. A confirmation message is displayed to tag an unused datasouce with workspace.

Figure 11-9 Unused Dataset Confirmation message



Here, either use the unused or create a new datasource using **New Datasource**. When you create a new datasource, it becomes as unused.

You can add more Data Sources using **Add More** link.

Use **Delete** to delete added Data Source.

2. Click **Next** to go to the next step or click **Previous** to go back to previous step.

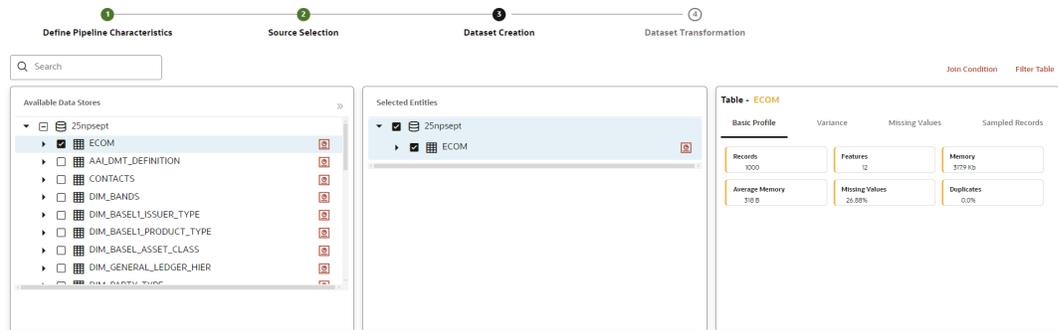
Create a Dataset

This window allows you to select entity part (for example, column of table) of the datasource. The data source selected in the previous step requires the definition of database objects to be used.

To configure the Dataset Creation:

1. Select the required data source from the **Available Data Sources** pane and click >> to move the selected data to **Selected Entities** pane. The **Available Data Sources** section shows the high-level data sources. You can select data from multiple data source entities. The selected Data Sources are displayed in **Selected Entities** pane. You can use to select or use to de-select the data in the **Selected Entities** pane. Additionally, clicking on **View Profile** icon displays the profile details in the **Basic Profile** tab.

Figure 11-10 Dataset creation page



2. For more information on Joins, see [Joining Tables](#) section.
For more information on Filtering, see [Filtering Tables](#) section.
3. Click **Next** to go to the next step or click **Previous** to go back to previous step.

Joining Tables

Use Join Condition to combine the data source details. Each data sources have multiple tables and multiple columns, so you can join them using Join Condition window. Joins can be applied only when user has selected more than one table.

Figure 11-11 Join tables



You can join Tables either using drop-down or using the Joinscript option.

Using Drop-down

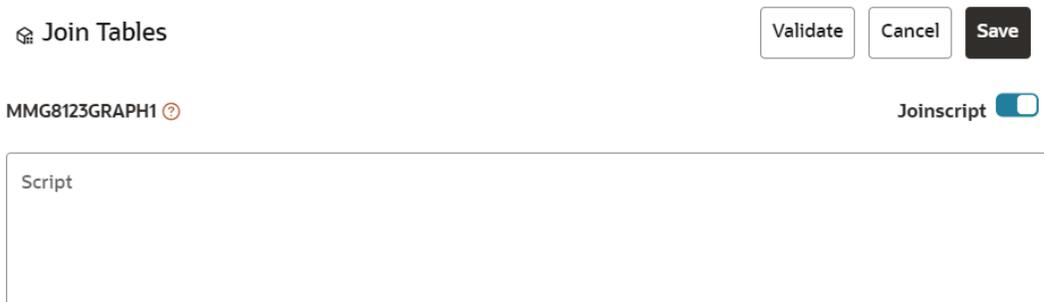
Select the Column name of Table from the drop-down list. You can use multiple join conditions by using icon.

Note

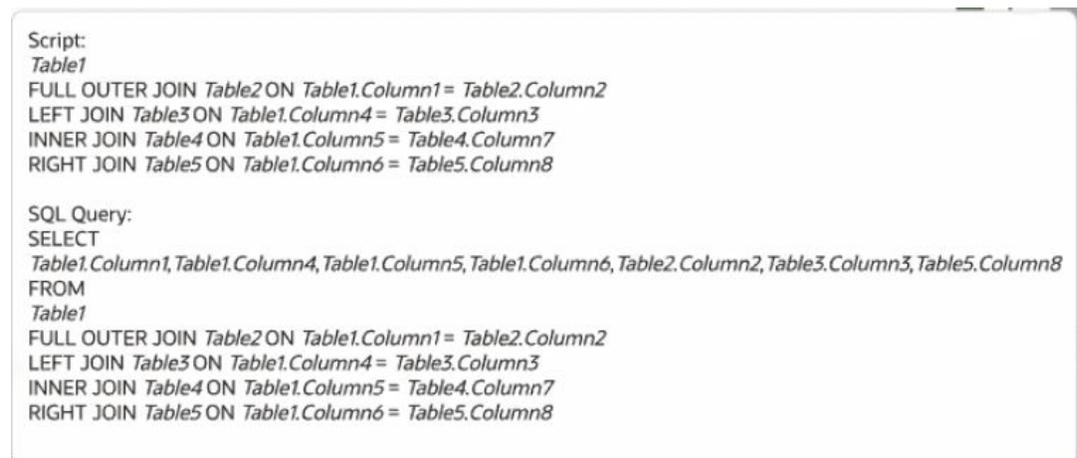
In the current release, only INNER JOIN is present.

Using Script

1. To use script, enable **Join Script** option.
The Join Tables window is displayed to add scripts.

Figure 11-12 Join Tables window

You must provide the script in below format:

Figure 11-13 Script

2. Click Validate to check the Join conditions.
3. Click Save.

Filtering Tables

You can also use the filters in dataframe creation from entity.

1. Click **Filter Table** to navigate to Filter tables window.

Figure 11-14 Filter Table

There are certain rules to be followed when adding filters.

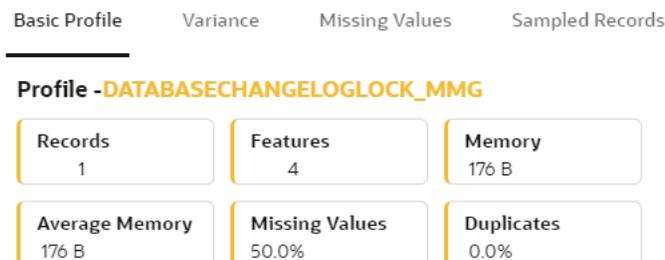
Figure 11-15 Rules to be followed when adding filters

- I. Filtervalues will not be applied in tables from datasources joined using joinscript. 
- II. Only one filtervalue entry is allowed for one table.If multiple entries are present, the last filtervalue entry will be used for that table.
- III. Example for non-file datasource: SQL Format
Filter : `Column1=1234`
SQL script: `SELECT * FROM tablename WHERE Column1=1234`
- IV. For file source, script follows python syntax
- V. Always assign the output dataframe as 'df_out='
- VI. Use 'df_prev' to access the original dataframe
- VII. Example for file source: Python
Filter : `df_out=df_prev.loc[df_prev['Column1'] == 1]`

2. Multiple filters can be added by using **Add**.
3. Click **Validate** to check the Filter conditions.
4. Click **Save**.

Viewing Profile

The Profile section from RHS shows the report based on Selected Objects (complete dataset) section. For example, Missing Values report shows the details of missing data in selected table in Selected Objects section.

Figure 11-16 Selected Objects section

Dataset Transformation

This window allows you to transform the data source information. This complete grid displays the Table like structure, which helps you to make the data better using many methods, such as by remove all the missing values, performing scaling and so on.

This window shows the following columns:

- **Physical Name:** Shows the Physical Name of column.
- **Feature Name:** Shows the Feature Name of column along with its data type. You can edit the Feature Name.
- **Observations:** The Observations field shows the value of column. For example, if any column is missing any value, you can easily identify missing value columns. Use Transformation button to fix these values.
- **Create Feature:** To create a new feature/column by doing operations on the existing features.
- **Calculate Fairness:** You can calculate the fairness metrics of the dataset by choosing the required metric, target variable, and protected features. For more details, see [Fairness](#) section. Click Help icon for more details on how the fairness is calculated.
- **View Report:** To view the complete report of the sampled data.
- **Re-Validate:** To re validate all the transformations.
You can also revalidate an individual transformation by clicking the More Action and select the Re-Validate option
- **Add Transformation:** To add new transformations

The profile button helps you to view profile the final data frame that you have selected. For more information, see the Viewing Profile section.

Here, you can perform the following actions:

- Re-ordering of transformations
- Insertion of transformation

Create New Feature

The Create New Feature window allows you to create a new feature. This is used to add a new column to dataset. This is useful, if you want to add a new column based on derived T2Ts (from data source).

To create a New Feature, follow these steps:

1. Click Create Feature on Dataset Transformation of Dataset UI.

Figure 11-17 Create Feature on Dataset Transformation of Dataset UI

Create New Feature Validate Cancel Add

Physical Name

Data Source: CUSTOM_DATASOURCE Entity: CUSTOM_ENTITY Physical Feature Name: Required

Feature Name

Feature Name: Required

Script ?

`df_prev['featureName1']+df_prev['featureName2']` Required

2. Enter the following details:
 - **Physical Feature:** Name of Physical feature
 - **Feature Name:** Logical Name of feature
 - **Script:** Update the script
3. Click **Add** after validating the feature.
The new feature will be added at end of LHS section.
4. You can also edit or delete a newly added feature.

Add Data Transformation

To add the data transformation, follow these steps:

1. Click **Add**.
The Transform window is displayed.
2. Select the Transformation Type.
Following types are available:
 - **Dataframe Transform:** This is used to transform entire dataframe. For example, if you want to remove all missing values from all the columns of entire sampled data.
 - **Feature Transform:** This is used to transform a particular column of dataframe.

Figure 11-18 Transformation screen

3. Select **Transform**: Here you either enter values using Method and Argument fields, or script.
4. Click **Validate** to validate the details.
5. Click **Add** to add the new transformation.
The New Transform is created and displayed at RHS section.
6. Click **Finish** to navigate to Dataset Summary window.
This saves the metadata of dataset.

Example:

If there is “missing value” for one of the columns, then perform following steps to add the transform.

- a. Click Add on Dataset Transformation window.
- b. Select Type as Feature Transform.
- c. Select Transform as Impute Missing Value.
- d. Select the Physical Name from Physical Name drop-down list.

- e. Select the Method and enter parameter in Argument field. The Method is updated on selected Column type. For example, if the selected column type is numerical, then following methods will be available: Simple, Constant, KNN, and Mice
OR

Enter the script

Below are the Sample Custom Scripts:

- Directly pass input data frame to output: `df_out=df_prev`
- Drop first row of data frame: `df_out=df_prev.drop(0,axis=0)` if not `df_prev.empty` else `df_prev`
- Drop column from data frame: `df_out = df_prev.drop('colname',axis=1)`

Sample Custom Scripts

Below are few sample scripts which the users can refer to create transformations.

Table 11-4 Sample Custom Scripts

Function	Script	Comments
Directly pass input data frame to output	<code>df_out=df_prev</code>	
Drop first row of data frame	<code>df_out=df_prev.drop(0,axis=0)</code> if not <code>df_prev.empty</code> else <code>df_prev</code>	
Drop column from data frame	<code>df_out=</code> <code>df_prev.drop('colname',axis=1)</code>	FEATURE TRANSFORMATION

Transformations

These are the various transformations which can be done from the UI.

Table 11-5 Transformations

No.	Transformation	Function
1.	Add New Feature	A new feature can be added to the dataset which could be derived from the existing features using Script. Physical Feature Name and Feature Name are the names of the new feature. Script can be used to create a pandas Series for the new feature
2.	Encode Categorical Features	This function performs One Hot Encoding on a categorical feature and replaces it with multiple numerical features in the dataset.
3.	Encode Datetime Features	This function encodes a datetime feature and replaces it with multiple numerical features having the following information derived from the datetime feature - year, month, week, day, hour, minute, dayofweek.

Table 11-5 (Cont.) Transformations

No.	Transformation	Function
4.	Encode Cyclical Features	<p>This function encodes a cyclical feature having hour, minute data, and so on and returns two features carrying the sine and cosine transformation of the cyclical data.</p> <p>'fmax ' denotes the maximum possible value of the cyclical feature data.</p>
5.	Impute Missing Data	<p>This function imputes missing data within a feature.</p> <p>For numerical features, there are 4 methods for imputing missing data.</p> <p>simple - imputes with mean, median, most_frequent values based on chosen arg value using the SimpleImputer in sklearn.</p> <p>const - fills the missing values with the value given in the arg</p> <p>knn - imputes using the KNNImputer in sklearn with k value given in arg</p> <p>mice - imputes using the IterativeImputer in sklearn</p> <p>For non-numerical data, missing values can be imputed using the 'const' method by replacing all missing values with the value given in arg</p>
6.	Feature Scaling	<p>This function is used to scale multiple selected numerical features using the StandardScaler in sklearn</p>
7.	Dimensionality Reduction	<p>This function performs PCA on selected numerical features to reduce the dimensionality using sklearn.decomposition.PCA module.</p> <p>The number of output features can be specified using dim field.</p> <p>The names of the output features' names can be specified in the fields 'Physical Feature Name' and 'Feature Name'</p>
8.	Outlier removal	<p>This function is used to remove outliers present in a feature based on the specified zscore value.</p> <p>Non-numerical features are label encoded before removing the outliers.</p>
9.	Duplicates Removal - Data Frame	<p>This function removes all duplicate rows in the dataframe.</p>
10.	Duplicates Removal - Feature	<p>This function removes all duplicate rows within a specified subset of features and consequently removes those rows from the data frame</p>

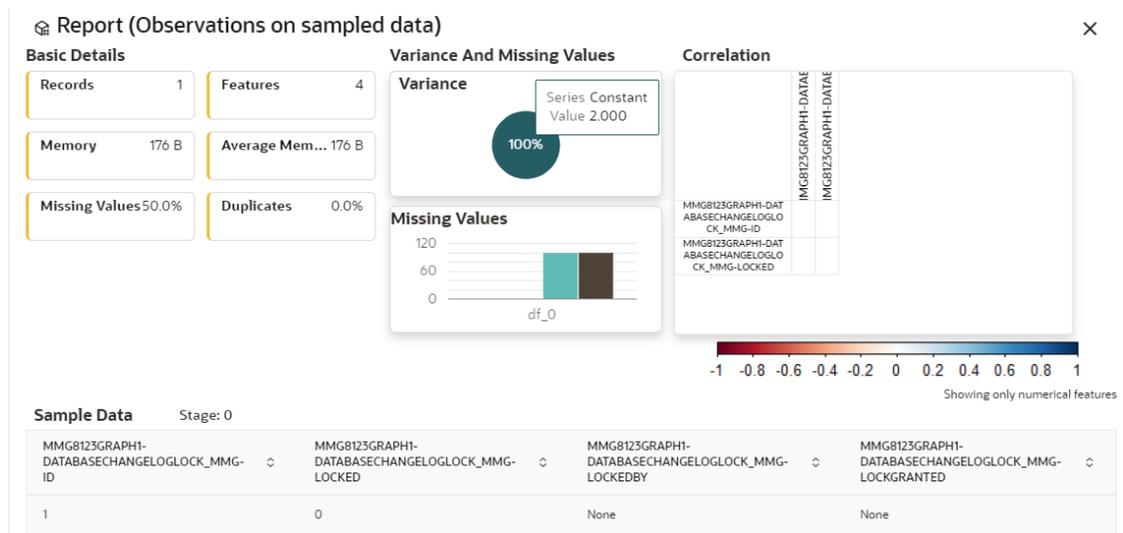
Table 11-5 (Cont.) Transformations

No.	Transformation	Function
11.	Filter Features	<p>This function is used to filter the data frame based on conditions specified on features</p> <p>Operations allowed : >, >=, =, !=, <, <=, isin</p> <p>When the chosen operation is 'isin', the input to 'Filter Value' is a list of values that should be present in the output data frame</p>

View Reports

This report shows the Observation on Sampled data, profiles, Variance, Correlation Matrix (correlation between columns), Sample Data. This report is based on sampled data.

Figure 11-19 Sampled Data Report



Re-ordering of Transformations

You can re-order the transformations using the drag-drop. During the transformation re-order, you can compare the profile of transformations. The transformation order gets adjusted accordingly.

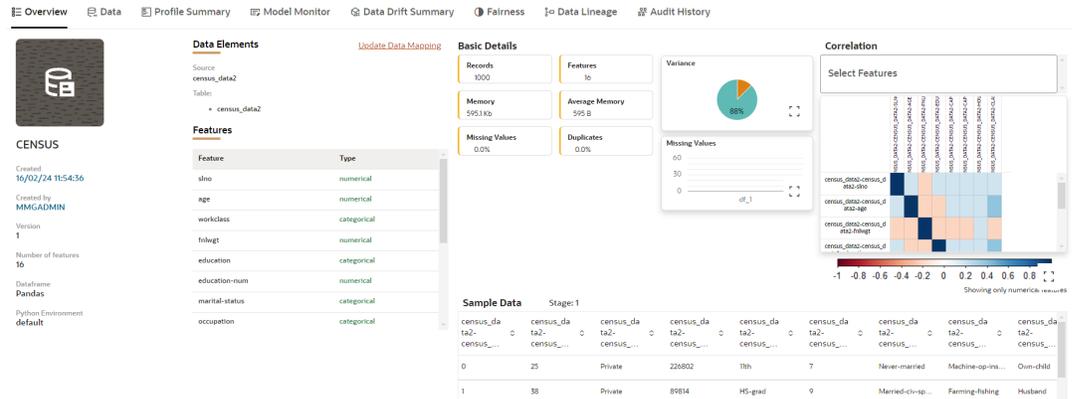
View a Dataset

To view a Dataset, follow these steps:

1. Navigate to Dataset Summary page.
2. Click **Action** next to corresponding Dataset and select **View**.

The Profile Summary screen is displayed.

Figure 11-20 Dataset Details screen



Overview

The details such as Data Elements, Features, Basic Details, Correlation, and Sample Data are displayed in graphical format.

To view the Overview of a Dataset, follow these steps:

1. Navigate to **Dataset Summary** page.
2. Click the **Next** to corresponding Dataset and select **View**.

OR

Click on the Dataset **Name** column which you want to view.

The Overview page is displayed.

You can update the data mapping details by clicking on Edit icon in Data Elements group.

Figure 11-21 Update Data Mapping

Update Data Mapping [X]

▼ **Source 1**

Existing Data Source
filedata

New Data Source
filedata ▼

Existing Table Name 1
filedata

New Table Name
filedata ▼

Features

Existing Column Name 1
cm3

New Column Name ▼

Existing Column Name 2
cm4

New Column Name ▼

Existing Column Name 3
cm

New Column Name ▼

Existing Column Name 4
cm2

New Column Name ▼

Existing Column Name 5
t1

New Column Name ▼

Data

To view the data of a Dataset, follow these steps:

1. Navigate to the **Dataset Summary** page.
2. Click **Next** to the corresponding Dataset and select **View**.

The Overview page is displayed.

3. Navigate to **Data** tab.

Profile Summary

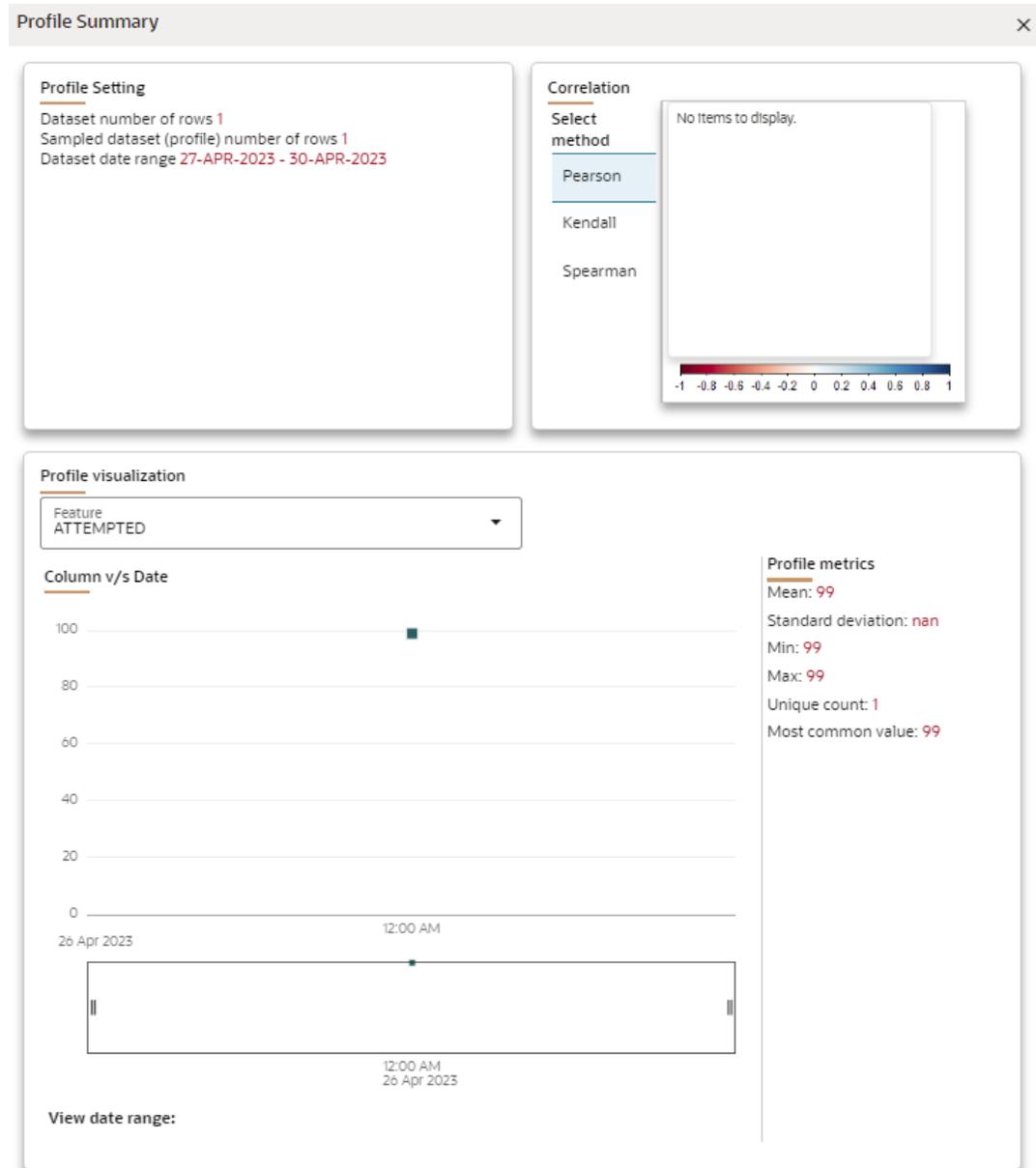
To view the Profile Summary, perform the following steps:

1. In the Profile Summary screen, click next to corresponding data profile and select View Summary.

This window displays the following details in graphical format:

- Profile Setting
- Correlation
- Profile visualization along with the metrics

Figure 11-22 Profile Summary



Start Profiling

Prerequisite: You must create a Dataset with a datasource having columns with **DATA_TYPE** as **DATE**.

To start a Profile:

1. Navigate to Dataset Summary page.
2. Click **Action** next to corresponding Dataset and select **View**.

This window displays the Dataset profiles in a table.

Figure 11-23 Dataset Profiles

Data Profile Id	Creation Datetime	Dataset Range	Action
<input checked="" type="checkbox"/> 1_DATAPROFILER	26-APR-2023 07:42:22 AM	START - 26-APR-2023	...
<input type="checkbox"/> 2_DATAPROFILER	26-APR-2023 07:48:29 AM	26-APR-2023 - 26-APR-2023	...
<input type="checkbox"/> 3_DATAPROFILER	26-APR-2023 01:00:04 PM	26-APR-2023 - 26-APR-2023	...
<input type="checkbox"/> 4_DATAPROFILER	27-APR-2023 03:27:42 AM	26-APR-2023 - 27-APR-2023	...

3. Click **Start Profiling** or click **Settings** .
The Profiler Settings page is displayed.

Figure 11-24 Profiler Settings page

Profiler Settings

Basic Information

Columns Required

Threshold ?

+ Add Column

Filter Column ? Required

Generate First Profile

Schedule

4. Under Basic Information section, enter the following details:
 - a. Select the required columns from the drop-down. All the numeric options are displayed under the Columns drop down. You can select multiple columns by clicking on + icon.
 - b. Enter the threshold for the columns. Threshold defines the Drift detection threshold for each column and only numerical columns can be selected for profiling.
 - c. Select the required filter column from the drop-down. Filter column contains date column of the dataset to filter.
 - d. Enable **Generate First Profile** option if you want to generate a profile.
5. Under Schedule section, enter the following details:
 - a. Enter a Schedule Name.
 - b. Select the Schedule Type as required:
 - **Daily**: Select to run the schedule everyday.
 - **Weekly**: Select to run the schedule once in a week or the selected days in a week.
 - **Monthly**: Select to run the schedule once in a month or the selected days in a month.
 - c. Click **Save**.

The Data Profiling has been started and Stop Profiling option is displayed in the Profile Summary screen.

For more details on disabling dataset profile, see [Stop Profiling](#) section.

6. Click **Generate Profile**.

The Data Profile is created.

Schedule Daily

To schedule the profiler to run daily, perform the following steps:

- a. In the **Profiler Settings** screen, select the Schedule Type as **Daily**.
- b. Select the start date on which you want to run the profiler.
- c. Select the end date on which you want to stop your schedule.
- d. Enter the time at which you want to run the profiler.
- e. Click **Save**.

Schedule Weekly

To schedule the profiler to run weekly, perform the following steps:

- a. In the **Profiler Settings** screen, select the Schedule Type as **Weekly**.
- b. Select the start date on which you want to run the profiler.
- c. Select the end date on which you want to stop your schedule.
- d. Enter the time at which you want to run the profiler.
- e. Select the day on which you want to run the profiler. You can select multiple days to run the profiler.
- f. Click **Save**.

Schedule Monthly

To schedule the profiler to run monthly, perform the following steps:

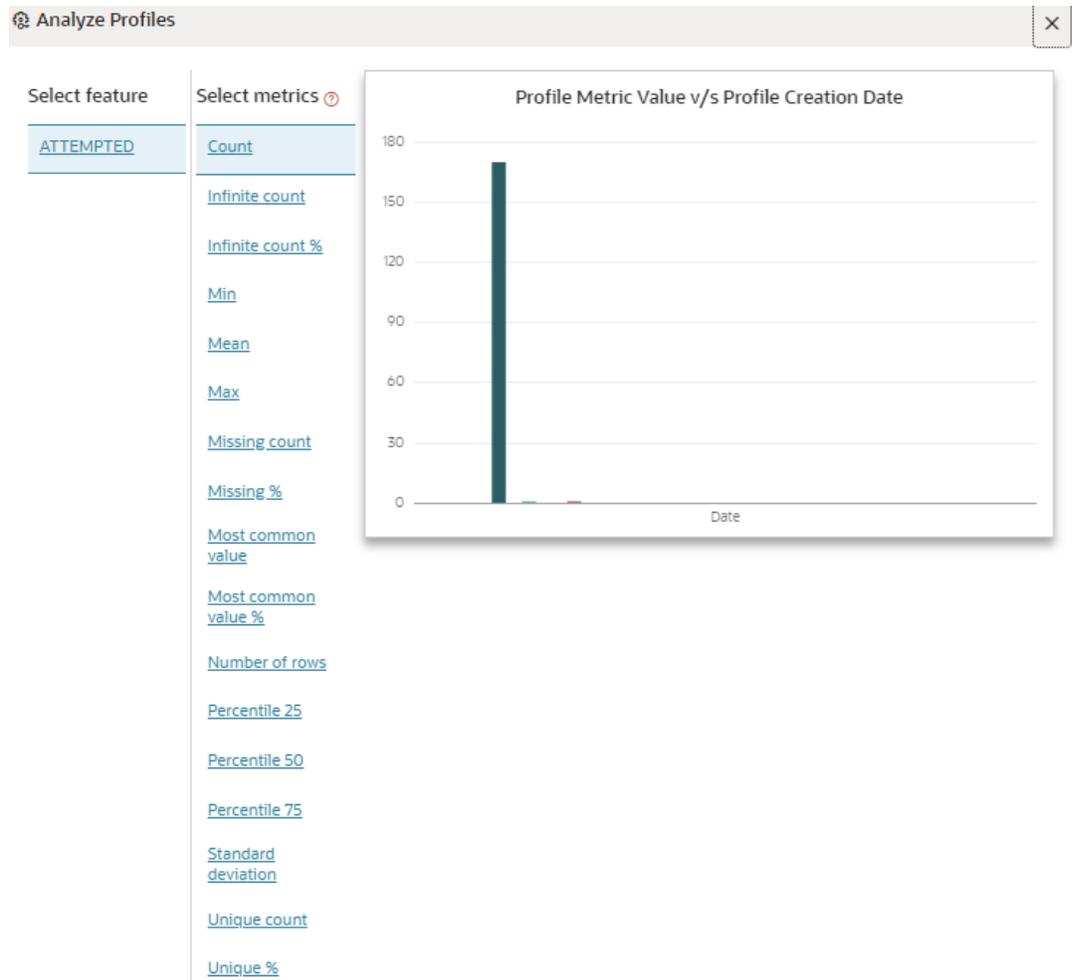
- a. In the **Profiler Settings** screen, select the Schedule Type as **Daily**.
- b. Select the start date on which you want to run the profiler.
- c. Select the end date on which you want to stop your schedule.
- d. Enter the time at which you want to run the profiler.
- e. Select the month on which you want to run the profiler. You can select multiple months to run the profiler.
- f. Select the date on which you want to run the profile of the selected month to run.
- g. Click **Save**.

Analyze Profiles

To analyze multiple data profiles, perform the following steps:

1. In the Profiler Summary screen, select the data profiles that you want to analyze profiles. You can select multiple data profiles.
2. Click **Analyze Profiles**.

A window is displayed with multiple metrics of the selected profiles. Click on each metric to view the data in graphical format.

Figure 11-25 Analyze Profiles screen

Compare Profiles

To compare the data between two data profiles, perform the following steps:

1. In the Profiler Summary screen, select two data profiles that you want to compare.
2. Click **Compare Profiles**.
The Profile Comparison is displayed.
3. In the Dataset Feature drop-down, select the feature that you want to compare.
The properties of both the profiles is compared and displayed in a tabular format.

Figure 11-26 Profile Comparison screen

Profile Name	Count	Mean	Standard Deviation	Min	Max	Unique Count	Most Common
1_DATAPROFILER	170	45.47	101.81	2	1200	26	12
2_DATAPROFILER	0	N/A	N/A	N/A	N/A	N/A	N/A

Stop Profiling

To stop the profiling, perform the following steps:

1. In the Profiler Summary screen, select the data profiles that you want to stop profiling. You can select multiple data profiles.
2. Click **Stop Profiling**.
A confirmation pop-up window is displayed.
3. Click **Yes** to disable the dataset profile.

The dataset profiling has been disabled and Start Profiling option is displayed. For more information on profile, see [Create a Profile](#) section.

Model Monitor

Model Monitoring refers to the process of closely tracking the performance of models in production and helps you to frequently monitor the distribution of the model and provides alerts in case of any exceptions.

In addition, it enables you to identify and eliminate bad quality predictions and poor technical performance of the models.

The model's robustness depends not only on the training of the feature-engineered data but also on how well the model is monitored after deployment. Typically a model's performance degrades over time, and it essential to detect the cause of the decrease in performance of the model. The main cause of the decline in performance can be drift in the independent or/and dependent features which may violate the model's assumption and distribution about the data. When models are built, the model builder will create a snapshot of the dataset used for training

the model and save it in a file which is later used for calculating drift with the current snapshot of the dataset.

To start the model monitoring, perform the following steps:

1. In the LHS menu, click **Dataset** to display the **Dataset Summary** window.
This window displays the dataset records in a table.
2. Click **Action** next to corresponding Dataset and select **View**.
The Profile Summary screen is displayed.
3. Navigate to **Mode Monitor** page.
All the models associated with the selected dataset will be displayed.

Figure 11-27 Mode Monitor page

Model Name	Model ID	Version	Is Monitored	User	Creation Date	Action
MM001	168373645792	0	<input type="checkbox"/>	MRAGSAML	2023-05-04 04:14:06	...

4. Click **Start Model Monitoring** or **Monitor Settings** icon.
The Monitor Settings page is displayed.

Figure 11-28 Monitor Settings page

Monitor Settings

Basic Information

Threshold ?

45

Schedule

5. Drag the slider and set the threshold settings. Threshold is the value in percentage, beyond which it is labelled as drifted.
6. Select the Schedule Type as required:
 - Daily: Select to run the schedule everyday.
 - Weekly: Select to run the schedule once in a week or the selected days in a week.
 - Monthly: Select to run the schedule once in a month or the selected days in a month.

The monitoring for the model is started and Is Monitored option is enabled by default and you can disable it if you do not want to monitor.

Stop the Model Monitoring

To stop the model monitoring, perform the following steps:

1. Navigate to **Model Monitor** page.
All the models associated with the selected dataset will be displayed.

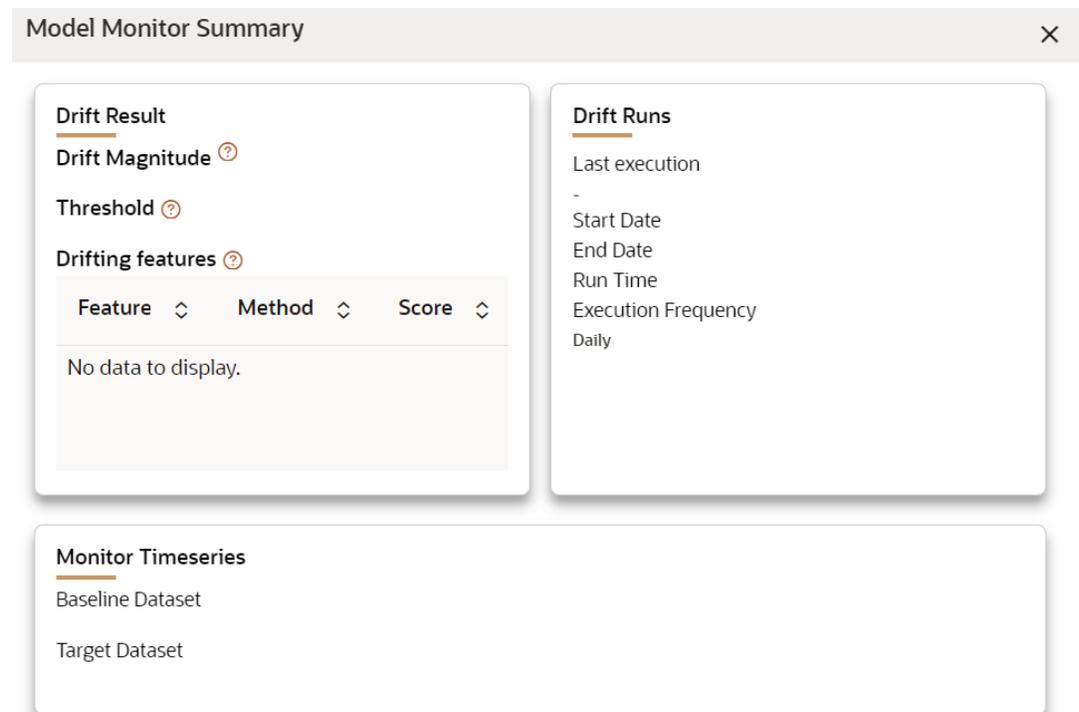
2. Click **Stop Model Monitoring**.
A confirmation pop-up window is displayed.
3. Click **Yes**.
The Model Monitoring is stopped.

Viewing the Model Summary

To view the Model Summary, perform the following steps:

1. Navigate to **Model Monitor** page.
All the models associated with the selected dataset will be displayed.
2. Click **Action** next to corresponding model and select **View Summary**.
This window displays the following details in graphical format.
 - Drift Result
 - Drift Runs
 - Monitor Timeseries

Figure 11-29 Model Monitor Summary



Viewing the Model Latest Report

To view the Model Latest Report, perform the following steps:

1. Navigate to **Model Monitor** page.
All the models associated with the selected dataset will be displayed.
2. Click **Action** next to corresponding model and select **View Latest Report**.
The execution details of the model are displayed.

Figure 11-30 Model Execution History

Report Id	Creation Datetime	Data Drift	Action
1	2023-05-06 00:00:04		...

(1 of 1 items) | < < 1 > >

To view the report of each execution, click **Action** next to corresponding report and select **View Report**.

Viewing the Model Execution History

To view the Model Execution History, perform the following steps:

1. Navigate to **Model Monitor** page.
All the models associated with the selected dataset will be displayed.
2. Click **Action** next to corresponding model and select **View Execution History**.
This window displays the details such as Baseline Distribution, Target Distribution, Drift Method, Drift Score, and Data Drift of the models.

Data Drift Summary

A Data Drift Summary report summarizes the changes in a dataset over time. It typically includes information for any changes in the data distribution of the selected features over the selected baseline and target date ranges.

The report is used to help identify potential issues with data quality which helps the users to take necessary actions further.

To calculate the data drift, perform the following steps:

1. In the Data Drift Summary screen, click Calculate Drift.

The Data Drift Analysis window is displayed.

Figure 11-31 Data Drift Summary screen

Data drift analysis

Select baseline data range: From: 04/12/2022 To: 07/18/2022

Select target data range: From: 10/18/2022 To: 01/18/2023

Select drift Method: Method: Kolmogorov-Smirnov Test [Calculate]

Feature	Action	Type	Baseline Distribution	Target Distribution	Drift Method	Drift Score	Data Drift
No data to display.							

(0 of 0 items) | < < 1 > >

Close

2. Select the data range (From and To) dates for the baseline.
3. Select the data range (From and To) dates for the target.
4. Select the drift method from the drop-down.

The available drift methods are:

- Kolmogorov–Smirnov (K-S) test
 - Only for numerical features
 - Output: p_value, drift detected when p_value < threshold.
 - Kullback-Leibler divergence
 - For numerical and categorical features
 - Output: divergence, drift detected when divergence >= threshold.
 - Wasserstein distance (normed)
 - Only for numerical features
 - Output: distance, drift detected when distance >= threshold.
 - Population Stability Index (PSI)
 - For numerical and categorical features
 - Output: psi_value, drift detected when psi_value >= threshold.
 - Jensen-Shannon distance
 - For numerical and categorical features
 - Output: distance, drift detected when distance >= threshold.
5. Click **Calculate**.
The drift analysis report is displayed.
 6. Click **Save**.
The reports are displayed under Data Drift Summary screen.

Disabling the Data Drift Monitoring

To stop the Data Drift monitoring, perform the following steps:

1. Navigate to **Data Drift Summary** page.
The data drift summary reports are displayed in a table.
2. Click **Stop Data Monitoring**.
A confirmation pop-up window is displayed.
3. Click **Yes**.
The Data Drift Monitoring is disabled.

Start the Data Drift Monitoring

To start the Data Drift monitoring, perform the following steps:

1. Navigate to **Data Drift Summary** page.
The data drift summary reports are displayed in a table.
2. Click **Start Data Monitoring** or **Data Drift Settings** icon.
The Data Drift Settings window is displayed.

Figure 11-32 Data Drift Settings

Data Drift Settings

Basic Information

Select reference dataset ?
Required

Select target dataset ?
05/31/2023, 06:49 PM

Drift Method ?
Kolmogorov-Smirnov Test

Dataset threshold (in %) ?
0

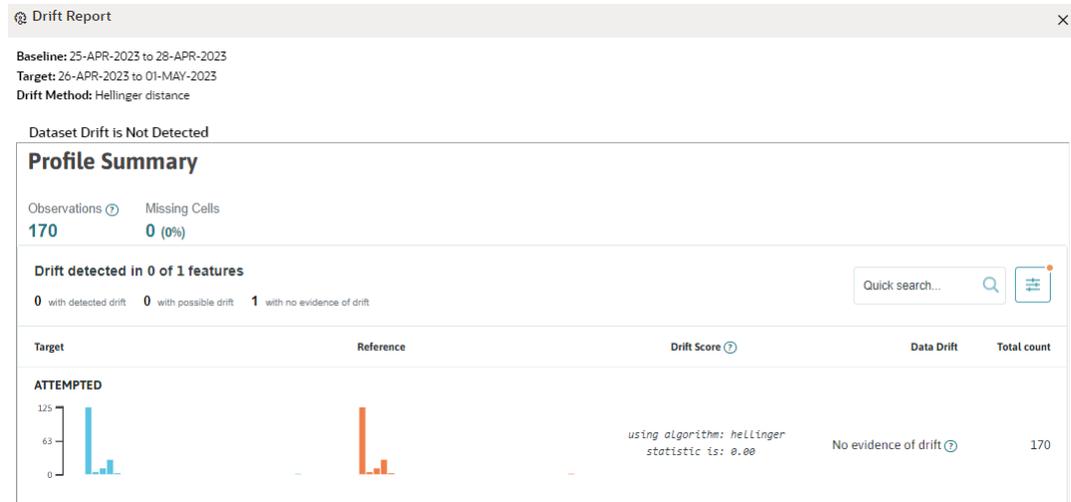
Schedule

3. Enter the basic information and schedule details and click **Save**.
For more details, see [Data Drift Summary](#) section.
The Data Drift Monitoring is started.

Viewing the Data Drift Summary Report

To view the Data Drift Summary Report, perform the following steps:

1. Navigate to **Data Drift Summary** page.
The data drift summary reports are displayed in a table.
2. Click **Action** next to corresponding drift report Id and select **View**.
The drift report is displayed in a graphical format.

Figure 11-33 Drift Report

Deleting the Data Drift Summary Report

To delete the Data Drift Summary Report, perform the following steps:

1. Navigate to **Data Drift Summary** page.

The data drift summary reports are displayed in a table.

2. Click **Action** next to corresponding drift report Id and select **Delete**.

OR

If you wish to delete multiple data drift reports, select reports and click **Delete**.

A confirmation pop-up window with report details is displayed.

3. Click **Delete**.

The Data Drift Report is deleted.

Fairness

You can calculate the fairness metrics of the dataset by choosing the target variable and protected features. The module provides metrics dedicated to assessing and checking whether the model predictions and/or true labels in data comply with a particular fairness metric. For this example, the statistical parity metric also known as demographic parity, measures how much a protected group's outcome varies when compared to the rest of the population. Thus, such fairness metrics denote differences in error rates for different demographic groups/protected attributes in data. Therefore, these metrics are to be *minimized* to decrease discrepancies in model predictions with respect to specific groups of people. Traditional classification metrics such as accuracy, on the other hand, are to be maximized.

You can perform the following functions:

- Measure Fairness Metrics of Dataset
- Models Bias Mitigation of Models
- Privacy Estimation of Models

For Example: In model pipeline, a new widget can be added to calculate the fairness metrics of a trained model or an advanced option can be added to model training widget to calculate the fairness metric of the model after training.

To calculate the dataset fairness, perform the following steps:

1. In the Fairness screen, select the Target Variable from the drop-down.
2. In the Protected features, select the required features to understand the fairness of the dataset.

Figure 11-34 Fairness screen

Metric	Calculated Value
Consistency	0.64
Smoothed EDF	0.83
Statistical Parity	0.17

The metric and calculated values are displayed. Click Help icon for more details on how the fairness is calculated.

Note

The fairness calculation in the dataset is only supported for python versions lesser than 3.12.

For python version 3.12, you will get the error *Error in calculating fairness metrics of dataset. No module named "guardian_ai"*. This is due to it's dependency on the older version of scikit-learn and numpy that are not supported on python version 3.12.

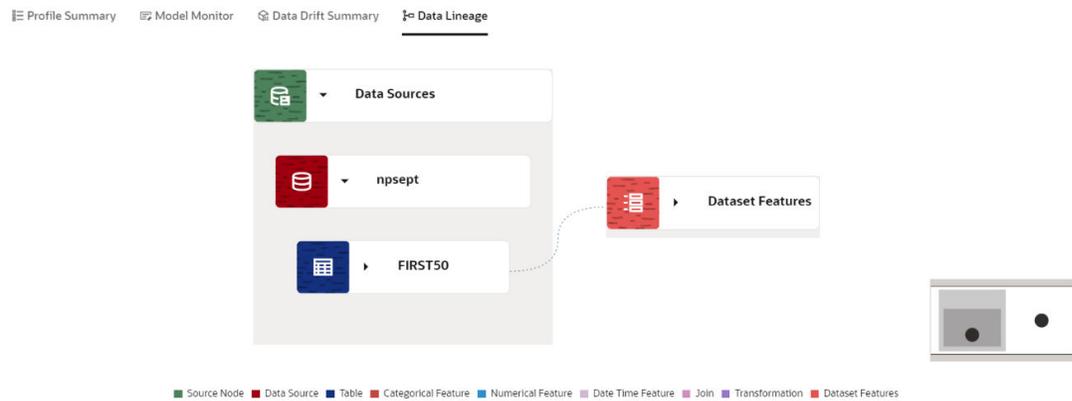
Data Lineage

Data Lineage provides a summary view or visualization of Data Sources and its associated dataset and features in a graphical format. Also displays the complete data flow from start to finish.

To view the data lineage, perform the following steps:

1. In the LHS menu, click **Dataset** to display the **Dataset Summary** window.
This window displays the dataset records in a table.
2. Click **Action** next to corresponding Dataset and select **View**.
The Profile Summary screen is displayed.
3. Navigate to **Data Lineage** page.

A summary view of Data Sources and its associated dataset and features in a graphical format.

Figure 11-35 Data Lineage page

Click the drop-down of the respective features to view granular details and its workflow.

Audit History

At any time, you can audit the datasets from the Audit History page. The Audit Trail window provides the complete details of datasets. The sequence of actions performed in the dataset lifecycle is listed in the table view and the timeline view (graphical representation).

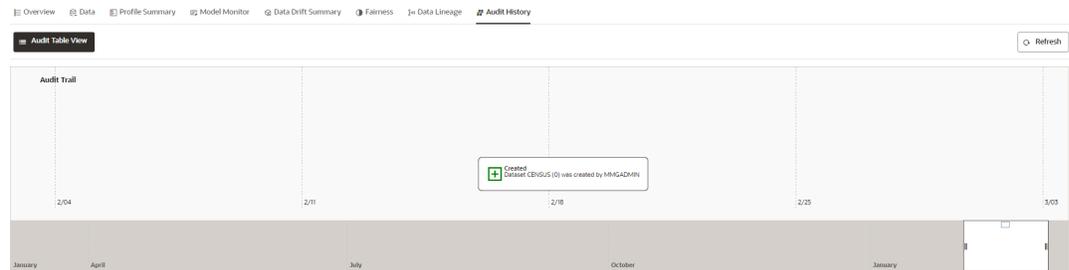
To audit datasets:

1. Click **Launch Workspace** next to corresponding Workspace to Launch Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Navigate to **Dataset Summary** page.
3. Click More Options icon next to corresponding Dataset and select View.
The **Overview** screen is displayed.
4. Navigate to **Audit History** page.

Figure 11-36 Audit History

Status	Type	Name	User	Comments	Time
Created	Dataset	CENSUS	MMGADMIN	Dataset CENSUS (0) was created by MMGADMIN	16 Feb 2024, 17:16:39

Click for **Audit Timeline View** for Timeline View and **Audit Table View** to switch to the regular view. Click to **Refresh** to refresh the Audit History. You can search the datasets by entering the name in the search box.

Figure 11-37 Audit History window Timeline View with Horizontal Time Axis

Create a Cache Snapshot

You can create a Cache Snapshot of the Dataset from this page.

To create a cache snapshot, follow these steps:

1. Navigate to Dataset Summary page.
2. Click next to corresponding Dataset and select View.
The Overview page is displayed.
3. Click on the header and select Create a cache snapshot.
4. Enter the **Snapshot Name** having alphanumeric character, underscore and hyphen not exceeding 30 characters and click Create.

This screen additionally allows users to cache a snapshot of the current state of data. The cached snapshots can be accessed later in Model Pipelines without any recompilation or re-read from data sources.

To use the dataset in model pipeline or data pipeline, the actual data is fetched using the Cache option. For example, to take the data from dataset on As of Date, create the data frame and provide the name to cache. Only Cache pulls the data from dataset.

This helps the things to work faster when you have millions of records, and you want to use intermediate data for use. For example, if you have 1 million records and want to use only 10,000 out of that, then perform the sampling for that 10,000 entries. This increases the speed of processing, validation of information.

- Once the metadata is created, the original data can be cached. A snapshot of the actual data in the dataset at the current time can be stored referenced by a tag name.
- The location for caching is in the datastudio server location `$DS_HOME/work/ftptshare/mmg/workspace_name/dscode/tag`. The dataset will be saved as a parquet file with name `dscode_tag.parquet`
- When executing all APIs from notebook, workspace has to be attached.
- Caching can be performed in two ways:
 - From UI, immediately after saving the metadata.

Or the users will have to fetch(create) a new snapshot/dataframe of the dataset using the API 'Fetch New Snapshot of dataset' and manually cache using the 'Caching Data Frame' API.

The following table provides information on the error and the troubleshooting procedure in case of dataset failure.

Table 11-6 Information on the error and the troubleshooting procedure

Error	Troubleshooting procedure
ModuleNotFoundError: No module named '_bz2'	Install the package 'libbz2-dev' before building python.
ModuleNotFoundError: No module named '_sqlite3'	Install the package 'libsqlite3-dev' before building python.
Python-env-health-check fails if pandas version is less than 1.4.1. NOTE: modin dataframe library is supported only from pandas version 1.4.1 and higher.	You can switch between the options modin[dask] and pandas for the underlying dataframe library if pandas version 1.4.1 is installed.
"Not a valid file" errorwhile profiling the Hive Data sources.	Copy the following required files : kbank.keytab , krb5.conf , hive-jdbc-driver.jar into the path : \$DS_HOME/conf folder of datastudio

Edit a Dataset

To edit a Dataset, follow these steps:

1. Navigate to **Dataset Summary** page.
2. Click on the Dataset Name column which you want to edit.

The Overview page is displayed.

3. Click **Edit**.

You can edit the Dataset fields except Code and Dataset Name.

Delete a Dataset

To delete a Dataset, follow these steps:

1. Navigate to **Dataset Summary** page.
2. Click next to corresponding Dataset and select **Delete**.

OR

Click next to corresponding Dataset and select **View**. You can delete a Dataset from this page.

Python functions for accessing Dataset from Model Pipelines/Notebooks

This section provides information on the Python functions for accessing Dataset from Model Pipelines/Notebooks.

List tags of Datasets Cached from UI

To get a list of all snapshots/tags of a dataset cached from UI, use the API below:

Table 11-7 Listing tags of Datasets Cached from UI

API	Notebook Script	Notes/Input
List Tags of a Dataset	<pre>from mmg.datasets import list_tags list_tags(dscore)</pre>	dscore = dataset code - string tag = tag with which the dataset was cached - string

Delete Cached Dataset

A Cached Dataset can be deleted by using the API below where tag refers to a particular snapshot/timestamp of the dataframe.

Table 11-8 Delete Cached Dataset

API	Notebook Script	Notes/Input
Delete Dataset	<pre>from mmg.datasets import delete_df delete_df(dscore, tag)</pre>	dscore = dataset code - string tag = tag with which the dataset was cached - string

List all Datasets with Metadata saved from UI

Table 11-9 List all Datasets with Metadata saved from UI

API	Notebook Script	Notes/Input
List Datasets	<pre>from mmg.datasets import list_datasets list_datasets()</pre>	

Fetch Dataset in Notebook

Dataset whose metadata has been saved from UI can be fetched in two ways:

1. Dataset that has already been cached from UI can be fetched using first API below by providing the dataset code and tag.(DataFrame will not be recalculated. It will be read from cache.)
2. A new data frame for a dataset can be calculated/fetched using data from the present time with the metadata saved for that dataset as given in second API.

Table 11-10 Fetching Dataset in Notebook

API	Notebook Script	Notes/Input
Fetch Dataset Cached from UI	<pre>from mmg.datasets import fetch_ds df = fetch_ds(dscore,tag)</pre>	dscore = dataset code - string tag - string
Fetch New Snapshot of Dataset	<pre>from mmg.datasets import fetch_ds df = fetch_ds(dscore)</pre>	dscore = dataset code - string

Cache User's Dataframe from Notebook

A user-made data frame can also be cached using the below API. It will be stored in the datastudio server in location : `$DS_HOME/work/ftpshare/mmg/workspace_name/cached/tag`. The dataset will be saved as a parquet file with name `cached_tag.parquet`.

Note

This dataframe is not related to dataset created from UI. This is independent of dataset metadata.

Table 11-11 Cache User's Dataframe from Notebook

API	Notebook Script	Notes/Input
Caching Data Frame	<pre>from mmg.datasets import cache_df path=cache_df(df, tag) path</pre>	<p>df = dataframe to be cached - pandas dataframe</p> <p>tag - string</p>

Fetching Data Frame Cached from Notebook

The data frame cached from notebook can be fetched using the below API.

Table 11-12 Fetching Data Frame Cached from Notebook

API	Notebook Script	Notes/Input
Fetch dataset cached manually	<pre>from mmg.datasets import fetch_ds df = fetch_ds("cached",tag)</pre>	<p>tag - string</p>

List Tags of all Data Frames Cached from Notebook

Table 11-13 List Tags of all Data Frames Cached from Notebook

API	Notebook Script	Notes/Input
List Tags of Manually Cached Datasets	<pre>from mmg.datasets import list_tags list_tags()</pre>	

Model Libraries

The Model Library information is used to bind a particular technique and its details to one unique Model Library. The Starting point for the Model builder is to register a model library with the application after it is properly set up in the python environment to be used.

Currently, the following libraries are supported:

- keras

- ONNX
- scikit-learn
- xgboost

As an example, the following section provides information on how to set up and register the scikit-learn library.

Setting up the Python Library in the python environment:

Open the terminal and install the scikit-learn library using the following command:

```
python3 -m pip install scikit-learn
```

Note

Proxy might be required to install the packages from pip.

Once the installation is complete, check for the package details using the following command:

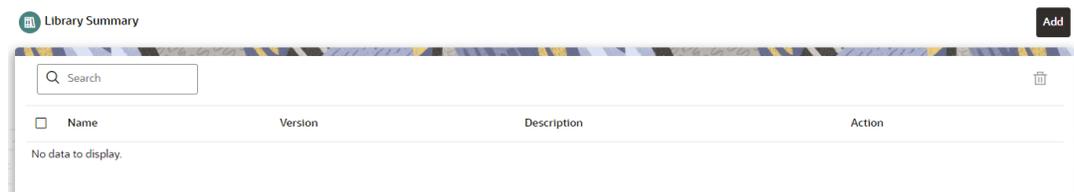
```
pip show scikit-learn
```

Once the installation is complete in Python environment, you need to register the library into the application.

Registering the Python Library

1. In the Mega menu, click Modeling > Model Libraries.
The **Library Summary** page is displayed.

Figure 11-38 Library Summary page



2. In the **Library Summary** page, click **Add**.
The **Add Library** page is displayed.

Figure 11-39 Add Library page

Add Library

[Use Template](#)

Name Required

Version Required

Description

Signatures

Import Signature ?

Load Signature ?

Train Signature ?

Save Signature ?

Infer Signature ?

3. Enter the name of the library.
4. Enter the version and description of the library.
5. Enter the signatures such as Import, Load, Train, Save, and Infer. For more details on the signatures, click the respective help icons. Some of the signatures captured in library stage might not be standard across different algorithm/techniques provided by the library.
6. Click **Create**.
The Model Objective is created and displayed in the Model Objective screen.

Note

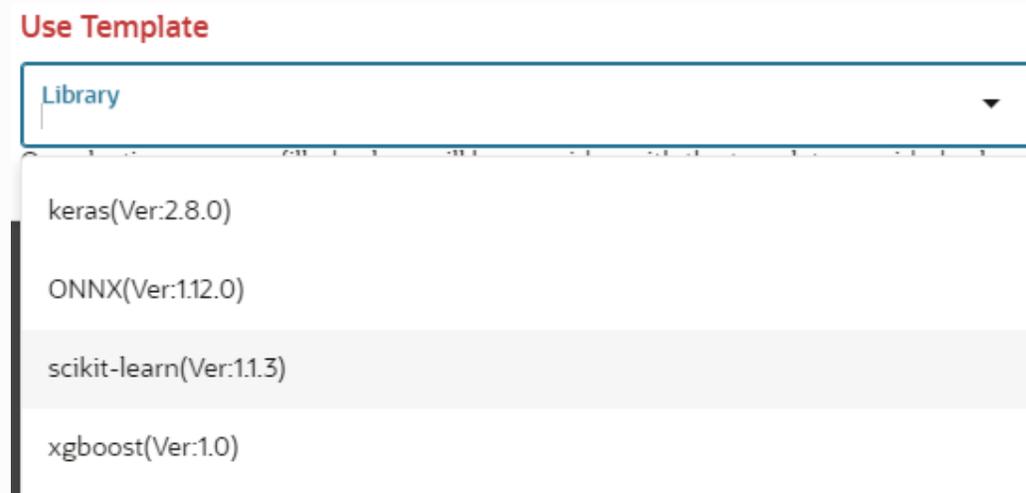
The User must check the details provided above from the home page of the library being captured.

For example: <https://scikit-learn.org/stable/>.

OR

You can select the **Use Template** option to pre-fill the entries from the seeded list of Libraries. Currently, the following libraries are present in the template.

Figure 11-40 Use Template



You can also edit the data based on your requirements.

Note

The .whl file for MMG-Python is available in the below location:
OFS_MMG/mmg-service/python-lib/mmg-8.1.3.2.0-py3-none-any.whl

This can be installed using the command `Python3 -m pip install MMG-8.1.3.2.0-py3-none-any.whl`

On successful installation, the user will get a message: "Successfully installed MMG-8.1.3.2.0".

Edit a Model Library

To edit a Model Library:

1. Navigate to **Library Summary** page.
2. Click and select Edit.

The **Edit Library** screen is displayed.

Figure 11-41 Edit Library screen

3. Make the necessary changes and click **Update**.

Note

The Model Library name and version are unique and cannot be modified.

Delete a Model Library

To delete a Model Library:

1. Navigate to **Library Summary** page.
2. Click and select **Delete**.

A confirmation message is displayed.

Figure 11-42 Confirmation Message

Code	Name	Description
<input checked="" type="checkbox"/> xgboost	xgboost	Xgboost Library for Modeling
<input checked="" type="checkbox"/> xgboost(Ver:1.0)		

3. Click **Delete**.

The selected model library and all the associated model techniques are deleted. The libraries that are deleted cannot be consumed by any model catalog.

To delete multiple model libraries select all the model libraries that you want to delete and click the icon on the page header.

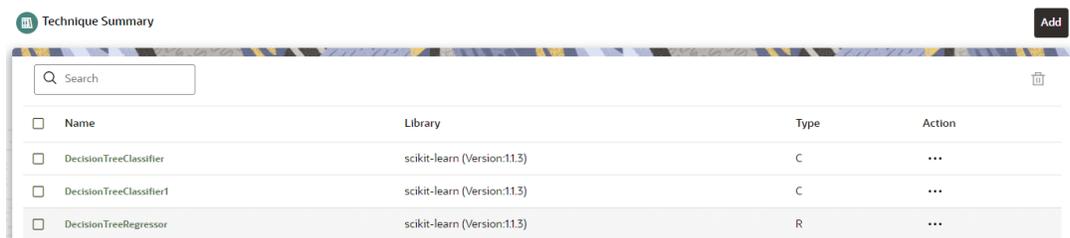
Model Technique

Model Technique is the algorithm/technique used to create python model using the library/package which was created in the Model Library Screen. It is the actual information captured in the application that helps in training the model (Upload and Build).

1. In the LHS menu, click **Model Catalog > Model Technique** option.

The **Technique Summary** page is displayed.

Figure 11-43 Technique Summary page



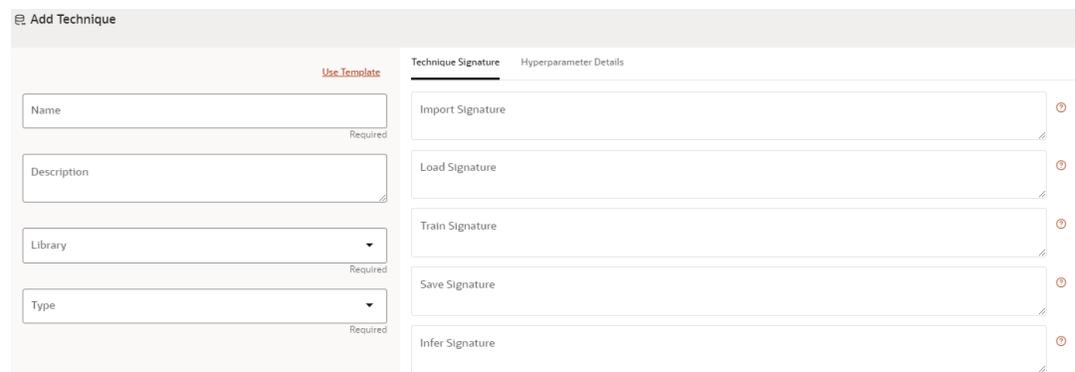
The screenshot shows the 'Technique Summary' page with a search bar and a table of techniques. The table has columns for Name, Library, Type, and Action. The 'Add' button is visible in the top right corner.

Name	Library	Type	Action
DecisionTreeClassifier	scikit-learn (Version:11.3)	C	...
DecisionTreeClassifier1	scikit-learn (Version:11.3)	C	...
DecisionTreeRegressor	scikit-learn (Version:11.3)	R	...

2. In the **Technique Summary** page, click **Add**.

The **Add Technique** page is displayed.

Figure 11-44 Add Technique page



The screenshot shows the 'Add Technique' page with a 'Use Template' link and two tabs: 'Technique Signature' and 'Hyperparameter Details'. The 'Technique Signature' tab is active, showing several text input fields for signatures.

Field	Requirement
Name	Required
Description	
Library	Required
Type	Required
Import Signature	
Load Signature	
Train Signature	
Save Signature	
Infer Signature	

3. Enter the name of the technique.

4. Enter the description of the technique.

5. Select the library from the drop-down. Currently, the application supports the libraries such as keras, ONNX, scikit-learn, and xgboost.

6. Select the type as either as **Classification** or **Regression**.

Classification: Classification is a process of finding a function which helps in dividing the dataset into classes based on different parameters. The task of the classification algorithm is to find the mapping function to map the input(x) to the discrete output(y).

Regression: Regression is a process of finding the correlations between dependent and independent variables. The task of the Regression algorithm is to find the mapping function to map the input variable(x) to the continuous output variable(y).

7. Enter the signatures such as Import, Load, Train, Save, and Infer. For more details on the signatures, click on the respective help icon.

Some of the signatures captured in library stage might not be standard across different algorithm/technique provided by the library.

8. Navigate to **Hyperparameter Details** tab and click **Add** to add the parameters.

You can add the type of parameters such as String, Integer, Float, and Boolean.

9. Click **Create**.

The Model Technique is created and displayed in the **Technique Summary** screen.

Note

The user needs to check the details provided above from the homepage of the library being captured.

For example: <https://scikit-learn.org/stable/>.

OR

You can select the Use Template option to pre-fill the entries from the seeded list of Libraries.

You can also edit the data based on your requirements.

Edit a Model Technique

To edit a model technique, follow these steps:

1. Navigate to **Technique Summary** page.
2. Click **Action** next to corresponding model technique and select **Edit**.

The **Edit Technique** screen is displayed.

Figure 11-45 Edit Technique screen

3. Make the necessary changes and click **Update**.

For more details, see [Model Technique](#) section.

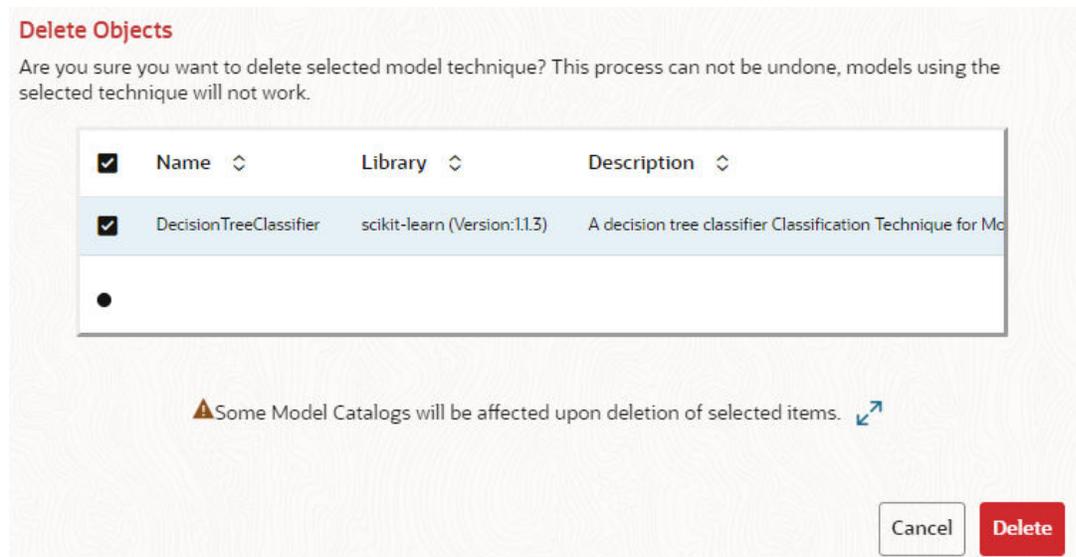
Delete a Model Technique

To delete a model technique, follow these steps:

1. Navigate to **Technique Summary** page.
2. Click **Action** next to corresponding model technique and select **Delete**.

A confirmation message is displayed.

Figure 11-46 Confirmation Message



3. Click **Delete**.

The selected model technique and all the model catalogs associated are deleted.

To delete multiple model techniques, select all the model techniques that you want to delete and click the **Delete** icon on the page header.

Model Catalog

The Model Catalog page allows you to add and manage the Model Technique, Model Library, and Model Objectives. You can either import the models from external sources or create, train and label it as champion model.

Model Objective

Model Objective is the top level metadata where further models are created or trained which can be consumed in the upcoming steps.

Note

Ensure Data Studio is up and running. A validation message is displayed when the Data Studio is down during creation of Model Objectives.

To add a Model Objective:

1. Click **Launch Workspace** next to the corresponding workspace to display the **Dashboard** window with application configuration and model creation menu.
2. In the LHS menu, click **Model Catalog** to display the **Model Objective Summary** window. This window displays the model objectives in a table. The following table provides descriptions for the fields and icons on the **Model Objective Summary** page.

Table 11-14 Fields and icons on the Model Objective Summary page

Field or Icon	Description
Search	The field to search for Model Objectives. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.
Code	The unique identifier of the Model Objective.
Name	The name of the Model Objective.
Dataset	The dataset which is used to create the Model Objective.
Target Variable	Target Variable is the important parameter after the dataset. During the dataset selection, target variable is selected that constitutes of number of columns/features.
Type	The type which is selected during model objective creation. The type can be either Classification or Regression.
Action	Click the three dots to perform View/Delete functions on the selected model objective..

Note

In the objective summary UI, a new button is introduced to show/hide empty objectives. By default "Show Empty Objectives" will be disabled (i.e. empty objectives wont be shown) User can enable this checkbox to view empty objectives. On successful addition of new objectives this checkbox is enabled automatically.

Add a Model Objective

To add a Model Objective:

1. In the **Model Objective Summary** screen, click **Add**. The **Add Model Objective** screen is displayed.

Figure 11-47 Add Model Objective screen

The screenshot shows a web form titled "Add Model Catalog" with a close button (X) in the top right corner. The form contains the following fields:

- Select Dataset:** A dropdown menu with a downward arrow. A "Required" label is positioned to the right of the field.
- Code:** A text input field. A "Required" label is positioned to the right of the field.
- Name:** A text input field. A "Required" label is positioned to the right of the field.
- Select Target Variable:** A dropdown menu with a downward arrow. A "Required" label is positioned to the right of the field.
- Type:** A dropdown menu with a downward arrow. A "Required" label is positioned to the right of the field.
- Description:** A text input field.

2. Select the required dataset from the drop-down.
For more details on the dataset, see [Dataset](#) section.
3. Enter the code for the Model Objective.
The code should be unique for each model objective.
4. Enter the name and description for the Model Objective.
5. Select one of the column/feature as the target variable for the models which needs to be trained/uploaded.
6. Select the Type from options such as **Classification** , **Regression**, or **Others**.

Classification: Classification is a process of finding a function which helps in dividing the dataset into classes based on different parameters. The task of the classification algorithm is to find the mapping function to map the input(x) to the discrete output(y).

Regression: Regression is a process of finding the correlations between dependent and independent variables. The task of the Regression algorithm is to find the mapping function to map the input variable(x) to the continuous output variable(y).

Others:If the Type is not Classification or Regression, select this option.
7. Enter the description for the Model Objective.
8. Click **Create**.
The Model Objective is created and displayed in the **Model Objective Summary** screen.

Note

In the objective summary UI, a new button is introduced to show/hide empty objectives. By default "Show Empty Objectives" will be disabled (i.e. empty objectives wont be shown) User can enable this checkbox to view empty objectives. On successful addition of new objectives this checkbox is enabled automatically.

View a Model Objective

To view a Model Objective:

1. Navigate to **Model Objective Summary** page.
2. Click **Action** and select **View**.

OR

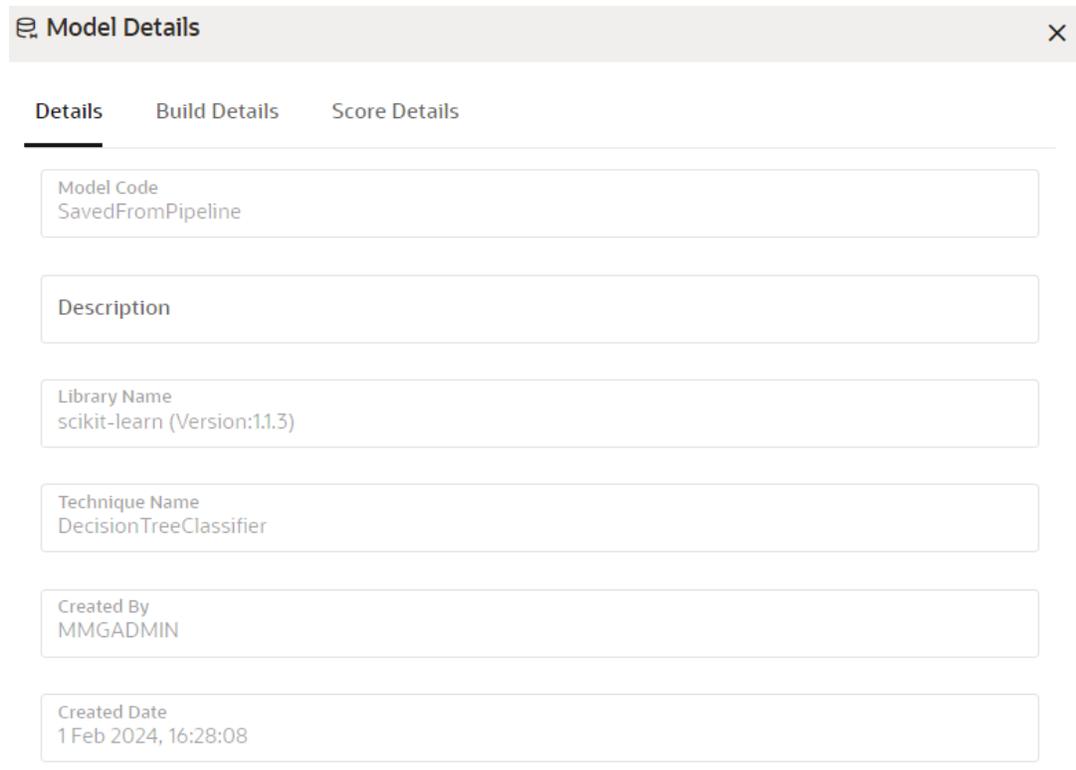
Click on the Model.

The following page is displayed.

Figure 11-48 View page

<input type="checkbox"/>	Model ID	Code	Version	Type	Status	Action	Champion
<input type="checkbox"/>	1706785088326	SavedFromPipeline	0	Pipeline Model		...	👤
<input type="checkbox"/>	1698309977652	TESTM	0	Local Model	✔	...	👤
<input type="checkbox"/>	1706785135628	UPLOADEDEXAMPLE	0	External Model		...	👤

Click on Type to view the model, build, and score details.

Figure 11-49 Model details

The screenshot shows a dialog box titled "Model Details" with a close button (X) in the top right corner. Below the title bar are three tabs: "Details", "Build Details", and "Score Details". The "Details" tab is selected and underlined. The dialog contains six input fields, each with a label and a value:

- Model Code: SavedFromPipeline
- Description: (empty)
- Library Name: scikit-learn (Version:1.1.3)
- Technique Name: DecisionTreeClassifier
- Created By: MMGADMIN
- Created Date: 1 Feb 2024, 16:28:08

Upload a Model

To upload a model:

1. In the **View Model Objective** page, click **Add** and select **Upload Model**.
The **Upload Model** page is displayed.

Figure 11-50 Upload Model page

Upload Model

Select Library
ONNX(Ver:1.12.0)

Select Technique

Model Code

Description

Model File Name

Upload from FTPSHARE

Import File

Import Archive File
(Drag & Drop file here)

Required

2. Select the library from the drop-down.
For more details on the model library, see the [Model Libraries](#) section.
3. Select the required technique from the drop-down.
For more details on the model technique, see the [Model Technique](#) section.
4. Enter the model code and description for the Model Objective.
5. Enter the name of model file with extension for use during Save and Load.
6. Import the model using **Import Archive File** option. File format should in be zip format.

OR

Enable **Upload from FTPSHARE** option to import the model. File Path entered is the zip file path present on the Server relative to ftpshare path including file.
Example: <installation path>/temp/models/model.zip

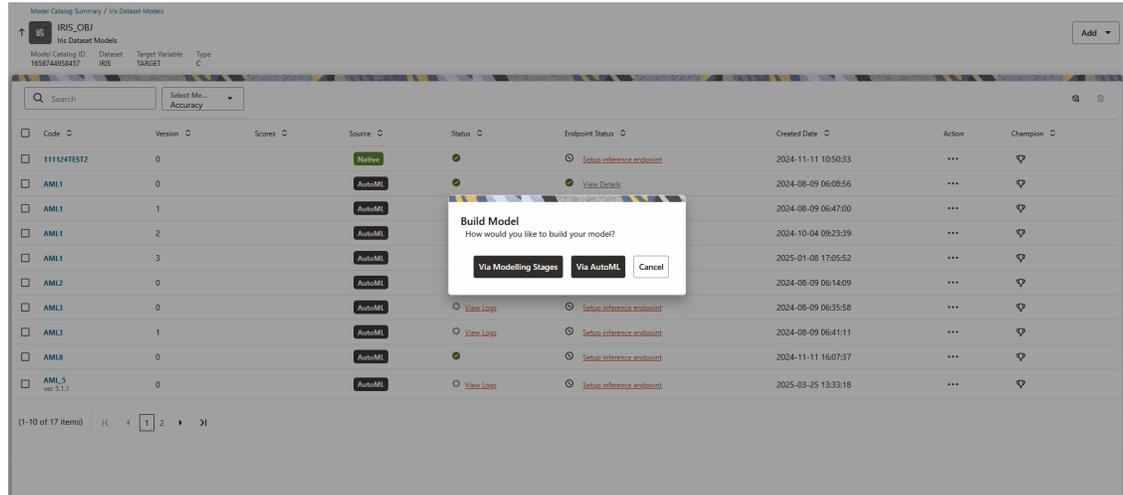
7. Click **Upload Model**.
The model is uploaded successfully.

Build a Model

Building a model consists of the following steps:

- [Model Details](#)
- [Pre-Processing Stage](#)
- [Model Training Stage](#)
- [Model Validation Stage](#)
- [Model Summary](#)

Figure 11-51 Model Building screen



Build via Modelling Stages

This section describes about building a model using traditional Machine Learning steps.

Model Details

1. In the **View Model Objective** page, click **Add** and select **Build Model**.

The **Model Details** page is displayed.

Figure 11-52 Model Details page

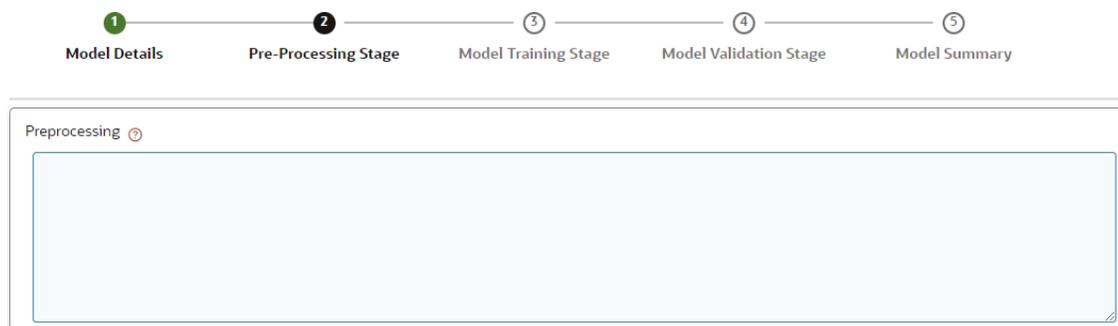
The diagram shows a horizontal navigation bar with five steps: 1. Model Details, 2. Pre-Processing Stage, 3. Model Training Stage, 4. Model Validation Stage, and 5. Model Summary. Below the navigation bar is a form with five fields: 'Select Library' (Required), 'Select Technique' (Required), 'Model Code' (Required), 'Description', and 'Model File Name' (Required).

2. Select the library from the drop-down. Currently, the supported libraries are keras, ONNX, scikit-learn, and xgboost. For more details, see the [Model Libraries](#) section.
3. Select the required technique from the drop-down. For more details, see the [Model Technique](#) section.
4. Enter the model code and description for the model.
5. Enter the name of model file with extension for use during Save and Load.
6. Click **Next** to go to the next step.

Pre-Processing Stage

After adding the details in the Model Details page, the Pre-Processing Stage screen is displayed.

Figure 11-53 Pre-Processing Stage screen



1. Enter the data in the Pre-Processing text box as shown in an example is provided here.

Figure 11-54 Pre-Processing text box

Pre-Processing Help:

1. The pre-processing block will be plain python script which will be executed before model train signature.
e.g. `print('inside pre-processing block')\nprint('pre-processing done')`
2. For line breaks in the python script use `\n`
3. Based on the dataset provided in the model objective, user will get data in form of pandas dataframe which can then be used in pre-processing block.
4. `'mmg_dataset_df_X'` is the variable name for training data which can be used to perform some operations on it but final variable that will be used in train signature will be `'mmg_dataset_df_X'` only.
e.g. `mmg_dataset_df_X = some_func(mmg_dataset_df_X)`

2. Click **Next** to go to the next step.

Model Training Stage

After adding the details in the Pre-Processing Stage page, the Model Training Stage screen is displayed.

Figure 11-55 Model Training Stage screen

Parameter	Value	Placeholder	Enabled
kernel (String)	poly	\$kernel\$	<input checked="" type="checkbox"/>
degree (Integer)...	3	\$degree\$	<input checked="" type="checkbox"/>
C (Float)	11	\$C\$	<input checked="" type="checkbox"/>
shrinking (Boolean)		\$shrinking\$	<input type="checkbox"/>
probability (Boolean)	True	\$probability\$	<input checked="" type="checkbox"/>
gamma (String)		\$gamma\$	<input type="checkbox"/>
C (Float)		\$C\$	<input type="checkbox"/>

Generate Train Signature

```
SVC(C=11,kernel=poly,degree=3,probability=True),fit($df_X,$df_Y)
```

1. Select the hyperparameters.
When the parameters is selected, the **Enabled** check box is selected by default.
2. Click **Generate Train Signature**.
The signature is displayed.
3. Click **Next** to go to the next step.

Model Validation Stage

After adding the details in the Model Training Stage page, the Model Validation Stage screen is displayed.

Figure 11-56 Model Validation Stage screen

No data to display

The Validation is work in progress.

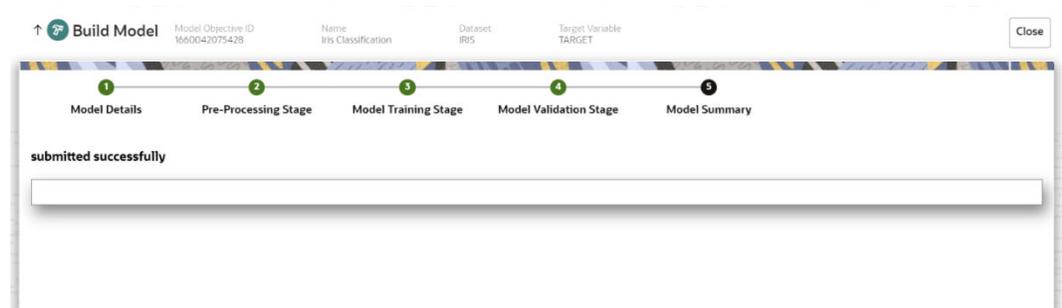
- Click **Finish** to start the training of the model.

Model Summary

Once the validation of the model is completed, the Model Summary screen is displayed. Follow the below steps to publish Models from Model Summary:

1. Navigate to Model Summary.
2. Click on the three dots.
3. Select publish Datastudio option.
4. Check the links.

Figure 11-57 Model Summary screen



- Click **Close**.

The model is created and displayed in the Model Objective screen.

Build via AutoML

The AutoMLx python package automatically creates, optimizes and explains machine learning pipelines and models. The AutoML pipeline provides a tuned ML pipeline that finds the best model for a given training dataset and a prediction task at hand. AutoML has a simple pipeline-level Python API that quickly jump-starts the data science process with an accurate tuned model.

The AutoML pipeline consists of five major stages of the ML pipeline: preprocessing, algorithm selection, adaptive sampling, feature selection, and model tuning.

These pieces are readily combined into a simple AutoML pipeline which automatically optimizes the whole pipeline with limited user input/interaction.

Note

The user needs to manually install the Oracle-AutoMLX Python package. The Python environment health check fails if install oracle-automlx[classic]==25.2.1

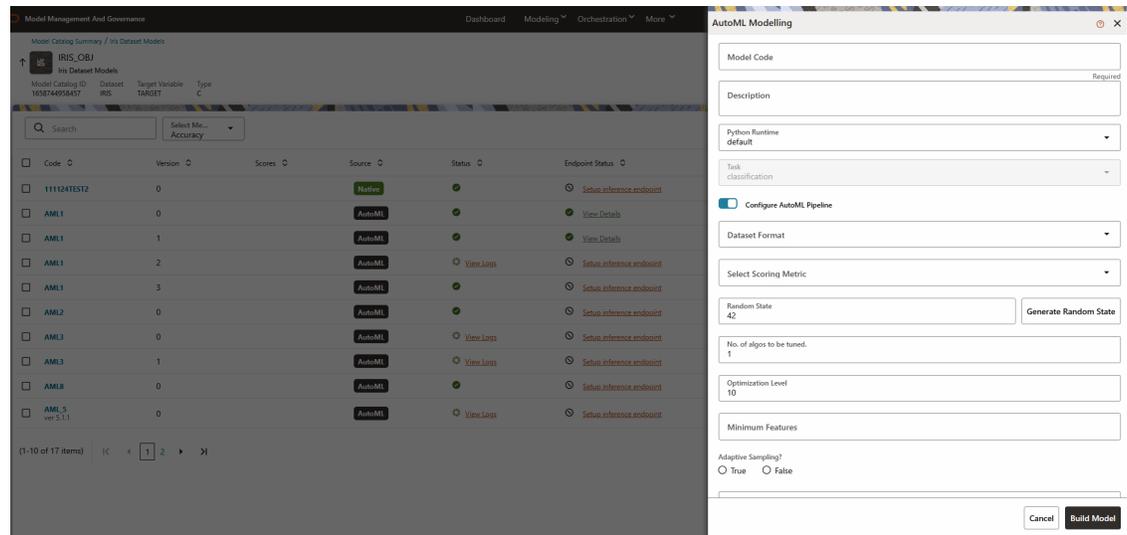
The following is the command to install AutoML: `python3 -m pip install oracle-automlx[classic]==25.2.1`

Installation of AutoML

To use Model Building via AutoML, the oracle-automlx version 25.2.1 package must be installed. Some existing packages will automatically be downgraded from their current versions. For a smoother experience, users are recommended to create a Conda environment and install the required packages.

For automlx installation refer to: <http://automl.oraclecorp.com/multiversion/v25.2.1/installation.html>

Figure 11-58 Build via AutoML



Follow these steps to build via AutoML:

1. Enter the **Model Code** for the model.
This field is mandatory.
2. Enter the **Description** for the model.
3. Click on the **Python Runtime** dropdown to select any Conda environment.
You can select the environment only if it is created. Otherwise it will be a default environment.
4. Click the **toggle** button to configure the pipeline.

Note

You can enter the required inputs for all the parameters. However, they are not mandatory. You can also click the Help icon on the top right of the drawer for more details on each one of them.

5. Click on the **Build Model**.
An AutoML model will be created and displayed in the **Model Objectives** page.

View Details of a Model

To view details of a model, follow these steps:

1. In the **View Model Objective** page, click **Action** next to corresponding model and select **View Details**.

The model details are displayed.

Figure 11-59 Model details

The screenshot shows a 'Model Details' window with the following fields:

- Model Code:** SavedFromPipeline
- Description:** (Empty field)
- Library Name:** scikit-learn (Version:1.1.3)
- Technique Name:** DecisionTreeClassifier
- Created By:** MMGADMIN
- Created Date:** 1 Feb 2024, 16:28:08

2. Click on the **Build Details** tab to view the Preprocessing, and Hyperparameters details of the model.
3. Click the **Score Details** tab to view the Latest score, and Scoring details of the model.

Note

The Download option is now available for Model Catalog-Upload/Build Model. In the View Model Objective page, click on *three dots* under *Action* next to corresponding model and select *Download option*. The models get downloaded in a zip format.

Setting the Model as Champion

You can set the trained models as champion model.

To set a champion model:

1. In the **View Model Objective** page, hover over the model which you want to set as champion and click the **Set Champion** icon.

Figure 11-60 View Model Objective page

Model ID	Code	Version	Type	Status	Action	Champion
<input checked="" type="checkbox"/> 1706785088526	SavedFromPipeline	0	Pipeline Model		...	<input type="checkbox"/>
<input type="checkbox"/> 1698309977652	TESTM	0	Local Model	●	...	<input checked="" type="checkbox"/> Set Champion
<input type="checkbox"/> 1706785335628	UPLOADEDEXAMPLE	0	External Model		...	<input type="checkbox"/>

A confirmation message is displayed.

2. Click **Update**.

The model is set as champion for the selected model objective.

Delete a Model

To delete a model:

1. In the **View Model Objective** page, select the models which you want to delete.
2. Click **Delete** and acknowledge the confirmation message.
3. Click **Delete**.

Delete a Model Objective

To delete a Model Objective:

1. Navigate to **Model Objective Summary** page.
2. Click **Delete** and acknowledge the confirmation message.
3. Click **Delete**. The selected model objective and all its associated models are deleted.

Pipelines

Model Pipeline enable you to create and publish models based on the workspaces created from datasources. The published models are then deployed in production to be consumed by other services and applications. Modelers create models by using model templates available as part of ML4AML or by developing them independently in Compliance Studio Notebooks. After building and evaluating, multiple models, the best model or champion model can be selected. The champion model can then be deployed for scoring. In this document the champion model can also be referred to as the scoring model.

Prerequisites

The prerequisites for model pipeline are as follows:

- To create a model, your user profile must be mapped to the Modeler Group.
- To create a model, a workspace must be deployed.
- To approve and deploy a model, your user profile must be mapped to the Modeling Administrator Group.

Access Model Pipelines

The Model Pipeline window allows you to create and publish models.

To access the Model Pipeline window:

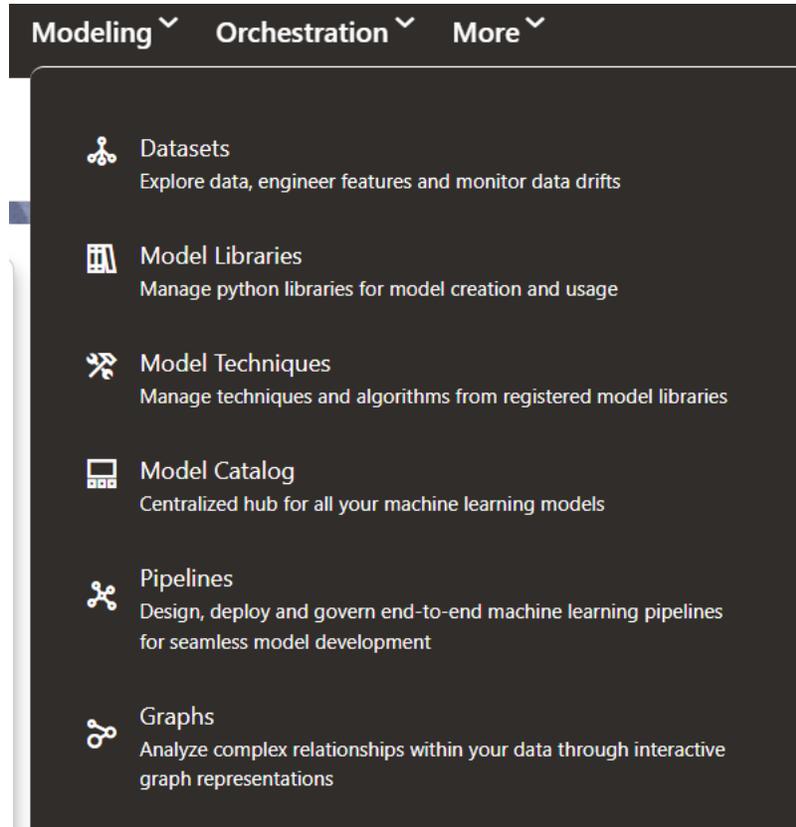
1. Navigate to **Workspace Summary** page.

The page displays workspace records in a table.

2. Click **Launch** next to corresponding Workspace to Launch Workspace.

The **Dashboard** window is displayed with application configuration and model creation menu.

- Click the **Modeling** drop-down list at the top of the page and then click **Pipelines**.



The window displays objectives that contain drafts and models. When you hover over the count that are adjacent to the **ID** column, it displays the count of sub objectives, drafts, models, and champion, if available.

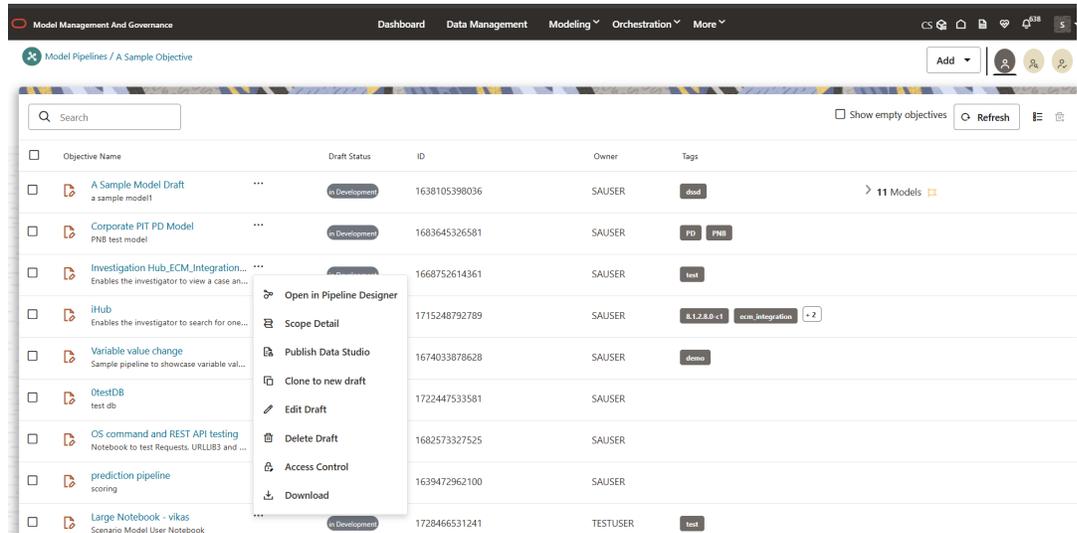
Figure 11-61 Model Pipelines page

The screenshot shows the Oracle AIF4AML Model Pipelines page. The page has a search bar and a table of objectives. The table has the following columns: Objective Name, ID, and a count of sub-objectives. The table lists five objectives:

Objective Name	ID	Count
AIF Admin Admin Activity for AIF4AML	AIF0000018	1
AIF Batch Framework Batch Framework for AIF4AML	AIF0000005	12, 1, 1, 1
AIF Big Data AIF Data Aggregation in Big Data	AIF0000020	
AIF Graph Analytics PGX for AIF4AML	AIF0000019	

Click the **More** icon adjacent to an objective to view all the available actions.

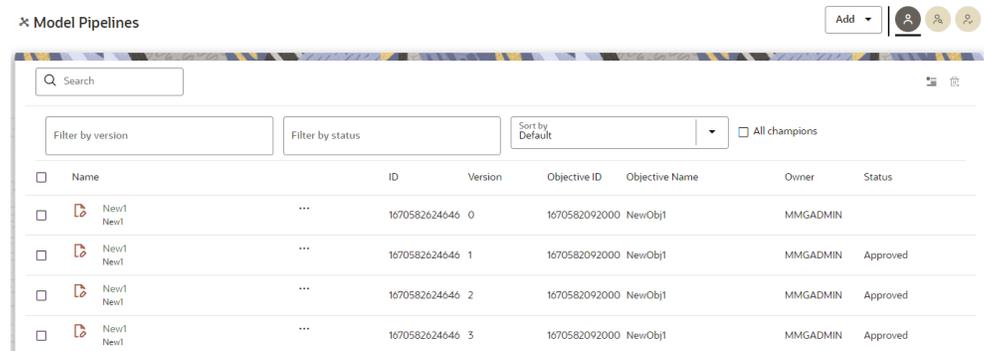
Figure 11-62 Actions for an Objective



You can click the **Flat List** icon, adjacent to the **Refresh** button, to change the view. Flat list is the default view in the production workspace while hierarchical list is the default in the sandbox.

- **Hierarchical:** This option displays the hierarchical list of objectives. In the **Hierarchical** view, you can view the details of the objectives such as the objective name and ID.
- **Flat List:** This option displays the flat list of all the models (published and drafts) by using the **Flat List** icon. Flat list is not objective-specific. It displays the models across all the objectives. You can search the models by using the **Filter by Version**, **Filter by Status**, **Sort by**, and **All champions**. The **Sort by** option contains two sorting options; **Default** and **Latest first**.
 - In the flat list view of models, you can see the details of the models such as the name, ID, version, objective ID, objective name, owner, and status.

Figure 11-63 Model Pipelines



The following table provides descriptions for the fields and icons on the Model Pipeline page.

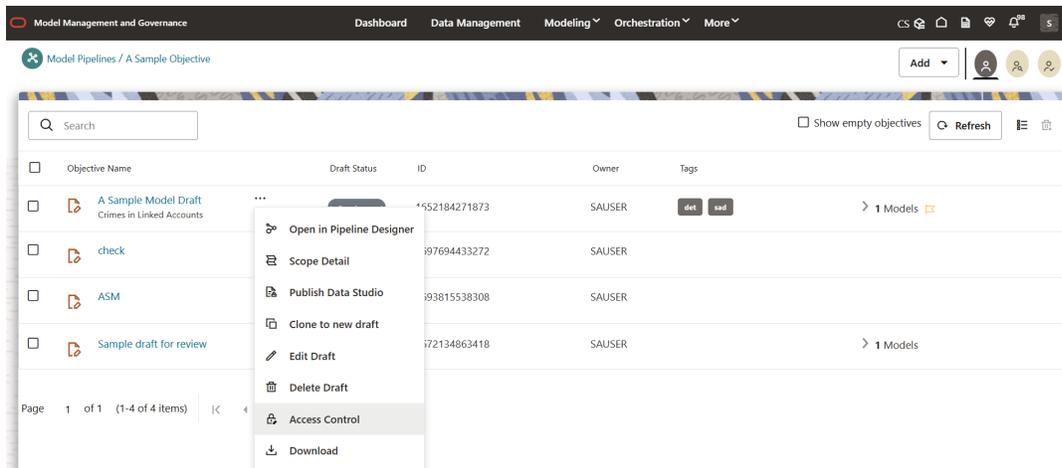
Table 11-15 Fields and icons on the Model Pipeline page

Field or Icon	Description
Model Pipeline Page Header	The header that follows the global header in the application.
Breadcrumbs	Indicates the position of the current page in the Model Management hierarchy. Use breadcrumbs locator links to navigate back to higher levels in the hierarchy after you have drilled down through levels of functions. Click to navigate to Model Summary page. Click or to navigate back to Workspace Summary page.
Create Objectives and Models	Select Add to display a list with the following options: <ul style="list-style-type: none"> Draft Objective Seeded Models To create models, select Draft . To create objectives, select Objective . See the Create Objective (Folders) and Create Draft Models sections for more information. However, if you want to know about the cycle of model creation, see the Create, Review, Approve, and Deploy a Model section.
Requester	Displays that the logged-in user has the requester privileges when the status is green. You can create a model. However, to approve and publish, the model must be reviewed by a user with reviewer privileges and approved by a user with approver privileges.
Reviewer	Displays that the logged-in user has the reviewer privileges when the status is green. You can review models. However, to approve and publish, the model must be approved by a user with approver privileges.
Approver	Displays that the logged-in user has the approver privileges when the status is green. You can approve models that are created and reviewed.
Model Pipeline Page Table	The table displays the objective and model records on the page.
Objective Name	Displays the name and description of the objective
ID	Displays the ID of the objective.
Owner	Displays the owner who created the drafts. This information does not display for an objective.
Tags	Displays the tags associated with the models or drafts. This information does not display for an objective.
Delete Objective	Select to delete the model from the confirmation dialog box. Click Delete to process or click Cancel to cancel. This information is not displayed for an objective.
Edit Objective	Select this option to edit the models in Data Studio. See the Edit Models section for more information. This information is not displayed for an objective.

Note

When executing PGX, the user is unable to access the interpreter 'PGX-Java' for the Notebook type 'Jupyter'. The allowed interpreters are: Python and MD. However, when the user changes the notebook type from the **Notebook** interface, the user is able to access the PGX-Java.

Figure 11-64 Access Control Menu option



The **Access Control** window has two radio buttons: **Public** and **Restricted**.

Figure 11-65 Public Mode

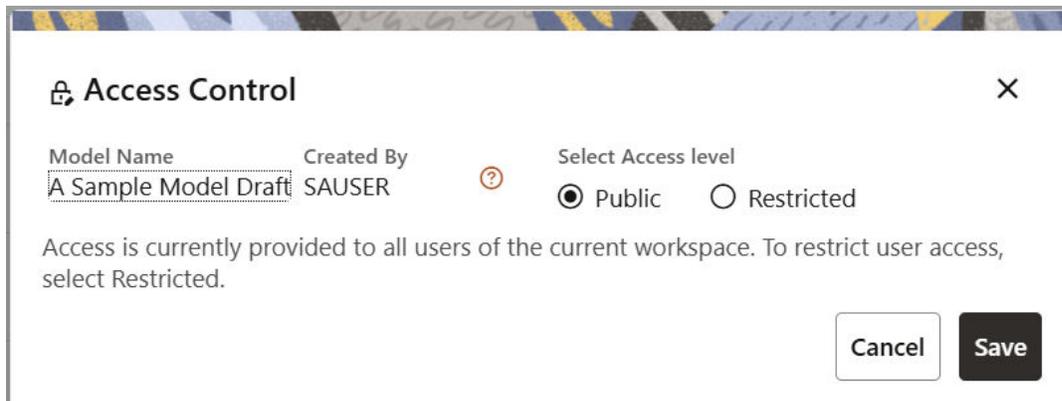


Figure 11-66 Restricted Mode

Access Control

Model Name Created By
A Sample Model Draft SAUSER

Select Access level
 Public Restricted

Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public.

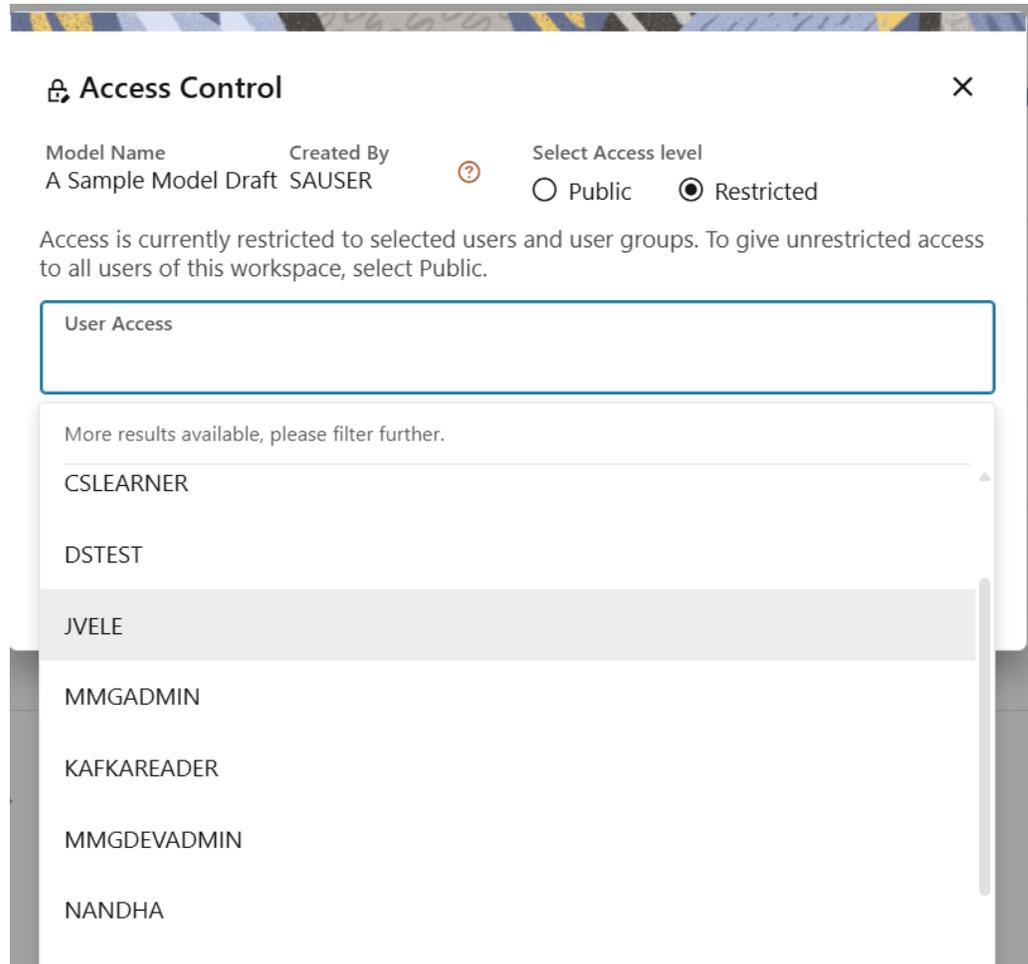
User Access

User Group Access

Cancel Save

The **Access Control** window contains the following details:

- **Model Name**
- **Created By**
- **Select Access level: Public and Restricted**
- **User Access (Restricted only):** Select the users with restricted access. This drop-down list fetches all the users present in the Identity Management.

Figure 11-67 Access Control User Dropdown

- **User Group Access (Restricted only):** Select the user groups with restricted access. This drop-down list fetches all the user groups present in the Identity Management.

Figure 11-68 Access Control User Dropdown

The screenshot shows a dialog box titled "Access Control" with a close button (X) in the top right corner. Below the title, there are three fields: "Model Name" with the value "A Sample Model Draft", "Created By" with the value "SAUSER", and "Select Access level" with two radio buttons: "Public" (unselected) and "Restricted" (selected). Below these fields, a message states: "Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public." A search box labeled "User Access" is present, and a dropdown menu is open below it, displaying a list of users: CSLEARNER, DSTEST, JVELE (highlighted), MMGADMIN, KAFKAREADER, MMGDEVADMIN, and NANDHA. A message above the list says "More results available, please filter further."

- Cancel
- Save

Note

The new stage selector in the execution drawer is added for executions from the pipeline canvas with the auto-populated option selected stage in the canvas. Users can now run independent stages from the pipeline canvas and even from the scheduler. The function of the stage remains the same as before as the stages are dynamic in nature. The executed stages are not shown in the execution history, instead the application follows the filtering of nodes to execute or nodes that have been executed at runtime.

Manage Models

A Model has to go through an approval workflow before it can be deployed to production. The following types of users in the system have privileges that restrict the activities, they can do in the model creation and deployment workflow.

Table 11-16 Field or icons for the Model creation and deployment

Field or Icon	Description
Requester	You can create a model. However, to approve and publish, the model must be reviewed by a user with reviewer privileges and approved by a user with approver privileges. NOTE: User Groups must be mapped to the MDLDEPLOY role to access Requester functions.
Reviewer	You can review models. However, to approve and publish, the model must be approved by a user with approver privileges. NOTE: User Groups must be mapped to the MDLREVIEW role to access Reviewer functions.
Approver	You can approve models that are created and reviewed. You can then promote to production and make the model the champion in the production. NOTE: User Groups must be mapped to the MDLAUTH role to access Approver functions.

The following sections in this topic provide details for the cycle of creation of a model, review, approval, and deployment:

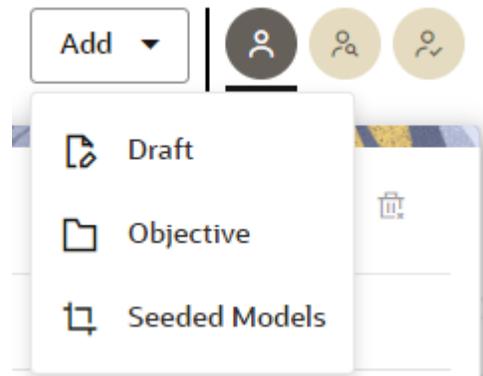
- Create Objective (Folders)
- Create Draft Models
- Publish Models (Scoring)
- View Model Details
- Compare Models
- Understand Model Governance
- Request Model Acceptance
- Review Models and Move to Approve or Reject
- Approve Models and Promote to Production
- Deploy Models in Production and Make it a Global Champion

Create Objective (Folders)

This topic provides information on how to create folders, also known as objectives, within which you can create Models. To create an objective, perform the following steps:

1. Click **Launch Workspace** adjacent to the corresponding workspace to launch the **Dashboard** window with the application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Click **Add** and select **Objective** from the list.

Figure 11-69 Objective Option



4. Enter details in the **Name**, **Description**, and **Tags** fields.

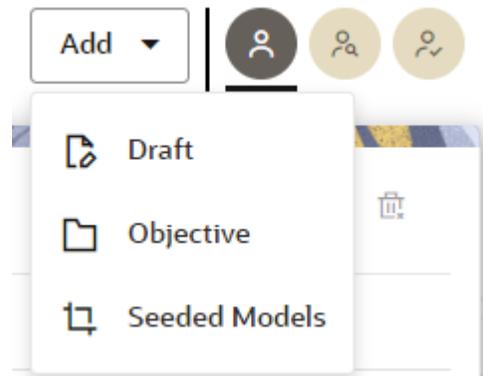
Figure 11-70 Add Objective

5. Click **Save**.

Create Draft Models

After creating an objective, a model draft or notebook can be created where the model development can begin. Perform the following steps to create a draft model:

1. Click **Launch Workspace** adjacent to the corresponding workspace to launch the **Dashboard** window with the application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
This window displays folders that contain models and model records in a table.
3. Click **Add** and then select **Draft** from the list to display the **Add Draft** dialog box.

Figure 11-71 Draft Option

4. **Create New Model** is the default setting in the **Model Details** dialog box. Click the toggle button to select **Import Dump**. You can also create a draft under Objective (Folder) also. Click an Objective to open it.

OR

Drag the toggle button to select **Import Dump**. Import Dump lets you drag and drop an existing file with model data and modify it. To import a model data dump from another model, see the [Import a Workspace Model Data into a New Model](#) section.

To create a new model:

- a. Enter details for Draft **Name** and **Description**.

Figure 11-72 Model Details- Create New ModelA screenshot of a dialog box titled 'Add Draft'. The dialog has a close button (X) in the top right corner. Below the title, there is a toggle switch labeled 'Import dump' which is currently turned off. Below the toggle, there are three text input fields: 'Draft Name' (with a 'Required' label below it), 'Description', and 'Tags'. At the bottom right of the dialog, there are two buttons: 'Cancel' and 'Create'.

- b. Enter a tag in the **Tags** field.

- c. Click **Create**.

Creating Drafts from within an Objective

- a. From the **Model Pipelines** list of objectives, select **Draft** from the **Add** drop-down list.

Figure 11-73 Add Draft from an Objective

Add Draft ×

Objective ANCHOR_TAG + Import dump

Draft Details [Pull from Workspace](#) [Use Template](#)

Draft Name Required

Description

Tags

Add comments Cancel **Create**

- b. Populate the following fields:
 - **Draft Name:** Enter a name for the draft.
 - **Description:** Enter a description for the draft.
 - **Tags:** Add tags for the draft.
 - **Comments:** Add comments for the draft.
- c. Additionally, you can use different methods to create drafts:
 - **Create a draft from an objective**
 - i. Click the **Add in new Objective** icon, adjacent to the **Objective** drop-down list.

Figure 11-74 Objective Details

The screenshot shows a modal window titled "Add Draft" with a close button (X) in the top right corner. Inside the modal, there is a toggle switch for "Import dump" which is currently turned off. Below this is a form titled "Objective Details" with a close button (X) in the top right corner. The form contains the following fields:

- Objective:** A dropdown menu with "ROOT" selected.
- Objective Name:** A text input field with the label "Required" to its right.
- Description:** A text area.
- Tags:** A text area.

Below the "Objective Details" form, there is a section titled "Draft Details" and two buttons: "Pull from Workspace" and "Use Template".

- ii. Perform the following actions to populate the form:
 - **Objective:** Select an objective from the drop-down list.
 - **Objective Name:** Enter a name for the objective.
 - **Description:** Enter a description for the objective.
 - **Tags:** Enter tags for the objective.
 - **Draft Name:** Enter a name for the draft.
 - **Description:** Enter a description for the draft.
 - **Tags:** Add tags for the draft.
 - **Comments:** Add comments for the draft.
 - iii. You can also select the **Pull from Workspace** or **Use Template** options to add the information via a selected workspace or a template.
 - iv. Click **Create**.
- **Pull from Workspace**

Figure 11-75 Pull from Workspace

- **Workspace:** Select a workspace from the drop-down list.
 - **Objective:** This option is enabled when a workspace is selected from the **Workspace** drop-down list. Select an objective from the drop-down list.
 - **Model Pipeline:** This option is enabled when an objective is selected from the **Objective** drop-down list. Select a model pipeline from the drop-down list.
 - **Comments:** Add comments for the draft.
 - **Pull:** Click this option to pull all the selected details from the workspace.
- **Use Template**

Figure 11-76 Use Template

Select a template from the **Templates** drop-down list. Note that any pre-filled values will be overridden with the values from the template that has been selected from the **Templates** drop-down list.

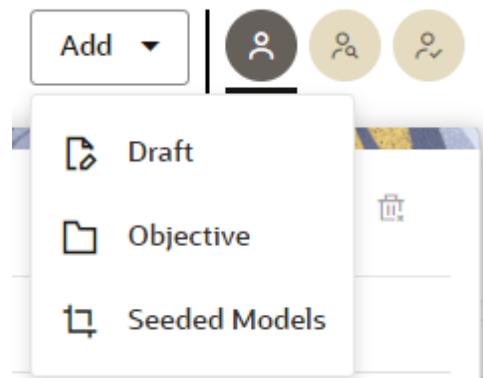
Create Seeded Models

You can seed the models from the external sources which can be imported in the application.

To import the models:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Click **Add** and select **Seeded Models** from the list to display the **Add Draft** dialog box.

Figure 11-77 Add Seeded Models



4. You must add the models in the following installed path location:

<FTPSHARE>/mmg/seeded/models

The added models is displayed in the Seeded Models page.

Figure 11-78 Seeded Models

Seeded Models ×					
<input type="checkbox"/>	Objective Path ◇	Model Name ◇	Model Description	Version ◇	Seeded File Name ◇
<input type="checkbox"/>	Test/NewDraft	NewDraft	Supervised learning analysis of loan application frauds	0	CS_1670477976040_0.zip
<input type="checkbox"/>	Test/NewDraft	NewDraft	Supervised learning analysis of loan application frauds	1	CS_1670477976040_1.zip

Import Seeded Models

5. Select the models which you want to import and click **Import Seeded Models**.
The selected models are imported and displayed in the **Model Pipelines** page.

Model Report

This option allows you to view the Model report and download the same in PDF format.

To view and download Report, follow these steps:

1. Open a Model in Pipeline Designer.
2. Click **Generate Model Report**.
The Model Report window is displayed.
3. Select the Parameters to generate the report.
4. Click **Preview Report**.
The report is displayed based on selected parameters.
5. Click **Export as PDF** to save the report as PDF in local system.

Download a Model

This option allows you to download the Model. To download a model, follow these steps:

1. Open a Model in Pipeline Designer.
2. Click **Download**.
A zip folder is downloaded. This folder contains .cfg and .dsnb files.

Delete

This option is used to delete the Model.

Cloning a Model

You can pick any published model and clone the contents to a new draft in the same objective or clone the content to the current parent draft. The cloned draft can be edited and used further. Audit Trail window also captures the clone information.

To clone the model details, follow these steps:

1. Open a Published Model in Pipeline Designer.
2. Select Clone to new Draft or Reimage parent draft with current.

Pipeline Designer

Clicking on the model navigates you to the Pipeline Designer page.

The following sections are available on the Pipeline Designer window:

- Pipeline Canvas
- Dashboard
- Notebook
- Simulations
- Execution History

- Compare

① Note

1. Models in Production workspace have the 'Dashboard' as the default tab.
2. Drafts or Models in the Sandbox workspace have the 'Pipeline Canvas' as the default tab.

Pipeline Canvas

You can perform following functions on Pipeline Canvas:

- Creating a Pipeline
- Creating Script Template
- Viewing a Pipeline
- Using Link Connector Nodes
- Execution of Pipeline
- Publishing a Pipeline

① Note

Users are now able to view the sub-stages from the canvas with the help of the new stage dropdown.

Create Paragraphs using Pipeline Designer

After creating the Models in the Workspace, create Paragraphs using the **Pipeline Designer** window.

Pipeline Designer enables you to design the paragraph using widgets (graphical representation) instead of using python codes. In addition, if you add new paragraphs in Data Studio, the added paragraphs are displayed in the widget format on Pipeline. Similarly, if you create a Notebook using Pipeline Designer, it can be opened for editing in Data Studio using Studio Notebook option.

① Note

When you open the notebooks from the UI, the attach workspace call will be made from mmg service and proper workspace will get attached. If the Studio is opened outside of mmg, then the attach_workspace command has to be used.

This helps the Financial Institutes and Banks in following ways:

- Visualization of the data (for example, based on data tasks)
- View the dependency
- Modify the flow of execution or execution order
- Easy for Auditing purpose

You can execute the flow based on requirement. For example, if you have created one flow and want to execute a flow of training paragraphs out of that and other flow as experimental way, then you can modify using the Training and Experimentation link types. One flow can be break into 2-3 flows for execution purpose.

When a draft is edited using the Pipeline Designer/Data Studio, and published, then a new version of published model is displayed in Model Summary page.

① Note

When you add a new paragraph from studio and opened the same in pipeline designer, it gets linked to multiple paragraphs. For example, you have paragraphs P1, P2 and P3 in the same order in a Notebook. It shows in Pipeline Designer canvas as P1 > P3 > P2.

If you add two new paragraphs P1a and P1b in the Notebook after the P1 and open the canvas, this gets reflected as the following:

1. P1 -> P1a -> P1b -> P2
2. P1 -> P3 -> P2

After opening a Model in the Pipeline Designer, following options are displayed:

- Generate Model Report
- Download
- Delete
- Clone Model

After opening a Draft in the Pipeline Designer, following options are displayed:

- Generate Model Report
- Download
- Delete
- Publish
- Script Template list

Create a Pipeline

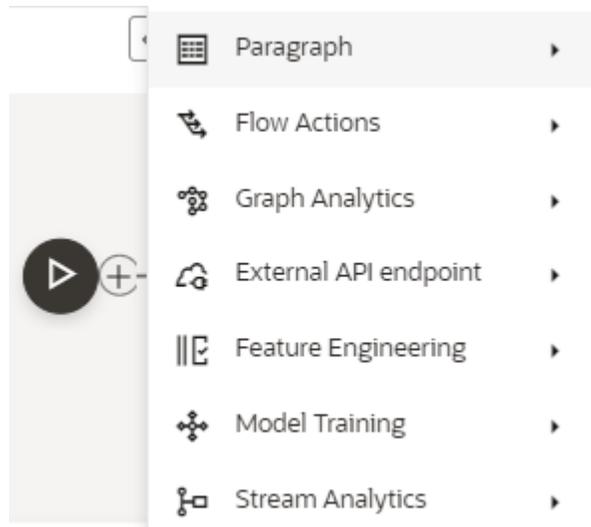
To create a paragraph using pipeline:

1. Navigate to the **Pipeline Designer** page.

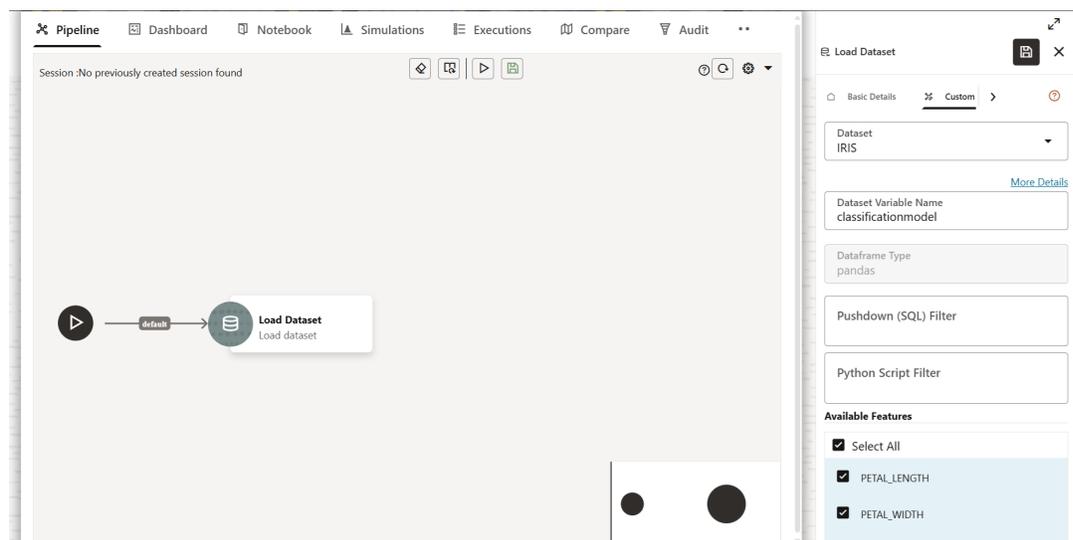
The Start widget is displayed by default in the canvas screen.

2. Hover over on Start widget and new nodes can be added using the add button in the Pipeline Canvas.

This will open a menu which contains all the widgets sectioned into different categories. The spectra diagram builder is based on row and column positioning.

Figure 11-79 Create a Pipeline menu

Push Down Filtering (SQL Filters) and Python Script Filtering is available for the **Load Dataset** widget in the pipeline.

Figure 11-80 Pushdown (SQL) Filter and Python Script Filter**Examples:****Pushdown (SQL) Filter:**

- column1 > 100 and column2 < 1000

Python Script Filtering:

```
df_out = df_prev[df_prev['column'] > 100]
```

Note

Click on each menu to select the widgets of the required types.

You cannot edit or delete this widget. Whenever a new draft is created (not by importing dump files), the default paragraph created is converted into a start widget. The visibility of code/result/title in notebook of this node will be kept to invisible.

Whenever the notebook is opened, init script execution including workspace attach will happen in this start node.

Title of the node is called 'Start widget'. Publish /download and import /promote to production of model with start widget will keep this widget. Publish from canvas will keep the start widget in the published model. When you publish model from model summary, you can explicitly select the start widget paragraph from the list of paragraphs. After this only, start widget will appear in the published model.

3. Click on the node to add the basic details.

The Basic Details page is displayed.

Figure 11-81 Basic Details for Paragraph

The screenshot shows the 'Basic Details' page for a Paragraph widget. At the top, there are two tabs: 'Basic Details' (selected) and 'Script'. Below the tabs, there are four main sections:

- Activity Name:** PARAGRAPH8b64bd7e-31aa-4db3-905f-d56f9a36c518
- Description:** (Empty text field)
- Task Type:** Default (Dropdown menu)
- Track Output:** (Toggle switch, currently off)

Below these fields is a code editor containing the following Java code:

```
%pgx-java
import java.io.*
import java.util.concurrent.TimeUnit
import org.apache.commons.io.*
import oracle.pgx.common.*
import oracle.pgx.common.mutations.*
import oracle.pgx.common.types.*
import oracle.pgx.api.*
import oracle.pgx.api.admin.*
import oracle.pgx.config.*
import oracle.pgx.api.filter.*
import oracle.pgx.api.PgxGraph.SortOrder
import oracle.pgx.api.PgxGraph.Degree
import oracle.pgx.api.PgxGraph.Mode
import oracle.pgx.api.PgxGraph.SelfEdges
import oracle.pgx.api.PgxGraph.MultiEdges
import oracle.pgx.api.PgxGraph.TrivialVertices
session
instance
analyst
// can define new classes
public class Functions {
    public static double haversine(double lat1, double lon1, double lat2, double lon2) {
        double delta_lon = (lon2 - lon1) * Math.PI / 180;
        double delta_lat = (lat2 - lat1) * Math.PI / 180;
        double a = Math.pow(Math.sin(delta_lat / 2), 2) + Math.cos(lat1 * Math.PI / 180) * Mat
        double c = 2 * Math.asin(Math.sqrt(a));
        double r = 6371; // Radius of the Earth in kilometers. Use 3956 for miles
        return c * r;
    }
}
```

4. Provide details as described in the following table:

Table 11-17 Adding Details

Field	Description
Activity Name	Enter the Activity Name
Description	Enter the description of Activity
Task Type	Select the task type. For example, Model Training, Data Analysis and so on. You can also search the task type.
Track Output	If this option is selected for any paragraph, then during the model comparison the output details are displayed. Keep the Track Output to ON in case you want to execute the paragraph and view the result from the Dashboard tab.
Script	Shows the script. You can edit the script in this screen or in the Script tab. This script can also be saved as the Script Template.

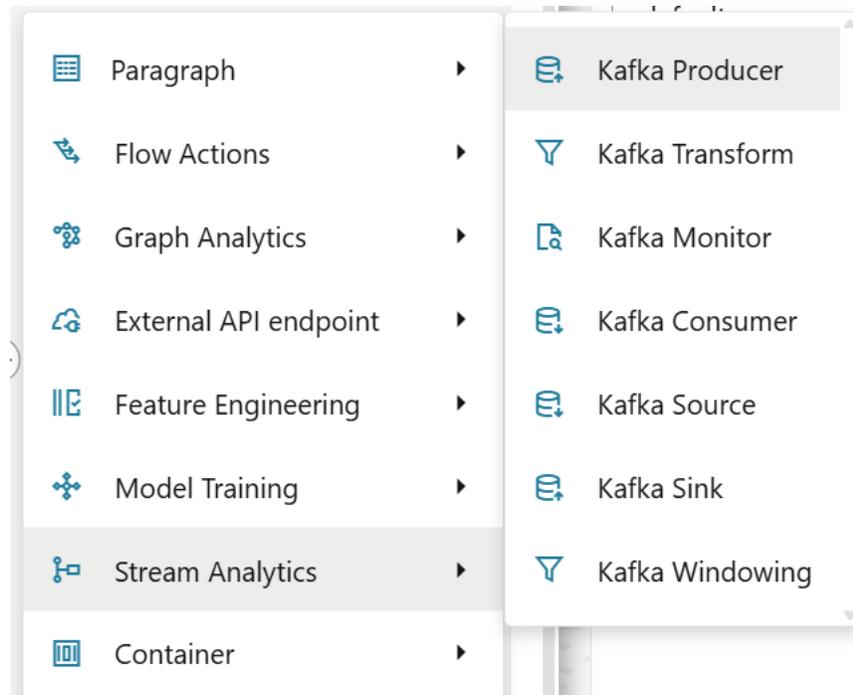
Note

If is a conditional node, which behaves based on the execution of the user. It has a script which evaluates "**True**" or "**False**" and based on the value, it chooses either of them. Additionally, we can also add a new node that comes with a default link. The default path will execute irrespective of the result.

Kafka Related Pipelines

Kafka widgets are available under the Stream Analytics. You can create kafka pipelines by using different kafka widgets like **Kafka Source**, **Kafka Sink**, **Kafka Consumer**, **Kafka Producer**, **Kafka Monitor** and a newly added widget called **Kafka Windowing**.

The following image illustrates the Kafka related pipelines:

Figure 11-82 Kafka related pipelines

The **Kafka Consumer**, **Kafka Producer**, **Kafka Transform** and **Kafka Monitor** widgets can be utilized in a pipeline in the following sequence

1. **Kafka Consumer**
2. **Kafka Transform**
3. **Kafka Producer**
4. **Kafka Monitor**

Create a Pipeline by Using the Kafka Widgets

Note

The **Kafka Input** and **Kafka Output** topics that are created under the **Kafka Topics** screen must be mapped to the workspace for user to select it when creating pipelines.

Kafka Source

When you click the **Kafka Source** widget, the following mandatory parameters must be provided:

Figure 11-83 Kafka Source

The screenshot shows the 'Kafka Source' configuration window with the 'Custom Parameters' tab selected. The parameters are as follows:

- Input Topic**: A dropdown menu, marked as Required.
- Auto Offset Reset**: A dropdown menu, marked as Required.
- Enable Auto Commit**: A dropdown menu, marked as Required.
- Consumer Timeout(ms)**: A text input field, marked as Required.
- Group ID**: A text input field.
- Transformation Script**: A text input field.

Table 11-18 Mandatory Parameters

Parameter	Description
Input Topic	Select the input topic.
Auto Offset Reset	Select either of the two options: <ul style="list-style-type: none"> Earliest- This displays the earliest starting data Latest- This displays the latest data.
Enable auto commit	Select either <i>True</i> or <i>False</i> .
Consumer Timeout(ms)	Provide timeout time in milli seconds.
Group ID	Provide any group ID (any data).
Transformation script	Provide a script. For example, the following is the script of how data will be printed to the file: <pre>import pandas as pd data_out= { "Msg": data_prev} pd.DataFrame(data_out).to_csv("/scratch/ofsaapp/kafka/output/data_out.csv", sep='\t')</pre>

In order to utilize the **Kafka Source** in a pipeline, an additional python paragraph must be added after the **Kafka Source** widget. The following is an example of the python script:

```

%python

import pandas as pd

# Sample data with Name and Gender columns

data = {
  'start': [1,1],
  'end':[11,10],
  'avg_amount':[100,124],
  'count':[12,11]
}

# Create the DataFrame

df = pd.DataFrame(data)

print (df)

```

Kafka Sink

When you click the **Kafka Sink** widget, the following mandatory parameters must be provided:

Figure 11-84 Kafka Sink

Table 11-19 Mandatory Parameters

Parameter	Description
Output Topic	Select the output topic.
Input Dataframe	Provide a value in this parameter.

In order to utilize the **Kafka Sink** in a pipeline, an additional python paragraph must be added first and then the Kafka Sink widget must be added.

Kafka Consumer

When you click the **Kafka Consumer** widget, the following mandatory parameters must be provided:

Figure 11-85 Kafka Consumer

The screenshot shows the 'Kafka Consumer' widget configuration window. The 'Custom Parameters' tab is selected. The following parameters are listed as required:

- Input Topic**: A dropdown menu. Required. [More Details](#)
- Auto Offset Reset**: A dropdown menu. Required.
- Enable Auto Commit**: A dropdown menu. Required.
- Consumer Timeout(ms)**: A text input field. Required.
- Group ID**: A text input field. Required.

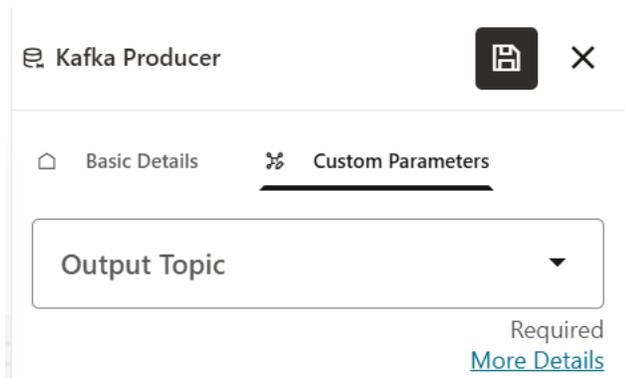
Table 11-20 Mandatory Parameters

Parameter	Description
Input Topic	Select the input topic.
Auto Offset Reset	Select either of the two options: <ul style="list-style-type: none"> Earliest- This displays the earliest starting data Latest- This displays the latest data.
Enable auto commit	Select either <i>True</i> or <i>False</i> .
Consumer Timeout(ms)	Provide timeout time in milli seconds.
Group ID	Provide any group ID (any data).
Transformation script	Provide a script. For example, the following is the script of how data will be printed to the file: <pre>import pandas as pd data_out= { "Msg": data_prev} pd.DataFrame(data_out).to_csv("/scratch/ofsaapp/ kafka/output/data_out.csv", sep='\t')</pre>

Kafka Producer

When you click the **Kafka Producer** widget, the following mandatory parameters must be provided:

Figure 11-86 Kafka Producer



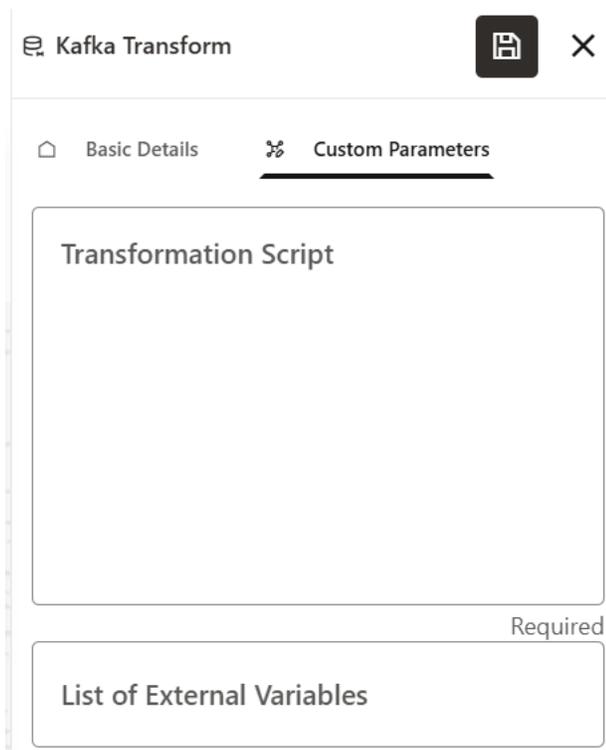
The screenshot shows the configuration window for the 'Kafka Producer' widget. At the top, there is a title bar with a close button (X) and a save icon. Below the title bar, there are two tabs: 'Basic Details' and 'Custom Parameters', with 'Custom Parameters' being the active tab. The main content area contains a text input field labeled 'Output Topic' with a dropdown arrow on the right. Below this field, the text 'Required' is displayed, followed by a blue link labeled 'More Details'.

- **Output Topic:** Provide the output topic.

Kafka Transform

When you click the **Kafka Transform** widget, the following mandatory parameters must be provided:

Figure 11-87 Kafka Transform



The screenshot shows the configuration window for the 'Kafka Transform' widget. At the top, there is a title bar with a close button (X) and a save icon. Below the title bar, there are two tabs: 'Basic Details' and 'Custom Parameters', with 'Custom Parameters' being the active tab. The main content area contains a large text input field labeled 'Transformation Script'. Below this field, the text 'Required' is displayed. Below the 'Transformation Script' field, there is another text input field labeled 'List of External Variables'.

- **Transformation script:** Provide a script. For example, the following is the script on how data will be printed to the file:

```
import pandas as pd

data_out= { "Msg": data_prev}

pd.DataFrame(data_out).to_csv("/scratch/ofsaapp/kafka/output/data_out.csv",
sep='\t')
```

Kafka Monitor

When you click the **Kafka Monitor** widget, the following mandatory parameters must be provided:

Figure 11-88 Kafka Monitor

Table 11-21 Mandatory Parameters

Parameter	Description
Time duration	Provide the time in milliseconds.
Output partition to monitor	Enter any numeric value.
Output variable name	Provide any name that will be the output

Kafka Windowing

The windowing widget integrates with the existing kafka producer and consumer widgets in a PyFlink pipeline. This widget enables you to perform windowed computations, tumbling or sliding, with various configurations on event streams flowing from Kafka.

It ensures that the windowing widget accepts inputs as specified, properly integrates within the producer-consumer pipelines, correctly applies windowing logic, and robustly handles legal/illegal configurations and edge cases.

The following are the inputs of the kafka window:

Figure 11-89 Kafka Windowing

Kafka Windowing

Basic Details Custom Parameters

Window Assigner Required

Window Size (sec) Required

Window Slide (sec)

Window Type Required

Watermark Column

Allowed Lateness (sec) Required

Key By Column Required

Window Function Body

Table 11-22 Kafka Windowing Fields

Field	Type	Values / Rules
Window Assigner	SELECT	Tumbling and sliding
Window Size (sec)	TEXT	Positive integer
Window Slide (sec)	TEXT	Positive integer (sliding only, optional)
Window Type	SELECT	event_time, processing_time
Watermark Column	TEXT	Optional; for event_time windowing
Allowed Lateness (sec)	TEXT	Integer; max allowed lateness for event_time
Key By Column	TEXT	Column name for grouping
Window Function Body	LONGTEXT	PyFlink aggregate function body (Python code)

Table 11-22 (Cont.) Kafka Windowing Fields

Field	Type	Values / Rules
Note:		
		<ul style="list-style-type: none"> • – Required fields must not be blank. – For <i>tumbling</i>, <i>Window Slide (sec)</i> must be ignored/hidden/optional. – For <i>sliding</i>, <i>Window Slide (sec)</i> is required. – For <i>event_time</i>, <i>Watermark Column</i> and <i>Allowed Lateness (sec)</i> are required and validated. – Negative/zero values or non-integer inputs are rejected with an error message.

The following is an example configuration for the kafka windowing feature:

```

from pyflink.common import Row

def sum_amount_reduce(a, b):
    """
    Reduce function to sum the Amount field between two Row objects,
    keeping other fields from the first Row and using the latest TimeStamp.
    Assumes Row schema:
    (Account_Id, Txn_Origin, Amount, Trxn_Id, Txn_Channel, TimeStamp)
    """
    return Row(
        a[0], # Account_Id
        a[1], # Txn_Origin
        a[2] + b[2], # Amount
        a[3], # Trxn_Id
        a[4], # Txn_Channel
        max(a[5], b[5]) # TimeStamp
    )

```

Note

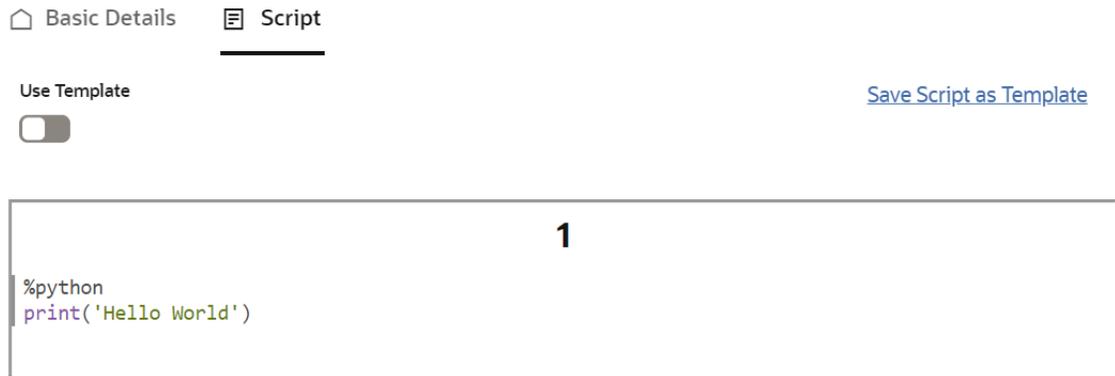
The following features will be implemented in a future release:

- – **Tooltips:** Informative tooltips for each input field will be provided for user guidance.
- **Conditional Field Visibility:** Display the *Window Slide (sec)* field only when the window assigner is set to *sliding*.
 - * * *Watermark Column* will be displayed only when the window type is *event_time*.

Create Script Template

Navigate to Script tab to manage scripts. These scripts can be called for paragraph. You can also create the Script template by clicking on the More options icon in the Pipeline Designer.

Figure 11-90 Script template



To add a script template:

1. In the **Script** tab, Click **Save Script as Template** link.

Figure 11-91 Script tab

The screenshot shows a form for creating a script template. It has four main sections:

- Script Name:** A text input field with the placeholder text 'Script name should be unique' and a 'Required' label to its right.
- Description:** A text input field with the placeholder text 'Description'.
- Script Type:** A dropdown menu with 'Default' selected.
- Script Content:** A text area containing the Python code: `%python print('Hello World')`.

 At the bottom right of the form are two buttons: 'Cancel' and 'Save'.

2. Enter the following details:

Table 11-23 Creating Script Template

Field	Description
Script Name	Enter the Script Name
Description	Enter the description of Script

Table 11-23 (Cont.) Creating Script Template

Field	Description
Task Type	Select the task type. For example, Model Training, Data Analysis and so on. You can also search the task type.
Script Content	Enter Python script

3. Click **Save**.

A node is created.

You can perform the following functions on Pipeline Designer window:

- **Parameter Sets:** Allows you to view, edit, clone, and delete the Parameter Set. Use the Clone option to duplicate the parameter set with different values and name.
- **Publish:** Allows publishing the pipeline. This option is displayed for the Drafts.
- **Deploy:** Allows you to deploy the model. This option is displayed for the published models.
- **More Actions icon:**
 - **Generate Model Report:** To generate model report
 - **Download:** Use Download to download the current working version in opened in canvas
 - **Script Template list:** Allows you to add, edit, view, and delete the script templates.
 - **Delete:** Allows to delete the current working version. If this is first draft of Model, it will delete all the dependent published version in the Sandbox. If the Model is not first version, then it will delete only the current working version.

The following options are displayed only for the published models.

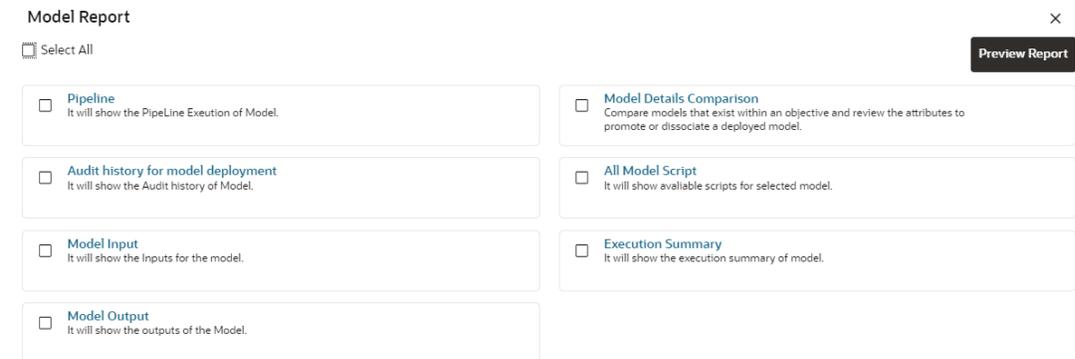
- **Clone to new draft:** Allows to create a new draft with the same pipeline.
- **Overwrite existing draft:** Allows to overwrite the existing draft with the current published model.

Model Report

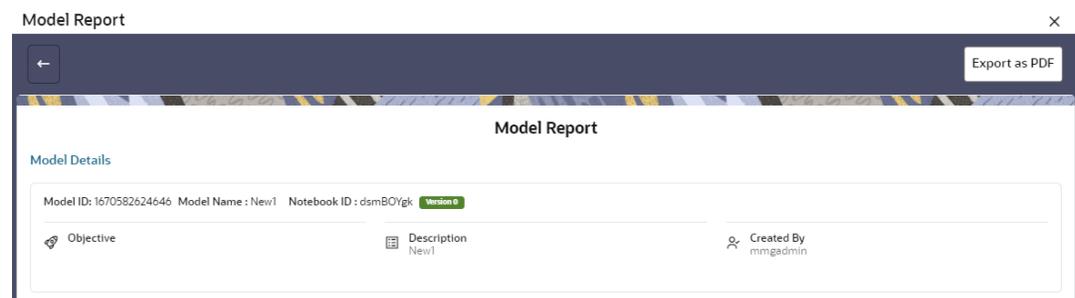
This option allows you to view the Model Report and download the same in PDF format.

To view and download report:

1. Open a Model in Pipeline Designer.
2. Click **Generate Model Report**.
The Model Report window is displayed.

Figure 11-92 Model Report window

3. Select the Parameters to generate the report.
4. Click **Preview Report**.
The report is displayed based on selected parameters.

Figure 11-93 Model Report

5. Click **Export as PDF** to save the report as PDF in local system.

Download a Model

This option allows you to download the Model.

To download a model:

1. Open a Model in Pipeline Designer.
2. Click **Download**.
A zip folder is downloaded. This folder contains .cfg and .dsnb files.

Delete

This option is used to delete the Model.

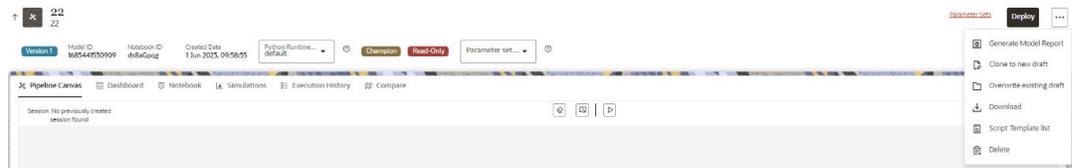
Cloning a Model

You can pick any published model and clone the contents to a new draft in the same objective or clone the content to the current parent draft. The cloned draft can be edited and used further. The Audit Trail window also captures the cloning information.

To clone the model:

1. Open a Published Model in Pipeline Designer.

Figure 11-94 Published Model in Pipeline Designer



2. Click **More Actions** icon and select **Clone to new draft** or **Overwrite existing draft** option.

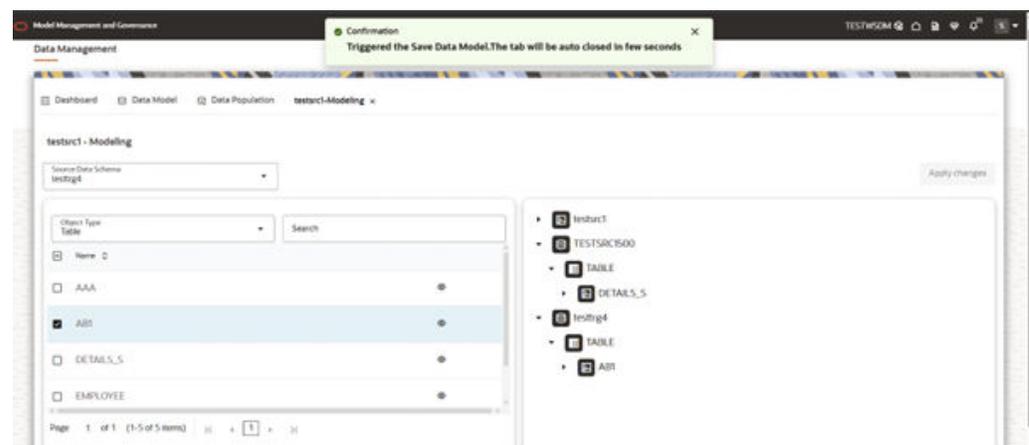
You can perform the following functions on Pipeline Canvas window:

- **Clear Execution Results:** Clears the execution details.
- **Invalidate session:** Deletes the previous session details.
- **Execute:** Allows you to execute the pipeline.
- **Save Now:** Allows saving the pipeline.
- **Open Notebook in session:** Allows you to open the notebook in canvas. This is displayed after the pipeline is executed.
- **Execute Notebook in another session:** Allows you to provide different data for the same notebook execution. This option is only displayed when execution is in progress.
- **Stop Execution:** Allows you to stop the execution which is in progress. This option is only displayed when execution is in progress.

Whenever user executes a batch, a user session is created. This execution time can be less or more for any execution. Sometimes, user doesn't want to wait for execution to complete and navigate away from the Pipeline Canvas page.

For example, if user does not want to execute all the paragraphs and want to execute only Paragraph 1 and Paragraph 2. Paragraph 1 takes 15 minutes time for execution and Paragraph 2 takes 10 minutes for execution. Paragraph 2 execution also wants to use the execution of Paragraph 1. In this case, user can navigate away from the page. A confirmation message is displayed to close the current session. Here, this session time is configurable.

Figure 11-95 Confirmation message



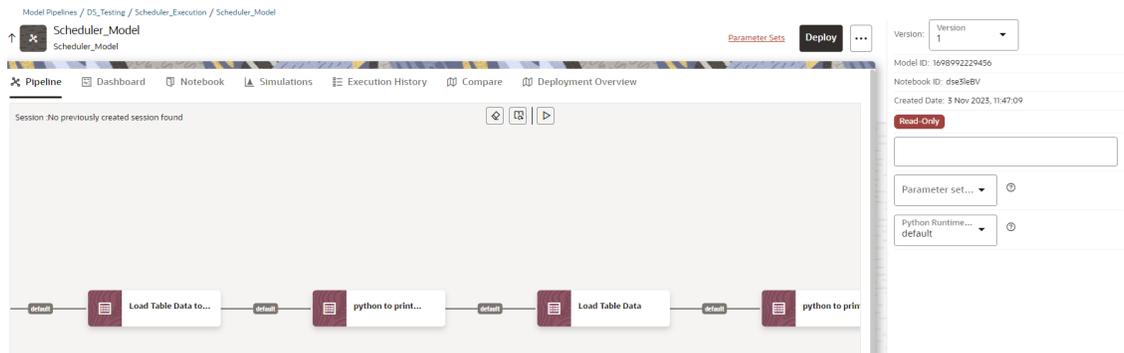
- If user clicks **Yes**, then execution thread will be closed for this given session time.
- If user clicks **No**, then execution thread will be valid for this given kill time and run in the background.

You can configure these values in Application.xml file. For more information, see the *MMG Installation Guide*.

Add a Version, Parameter set, and Python Runtime parameters

The Pipeline canvas window allows you to add a Version, Parameter set, and Python Runtime parameters.

Figure 11-96 Pipeline canvas window



- **Version:** Select the version for the pipeline.
- **Parameter Set:** Select the required parameter set. The selected parameter set will be promoted along with the model pipeline at the time of deployment and you can update this dependency based on your requirements.
- **Python Runtime Parameters:** Select the required Python Runtime parameter. The selected Python runtime parameter will be used during all the executions and you can update this dependency based on your requirements.

View a Pipeline

The Pipeline canvas window allows you to view the Pipeline using the following options:

- **Auto-align:** Arrange all the widgets in vertical order. After saving, the reverting option will not work.
- **Revert-align:** Revert all the widgets if they are Auto-aligned.
- **Refresh:** Refresh the pipeline canvas.

Execute a Notebook

The Execute icon on Pipeline Canvas allows us to execute the notebook. The following link types are available in the Pipeline Designer:

- Default
- Scoring
- Training
- Experimentation

Note

When a model gets published from Model summary page, the Link types configured in Pipeline Designer are set to Default link type.

To execute the notebook:

1. Click **Execute** to view **Execute Pipeline** window.

Figure 11-97 Execute Pipeline

2. Execution parameters are the parameters defined in the notebook required for execution. Select the flow, which you want to execute Scoring, Training, and Experimentation. It displays all the keys defined for all the paragraphs in the notebook with a placeholder for providing the values.
3. Enter the execution Key and Value.

You can also use Runtime parameters for execution. This runtime parameter must be defined in Notebook. If this is defined, you can enter execution value during the process execution.

For more information, see [Create Paragraphs in Model Studio Notebooks](#) section.

The System Parameters window also shows the execution ID, execution Date, and execution Batch. These are required for executing all the paragraphs along with other parameters. It also shows from where the parameter comes from as a subscript.

- **Parameter Sets:** These are the set of parameters with a specific value required for an instance of execution. It consists Key and Value. You can save the parameters set that can be used for one execution instance and reuse it for the next execution. It consists of parameters with a specific value to each parameter. Parameters containing no value will not be taken. Each set is identified with a unique code for each objective. While saving the parameter, you have to provide a code for identifying the name and description which is not mandatory. You can save Key Value parameter set using the Save Parameter Set option. To Save Parameter Set, enter the Threshold Value and Description in the Parameter Set window.
- **Selecting Parameter Set:** These saved Parameter Set can be selected during the execution. It will replace the values of the parameters from the chosen Parameter Set. To select the Parameter Set, follow these steps:

- Click “Open from saved Parameter Set”. The Threshold Code window is displayed.
 - Select the Parameter Set from the available list. You can select multiple Parameter Set in the same execution instance. In that case, if there are any common keys, value will be replaced with that from the latest Parameter Set selected.
4. You can add new parameters using **Save parameter set** option.

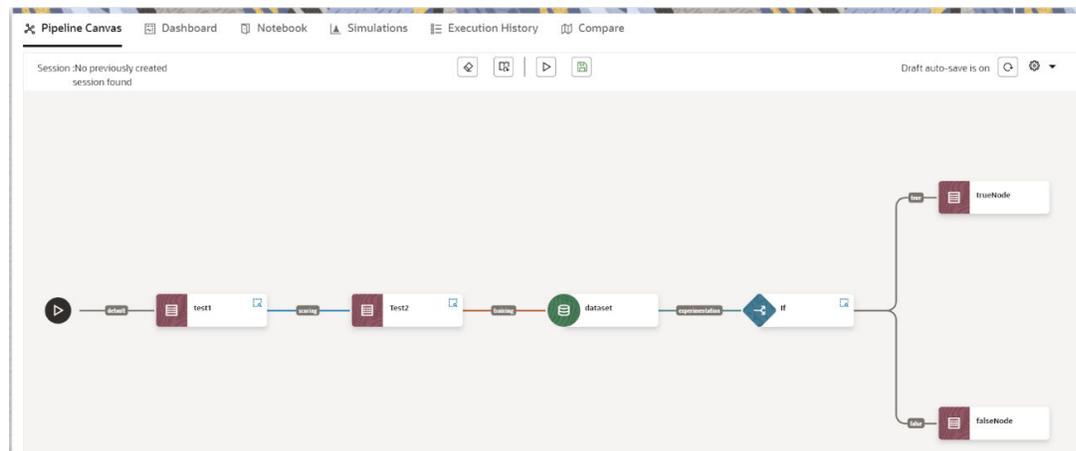
Note

If the parameter is not defined in the notebook, it will not be used for the execution. In case of multi select, if there are common parameters among the chosen scenarios, it will take the value based on the order of selection. that is first chosen scenario parameter will be taken.

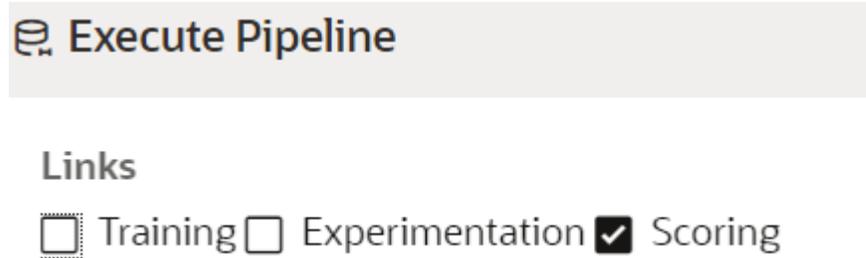
5. But if open from saved Parameter Set again (not on single go), then already added will get replaced by the newly added (same as what existed).
6. Execution is performed based on selected link types. It filters out all the not required/ unused parameters. And all the unused parameters for the current execution are displayed with a warning. To view the only required parameters, click **Show only required** link.
7. Click **Reset** to reset the entered data.
8. Click **Delete** to delete the entered Key and Value.

For example, refer to below figure.

Figure 11-98 Example of Pipeline Canvas



- Here, if you want to execute this Notebook for scoring purpose, then the flow will be executed till Test 2. To perform this, Click Execute Notebook and select Links as Scoring.

Figure 11-99 Execute Pipeline

- Similarly, if you want to execute this Notebook for training purpose, then the flow will be executed till dataset with default paragraphs. To perform this, Click Execute Notebook and select Links as Training.

Publish a Pipeline

To publish the pipeline:

1. Click **Publish** to view **Publish Pipeline** window.

Figure 11-100 Publish Pipeline

Publish Pipeline [X]

Model Name
Retail PD Model using Logistic Regression

Technique
logistic regression

Run Version
2

Description
PDMModel Logistic Regression

Tags
basel × regression ×

Attach parameter set [?]

Parameter set
TuningSet1

Associate python runtime [?]

Python Runtime
demo_home

Add comments
publishing with updated widgets

Cancel Publish

2. Enter the details as shown in the following table.

Table 11-24 Field and its description for the Publish pop up window

Field or Icon	Description
Model Name	The field displays the name of the Model.
Model Description	The field displays the description for the Model. Enter or modify the description if required.
Technique	Enter the registered technique to use.
Run Version	Select a run version.
Tags	Enter one or more tags to associate with the Model Pipeline. The tags help to categorize your pipeline. These tags are permanent and cannot be modified after publishing.

Table 11-24 (Cont.) Field and its description for the Publish pop up window

Field or Icon	Description
Attach Parameter set	<p>If this option is enabled:</p> <ul style="list-style-type: none"> Selected parameter set will be associated with the published model. Best performing parameter set or prime reference of the objective is recommended to be associated with the model. This set is further used for promoting to production at the time of deployment. Users can update this dependency from model details screen.
Associate python runtime	<p>If this option is enabled:</p> <ul style="list-style-type: none"> Selected python run time will be associated with the published model. This python run time will be further used for promoting the model to production at the time of deployment. Users can update this dependency from model details screen.

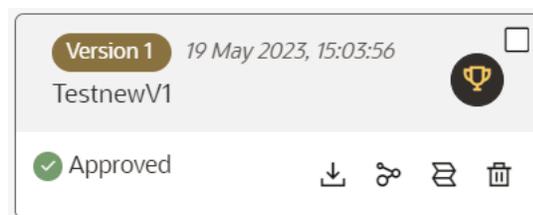
3. Click **Publish**.

View Model Details

You can view model information for deployed models, models that require approval, and so on.

To view model details:

- Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
- Click **Model Pipelines** to display the **Model Pipelines** window. The window displays folders that contain models and model records in a table.
 - The icon indicates that Model 1 is the champion.
- Hover over the model records to view the various icons.

Figure 11-101 Mouse over the Model

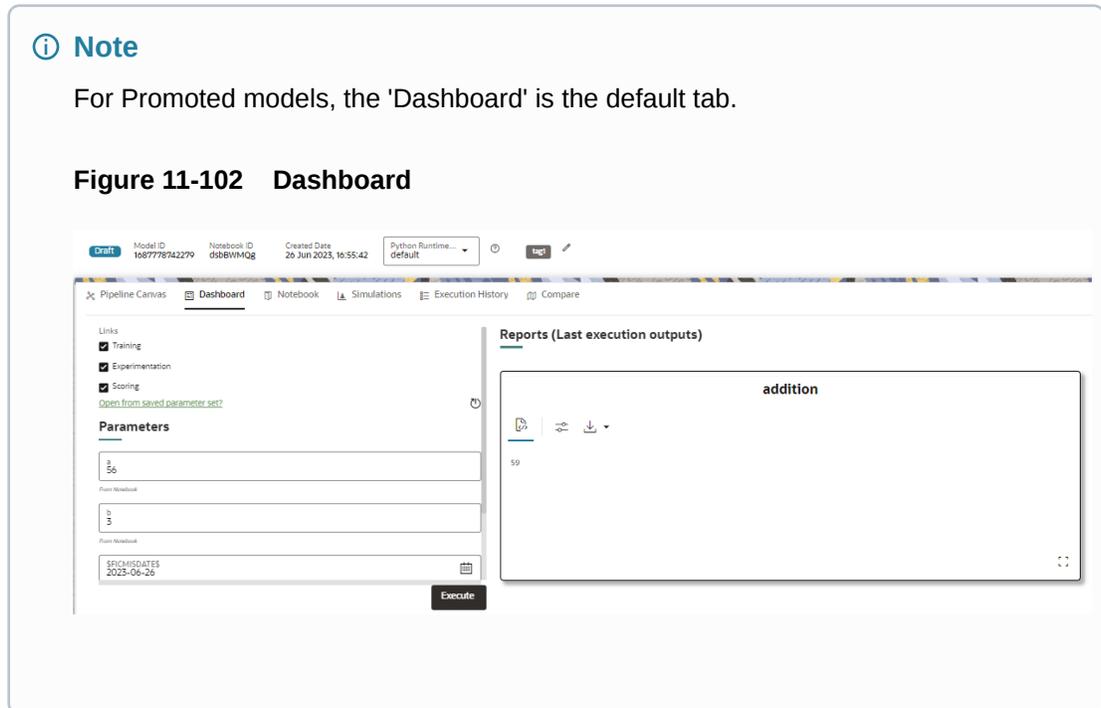
The icon actions are listed in the following:

- Download
- Open in Pipeline Designer. See the Create Paragraphs using Pipeline Designer section for more details.
- Scope Detail. See the Scope Detail section for more details.

d. Delete Model.

Dashboard

This section of Pipeline Designer allows users to execute the Models and also shows the execution output of Model if the widgets are saved with Track output option enabled.



Note

There is no Cancel button in Settings tab of reports in the Dashboard tab. You can press 'Escape' key to close the Settings.

Notebook

Note

The Username is case sensitive. Ensure you use the correct case for the username when accessing and executing the notebook.

Navigate to **Notebook** page to view the paragraphs. You can run, Restart session, edit, add, export the notebook, and so on.

Figure 11-103 Notebook page



Note

The following features on Notebook are not supported in the current release.

- Cloning the Notebook
- Sharing the Notebook
- Versioning the Notebook
- Modifying the Notebook
- Deleting the Notebook
- Attaching credentials to the Notebook
- Entering dependency modes to the Notebook

Steps to enable two or more users to access the same Notebook with distinct sets of permissions

To enable two or more users to access the same notebook with distinct sets of permissions, follow these steps:

1. **Create a New Role in Data Studio**
Log in to MMG with a user account that has DSUSRGRP (Data Studio User) access:
 - Navigate to Data Studio Options > Permissions and switch to the Roles tab. # Click Create Entity and create a new role (For example, DSREADONLY). Assign the desired permissions, such as:
 - View Code
 - View Notebook
 - View Sessions
 - View Result
 - View Paragraph for Read-Only access to Data Studio Notebooks.
2. **Create a Corresponding User Group.**
Navigate to your Identity Management page and create a new user group with the same name as the role created in Data Studio. For example, DSREADONLY.
3. **Update Application Configuration.**
To validate the new user group, update the `application.yml` file:
 - Navigate to `mmg-studio/conf/application.yml`. #
 - Under `studio-server.security`, append the newly created role to the `validRoles` field, separated by commas (For example, `validRoles[*]: "MDLUSR,MDLREV,MDLAPPR,MDLBATCHUSR,DSUSRGRP,DSREDACTGRP,MDLVIEW,DSREADONLY"`).
4. **Restart Data Studio.**
Restart Data Studio using the following commands:
 - # `./shutdown.sh`
 - # `./startup.sh`
5. **Map the New User Group to a User.**
Navigate back to the Identity Management screen and map the newly created user group (DSREADONLY) to the user(s) you want to grant these permissions.

Markers for track output and link types from Notebook and syncing up with Canvas

Markers for track output and link types from notebook and syncing up with canvas Using specific key words like "##track output##", there should be an option to provide these key words from the notebook paragraph and sync up with the pipeline. User will not have to use canvas interface here. This has to be done for output tracking and link types to determine execution paths (training,scoring,experimentation).

These are only applicable for draft and 'PARAGRAPH type nodes:

- `##track_output##` - For paragraph output tracking
- `##training##` - For training nodes
- `##scoring##` - For scoring nodes `##Experimentation`
- `##` - For Experimentation nodes

If the marker "`##track_output##`" is added in the Notebook paragraph, the canvas will automatically be synced up with this and corresponding node output will be tracked. If the marker is removed, the output tracking will be set to off.

On execution, if any of the nodes are associated with a path, the paths added on the links will not be considered for the execution. There is also an option to choose these path types from the node basic details window.

Access Module for File/Folder Operations

The OFS MMG Access feature provides utility functions where code can be used with OFS MMG workspaces and the file system within the OFS MMG Studio to perform file/folder related operations.

It is assumed that the module has already been imported by using one of the following codes:

```
from mmg.access import
```

or

```
import mmg.access as mac
```

Note

The workspace attachment step must be performed before using any of the following API mentioned in this section. The current working directory will always be the path inside the `FTPSHARE` of the attached workspace. For example, `<FTPSHARE_PATH>/mmg/<WS>`

If the source and the destination starts with `/`, then it is assumed to be an absolute path (as is the case with `os` and `shutil`). Hence, to maintain the compatibility, the same is the case with the OFS MMG access feature module APIs

Table 11-25 List of APIs

Name	Description	Usage Example	Note
getcwd	Returns the current working directory path within MMG FTPSHARE for the active workspace.	getcwd()	The <code>getcwd()</code> function returns the path inside the attached workspace that is within FTPSHARE. It does not reflect the actual <code>cwd</code> path that is returned from the <code>os.getcwd()</code> function. Hence, using <code>os.chdir()</code> will not affect the <code>getcwd()</code> function implemented from the OFS MMG access module library.
move	Moves a file or directory to another location.	<code>move(src='/source/path', dst='/destination/path')</code>	
copy	Copies a file with metadata from one location to another.	<code>copy(src='/source/path', dst='/destination/path')</code>	
copyfile	Copies a file without its metadata.	<code>copyfile(src='/source/file', dst='/destination/file')</code>	
copytree	Recursively copies a directory tree to the destination.	<code>copytree(src='/source/dir', dst='/destination/dir', dirs_exist_ok=True)</code>	
listdir	Lists all files in the specified directory.	<code>listdir(path='/your/dir')</code>	
remove	Removes a single file from the specified path.	<code>remove(path='/file/to/remove')</code>	
rename	Renames a file within the current workspace.	<code>rename(src='/old/filename', dst='/new/filename')</code>	

Create Paragraphs in Model Studio Notebooks

When logged in to the OFS MMG application, the operating system and similar libraries can be used to view/edit/delete the files that are outside the FTPSHARE or workspace bounded path.

1. Log in to the application.
2. Create a notebook.
3. Create script by using the operating system library.
4. Use `os.remove()`, `os.listdir()` by providing the path that is outside the scope of FTPSHARE.
5. The above operations will be run successfully.

Create Paragraphs

The paragraph should be added after the start widget paragraph. Navigate to Notebook page to create the paragraphs.

The following types of paragraph creation are supported for model creation:

- Python Paragraph
- Shell in Python Paragraph
- Conda Paragraph
- PGQL Paragraph
- PGX - Java Paragraph
- Paragraph
- PGX - Python Paragraph
- PGX – Algorithm Paragraph
- PGX - PySpark Paragraph
- MySQL Paragraph
- Markdown Paragraph

To create Paragraphs in the Model Studio Notebooks:

1. Open the draft in Pipeline Designer and click **Notebook** tab.
2. Hover your mouse above or below a paragraph to display the interpreter toolbar.

Figure 11-104 Interpreter toolbar



3. Click an icon in the toolbar to select an interpreter and add a new paragraph that uses that interpreter. When you click some of the icons, the paragraph that's created includes example code that you can run.

Note

You should not delete the start widget paragraph or rewrite the contents of this seeded paragraph.

Add Python Paragraph

- Click **Add Python Paragraph** in Model Studio to add a python paragraph in the Notebook and add the following scripting instructions.

1. To fetch connection objects:

```
conn = mmg.getConnection(<workspace name>)
```

This creates a cx_oracle based connection object to the datadom of the workspace being passed:

2. To fetch current Notebook and Model Objective details, use the following predefined parameters:

- `currentNotebookId` - fetches the current notebook ID
- `objectiveId` - fetches the current objective ID

To fetch the runtime parameters supported in Model Studio runtime or to access optional parameters passed from scheduler services:

- To access predefined batch runtime parameters like `taskId`, `batchrunid`, and `ficmisdate` passed during the model execution from scheduler, use the following variables : `$BATCHRUNID$`, `$TASKID$`, `$FICMISDATE$`.

Example:

```
%python
print('BATCHRUNID value is : ${BATCHRUNID$}')
print('TASKID value is : ${TASKID$}')
print('FICMISDATE value is : ${FICMISDATE$}')
```

- To access any optional parameters passed during the model execution from scheduler, use the below sample script:

```
%python
print('threshold value is : ${threshold}')
```

Note

Threshold is the optional parameter passed during the model execution.

Note

For more details on the predefined functions available in MMG for python scripting, see [Appendix -I](#).
Click the icon to run the script.

Note

The execution result from the Pipeline Canvas screen is not displayed in the notebook tab.

Simulations

Simulation run is for executing a single notebook in parallel by giving different values for same parameters. The simulation flow allows for iterative execution along that path with input drivers (variables) that are passed through a parameter set.

You can either create a new parameter set or use the existing parameter set and execute it from this tab. In addition, you can select or deselect the link types which you want to execute.

Figure 11-105 Simulation tab

» [Choose from parameter sets](#) ▶ ✕

Parameter sets:
Default(From notebook) ⌵

[Save parameter set](#)

MISDATE
2023-07-12 📅

Batch Run Identifier
Batch_simulation_45bf0b04-063a-4d24-9efa-40e0

Task Identifier
Task_0

▼ Run Stats

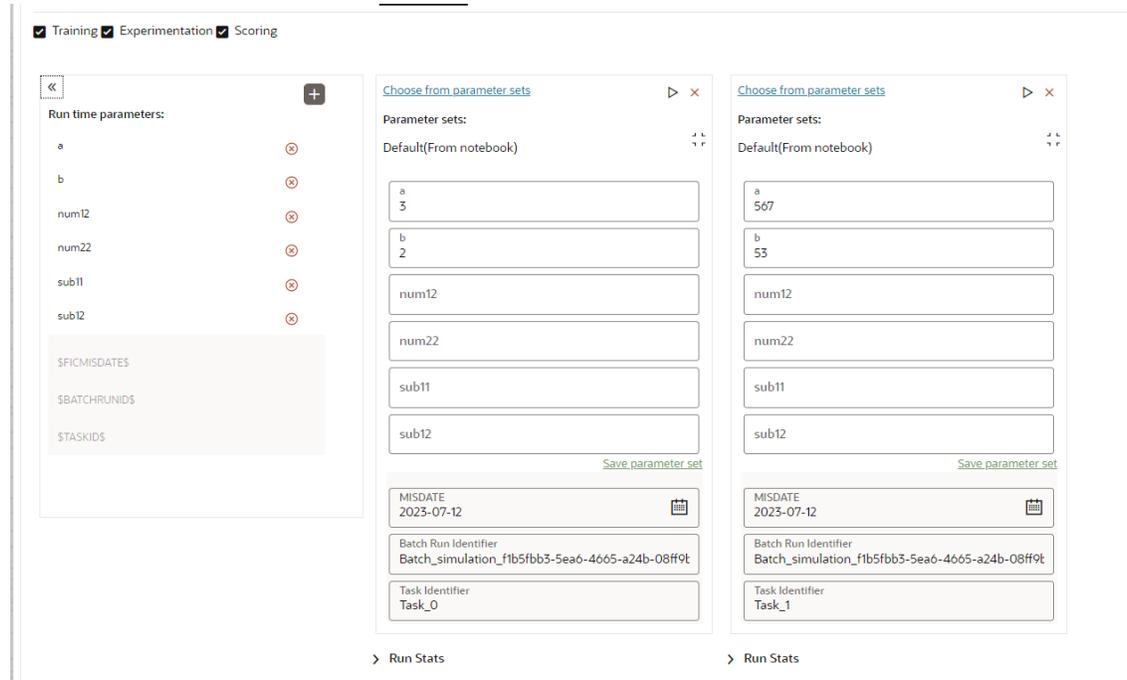
Start time: NA
End time: NA

▼ Outputs

No items to display.

If you want to add different values for the same parameter, click Add Run. You can add any number of Run based on your requirements and then click Execute all to execute at one time. The Batch Run Identifier will be same for all runs and the Task id will be incremented for newly created run.

Figure 11-106 Run Stats



The Run Stats displays the execution start and end date. The Outputs of the tracked nodes will be shown under the inputs of each run panel.

Execution History

This section of Pipeline Designer shows the history of the executed pipelines.

Note

Add file output button in execution history tab that opens file manager at a specific path exposed by the save dataframe widget in the pipeline window. When the save dataframe widget is not available in the pipeline, show that the path does not exist from file manager.

Figure 11-107 Execution History tab

Navigation: Pipeline Canvas | Dashboard | Notebook | Simulations | **Execution History** | Compare

Search:

Batch Run Identifier	Inputs	Task Identifier	Status	Outputs	Canvas view	MISDATE	Start Time	End Time
Batch_auto_d56c8079-eccl-4383-a2d1-d26400bec9b8	Custom	task1	⊙	📄	✖	2023-08-07	7 Aug 2023, 18:15:18	7 Aug 2023, 18:15:18
Batch_simulation_771b3711-5744-491c-8a45-08fa2fcd...	Default (from notebook)	Task_0	▲	📄	✖	2023-07-12	12 Jul 2023, 17:14:42	12 Jul 2023, 17:15:30
Batch_auto_ccceb145-aa70-4b45-b1bb-d4b55eecd7e0	Custom	task1	▲	📄	✖	2023-06-26	26 Jun 2023, 16:55:16	26 Jun 2023, 16:56:15
Batch_auto_ccceb145-aa70-4b45-b1bb-d4b55eecd7e0	Custom	task1	▲	📄	✖	2023-06-26	26 Jun 2023, 16:55:15	26 Jun 2023, 16:56:17
Batch_auto_ccceb145-aa70-4b45-b1bb-d4b55eecd7e0	Custom	task1	▲	📄	✖	2023-06-26	26 Jun 2023, 16:55:10	26 Jun 2023, 16:56:14

Page 1 of 2 (1-5 of 6 items) | < | 1 | 2 | >

You can compare and refresh the executions by clicking on **Compare** and **Refresh** icons.

To view the inputs, output, and canvas view, click on the corresponding options in the table.

Clicking the **Output** icon displays the Output Details page, where you can view "Executed-only" paragraph outputs. By default, this option is enabled, you can disable this option based on your requirements.

By default, **Tracked** option is selected that displays outputs tracked from the time of the execution. Select **All** option to view all the outputs.

Compare

This section allows you to compare the models with Champion model.

To compare:

Navigate to Pipeline Designer window and click **Compare** tab.

This shows the following comparison details:

- Model Properties
- Model Inputs (Last Execution Details)
- Audit Log
- Model Script
- Model Output

Figure 11-108 Compare tab

Pipeline Canvas Dashboard Notebook Simulations Execution History **Compare**

<input type="checkbox"/> Highlight Same Data	feb08 ver 2	feb08 ver 0
Model Properties		
Objective	obj1	obj1
Description	test	test
Version	2	0
Language	Default	Default
Technique		
Model Inputs (Last Execution Details)		
Status of execution		COMPLETED
Start time		Feb 8, 2023, 1:08:17 PM
End time		Feb 8, 2023, 1:08:33 PM
a		2
b		2
\$FICMISDATES		2023-02-08
\$TASKIDS		task1
\$TASKID		task1
\$MISDATE		2023-02-08
\$BATCHID		Batch_auto_d3f6eb01-0b78-4129-ac11-50f0d867d3bb
\$BATCHRUNIDS		Batch_auto_d3f6eb01-0b78-4129-ac11-50f0d867d3bb
Audit Log		
Created By	MMGADMIN	MMGADMIN
Created Date	Feb 8, 2023, 1:09:24 PM	Feb 8, 2023, 1:02:20 PM
Modified By	MMGADMIN	

Parameter Set

The Parameter Set page displays a list of parameter sets added by the user. For each parameter set, the following information is shown: the set code (also known as the threshold code), a description of the set, its champion status, and a list of available actions (edit, clone, delete, export). A checkbox is located on the left side of each parameter set; when checked, it indicates that the corresponding set will be used during pipeline execution.

To add a new parameter set follow these steps:

1. Click the "+" icon at the top right of the page.

You will need to provide a set code, which serves as the name of the parameter set. The description field is optional and can be used for your own reference.

2. Click the "+" icon next to Execution Parameters to add new key-value pairs.

This will insert an empty key-value pair where you can input the desired values.

The newly created parameter set will be displayed in the list. Click the **star** icon under the Prime Reference column to assign champion status to this set.

Note

Only one set can hold champion status at a time.

You can perform several actions on the parameter set from the Actions column:

- **Edit:** Opens the same popup as the Create Parameter Set option. You can modify the description and key-value pairs, but the set code cannot be changed.
- **Clone:** Opens the Create Parameter Set popup with values pre-filled from the selected set. You will need to provide a new, unique set code.
- **Delete:** Removes the parameter set from the list.
- **Export:** Generates a ZIP file containing the parameter set, which is downloaded to your local machine. A corresponding dump is also created on the server.

There are also several buttons located at the top right of the page:

- **Import:** Opens a popup to upload a ZIP file containing an exported parameter set. A new parameter set will be created based on the contents of the file.
- **Create:** Opens the Create Parameter Set popup, as described earlier.
- **Edit:** Opens a Bulk Edit popup, allowing you to edit multiple parameter sets, simultaneously.
- **Delete:** Deletes the selected parameter sets.
- **Refresh:** Refreshes the list of parameter sets.

Import/Export Parameter Sets using Utility

You can import and export the parameter sets between Sandbox to Production or vice versa using the Utility.

Prerequisite:

Before you import, ensure the Parameter Sets artifacts are available in the <installed path>/ftpshare.

To import the parameter sets:

- Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:

```
./parameter_set_export_import_utility.sh IMPORT TARGET_WORKSPACE LOGIN_USER
LOCALE FILE_NAME OBJECTIVE_ID OVERWRITE_FLAG
```

Example:

```
./parameter_set_export_import_utility.sh IMPORT CS SAUSER en_US
CS_1638105398036_set-1 1638105398036 true
```

Table 11-26 Fields and icons on the Sandbox Summary page

Field	Description
Import	Command to import the Parameter Set into Workspace
Workspace	The name of the Workspace to which you are importing the Parameter Set
Login_User	The logged in user name.
Locale	The application language preferences For example: en_US
File_name	The file name with the following format: Workspace name_Objective ID_Threshold Code For example: CS_1638105398036_set-1
Overwrite_Flag	To be set True, if existing threshold has to be overwritten. Else, False.

To export the parameter set, perform the following steps:

Prerequisite:

Before you export, ensure the Parameter Sets are available in the UI / setup.

1. Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:

```
./parameter_set_export_import_utility.sh EXPORT WORKSPACE LOGIN_USER
LOCALE THRESHOLD_CODE OBJECTIVE_ID
```

Example:

```
./parameter_set_export_import_utility.sh EXPORT CS SAUSER en_US set-1
1638105398036
```

Table 11-27 Fields and icons on the Parameter Set page

Field	Description
Export	Command to export the Parameter Set from the Workspace
Workspace	The name of the Workspace from which the Parameter Set is exported
Login_User	The logged in user name.
Locale	The application language preferences. For example: en_US
Threshold Code	Parameter Set Name
Objective ID	The Objective ID

Scheduler Configuration UI

A new UI is introduced for Scheduler notification and other configurations.

To integrate the new UI to MMG, a new user group called SCHEDULERADMIN is created, which has all roles and functions related to Scheduler.

The SCHEDULERADMINACCESS Role is created, which has all functions related to LINKSCHDLR.

The new function LINKSCHDLRCNF opens the Scheduler Configuration. The SCHEDULERADMIN group is mapped with BATCH_ADV, SCHEDULERADMINACCESS, and BATCH_ADMIN_ROLE.

Note

Admin UI is now available. Now the user is able to configure email notification to specific users. To access UI user requires BATCH_ADMIN role. This role is default mapped to the seeded group SCHEDULERADMIN group. All the configurations are SYSTEM level, which means, if the configuration is changes in one workspace the same will be reflected in all. To access UI user requires BATCH_ADMIN role. This role is default mapped to the seeded group SCHEDULERADMIN group.

Publish Data Studio

After creating the Draft Models, publish the Notebooks, which have the Model script.

To create a Scoring Model:

1. Create a Draft Model. See the [Create Draft Models](#) section for more information.
2. Click next to corresponding Draft Model and select **Publish Data Studio** option.

The Publish Model window is displayed.

Figure 11-109 Publish Model

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

Table 11-28 Field and its description for the Publish popup window

Field or Icon	Description
Model Name	The field displays the name of the Model. Modify the name if required.
Model Description	The field displays the description for the Model. Enter or modify the description if required.
Technique	Enter the registered technique to use.
Run Version	Select a run version.

Table 11-28 (Cont.) Field and its description for the Publish popup window

Field or Icon	Description
Script	The table displays the Paragraphs created in the Training Model. Select the Paragraphs that you want to use to create the Scoring Model. Track Output - Select this to track the output of the paragraph.

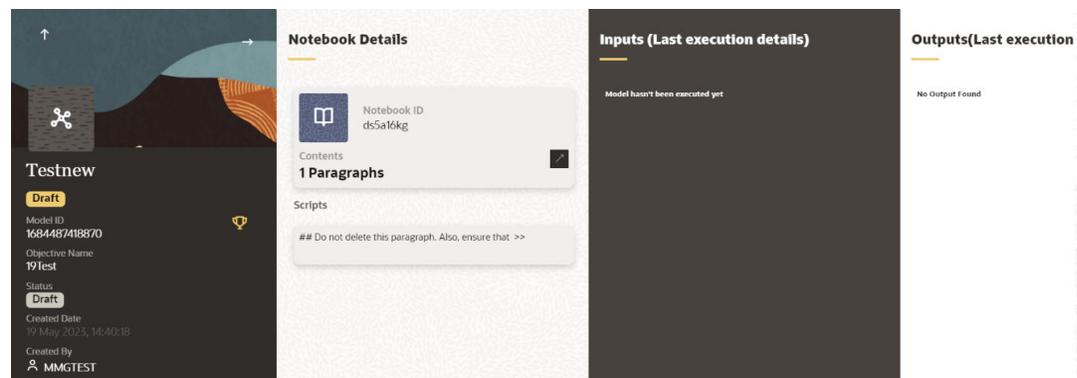
4. Click **Publish**.

Scope Detail

To view the scope details of the model:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Click next to corresponding Model and select **Scope Detail** option. This is available for both Draft and Published models.

The details such as Notebook, Inputs (Last execution details), Deployment Details, and Outputs (Last execution outputs) are displayed.

Figure 11-110 Scope Details

View Model Details

You can view model information for deployed models, models that require approval, and so on.

To view model details:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
The icon indicates that Model 1 is the champion.
3. Hover over the model records to view the various icons.
The icon actions are listed in the following:

- a. Download
- b. Open in Pipeline Designer. See the [Create Paragraphs using Pipeline Designer](#) section for more details.
- c. Scope Detail. See the [Scope Detail](#) section for more details.
- d. Delete Model.

Model Governance

After comparing models, you must understand the Model Governance system in the application. The Model Governance has an impact on how the application functions with the various user types and the requests they can place from the Model Details window. You must understand Model Governance before you request model acceptance, review models, or approve models for production.

There are three types of users:

- Requester
- Reviewer
- Approver

The User-Level Approval Restriction feature provides the user-level approval restriction for model deployment in MMG. This is controlled by the global configuration flag `IS_SELF_USER_AUTH_ALLOWED` within the `NEXTGENEMF_CONFIG` table. There are two possible flag values:

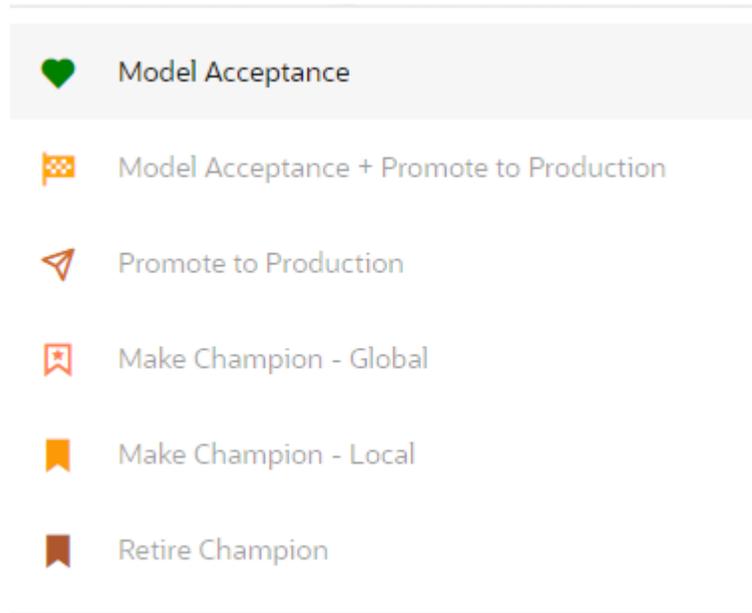
- **{*}Y{*}**: This is the default value and allows users to approve their own model submissions. This is backward compatible.
- **{*}N{*}**: This prevents users from approving models they have themselves requested for approval.

Note

You must restart the application everytime you update this value.

The request consists of the following phases in the Request drop-down list (see the [Request Model Acceptance](#) section for how to access the drop-down list in the Model Deployment window):

- Model Acceptance
- Model Acceptance + Promote to Production
- Promote to Production
- Make - Champion - Global
- Make - Champion - Local
- Retire Champions

Figure 11-111 The Request Drop-down List

The values in the drop-down list are active based on the type of user (Requester, Reviewer, or Approver) and the phase that the model is in (accepted, promoted to production, global champion, and so on). Let us look at these with a few examples.

Example 1:

Assume that you are a user with Requester privileges, and you create a model. Now you can request for the model to be accepted on the Model Details window from the **Request** drop-down list. The values enabled for selection are **Model Acceptance** and **Model Acceptance + Promote to Production**. Let us proceed and assume that you select Model Acceptance, then a user with Reviewer privileges forwards your model to a user with Approver privileges. At this stage, the Approver can choose to reject or accept your model acceptance request. A rejection would bring the model back to the initial state with comments on the updates required before it can be requested for acceptance again. However, if the Approver accepts your model, then the **Make Champion- Local** selection is enabled when you log in. You can create many models and send them for acceptance. After acceptance, any model that is accepted can be made the champion on your local workspace at any time replacing the earlier local champion.

Example 2:

Assume that in the previous example, you selected **Model Acceptance + Promote to Production**, then a Reviewer forwards it to the Approver. The Approver, at this stage, chooses to promote the model to production by selecting **Promote to Production**. The model is now available in the production environment and the Approver can choose at any time to select a model from these models in production and select **Make Champion- Global**. If there exists a Champion model in the production environment, then it will be replaced by the new global champion. However, the earlier champion will still be available in the production environment along with other models and the Approver can choose to make it the global champion again at any time or select any of the other models and make one of them the global champion.

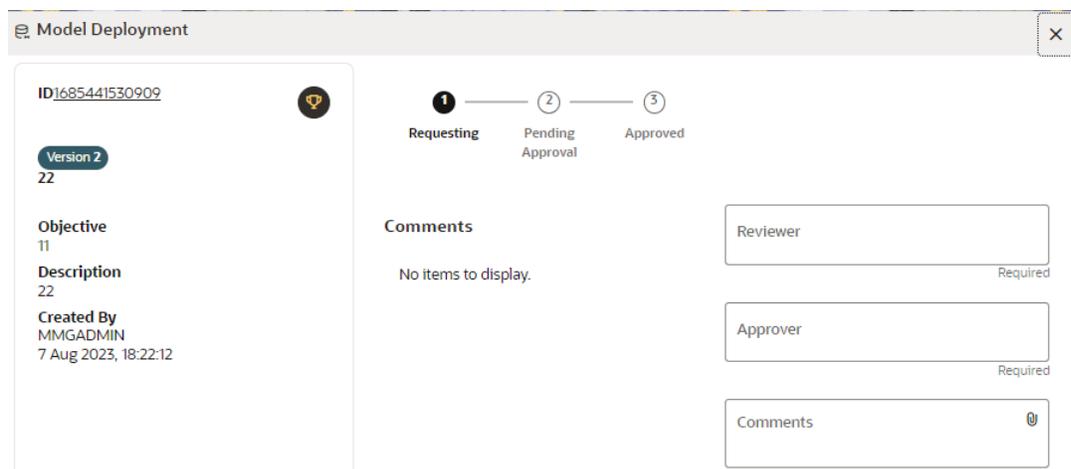
Request Model Acceptance

After comparing models, move the selected models to acceptance. Only a user with the Requester role can request for model acceptance. The model will be moved to review which will be available to Reviewer and Approver role, and then to acceptance is available to users with the Approver role, who can promote to production. See the [Understand Model Governance](#) section before you start here.

To request a model to promote to production, follow these steps:

1. Click **Model Pipeline** to display the **Model Pipeline** window.
The window displays folders that contain models and model records in a table.
2. Click on the Deploy icon on the Model to open it.
3. If a model is a champion, the icon is displayed on it.
4. To make a model a champion from this window, place the mouse in the selected model columns. and click to display the **Model Details** window. If a model is a champion model, the icon is displayed.

Figure 11-112 Model Deployment



5. Select the Reviewer group from the **Reviewers** drop-down list.
6. Select the Approver group from the **Approvers** drop-down list.
7. Enter comments in Comments and click **Attach** to attach files supporting the comments.
8. Use the following features on the window to perform additional actions.
 - View the model status change in the progress indicator. The Progress Indicator displays the various states of progress that the model has been through. Accordingly, you must request, review, or approve models.

Figure 11-113 Model Approval Progress Indicator



- Click the type of **Request** from the drop-down list:
 - Model Acceptance: To review and accept the model creation.
 - Model Acceptance + Promote to Production: To review and promote the model to production.
 - Make Champion- Local: If the model is not the champion model, select to make it the local champion.
 - Promote to Production: To promote a model to production
 - Make Champion- Global: If the model is not the champion model, select to make it the Global champion.
 - Retire Champion: To retire a Champion model
- Comment History: A record of comments entered in the cycle of model creation and approval with the feature to download attachments.

The model sent for acceptance or for promotion to production is now displayed to a Reviewer to review it and then to Approver when signed in, who must either accept the request or reject it.

Review Models and Move to Approve or Reject

The Reviewer must provide comments describing the action (approve or reject). If comments are related to rejection and if the Approver rejects, then model goes back to the Requester to make changes or to delete it. If comments are related to approval, then model moves further in the workflow and is displayed to an Approver. See the [Understand Model Governance](#) section before you start here.

To review a model:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.

The window displays folders that contain models and model records in a table.

3. Click to display the **Model Details** window.
4. Review the details and send it back to the Requester for modifications or send it to an Approver.

Approve Models and Promote to Production

The models reviewed and set to promote to production by either the Requester or Reviewer is displayed to the Approver when signed in. The Approver has to either reject the model and send it back to the requester with supporting comments or approve it for pushing to production. See the [Understand Model Governance](#) section before you start here.

Note

When dataset has used the datasource which is of order (N) for example 5 , and the Production workspace does not contain the datasources at order 5, then promotion of models containing dataset from Sandbox to Production workspace fails. To remedy this issue, ensure that Sandbox and Production workspace contain the same number of Datasources before you perform promotion of models.

To approve or reject models:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Select a model from the records in the objective.
4. Click to display the Model Details window.
5. Click **Approve** or **Reject** with appropriate comments.

Deploy Models in Production and Make it a Global Champion

After approving the models, deploy it to the production environment. You must have an Approver function role and privileges to do this activity.

Note

Sandbox should have the production workspace attached in order to have this option enabled.

To deploy Models in production:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Select the model to deploy and click **Model into Champion** to display the details of the model. If a model is a champion, the **Champion** icon is displayed.

Import a Model Data into a New Model

The model data from existing models in .dmp format and the existing model data in .dsnb can be imported during the creation of a new model.

Note

The import should happen inside an Objective only.

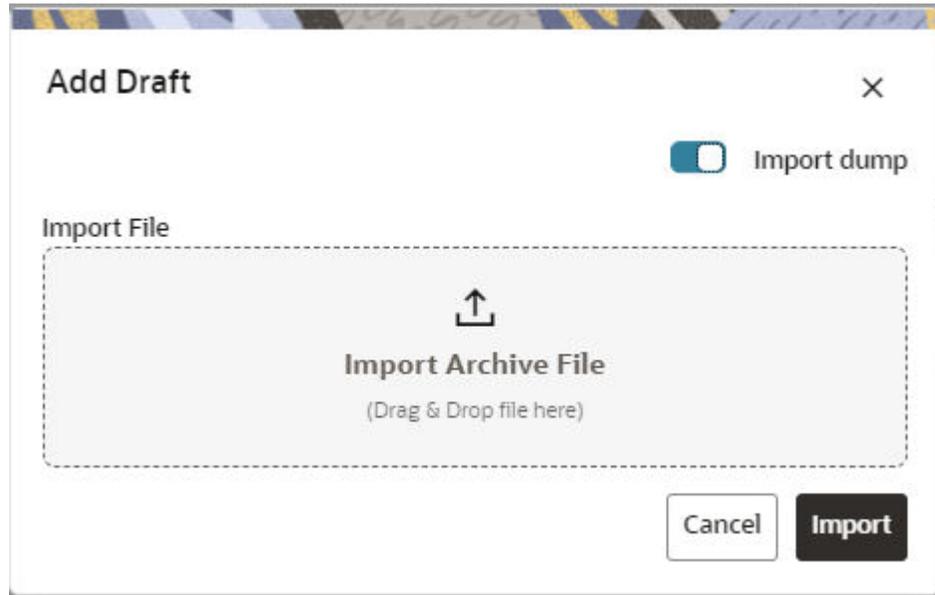
The import of model data lets you reuse and extend on model creation. This topic is part of the procedure of creating Draft Models and after creating a new model using this method, see the [Create Draft Models](#) section for instructions on how to proceed further.

To import model data:

1. Click **Launch Workspace** next to corresponding Workspace to Launch Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.
The window displays folders that contain models and model records in a table.
3. Select an **Objective**.

4. Double click a model to display the model versions in the expanded display.
5. Hover over a model and click **Download** to download the model data dump.
6. Click **Add** and **Drafts** to display the **Model Details** dialog box for the creation of a new model.

Figure 11-114 Model Details - Import Dump



7. Drag the toggle switch to select **Import Dump**.
8. Drag and drop the file into the **Import Dump File** field or click in the box to open the file selector dialog and select a file.

Note

If the upload file size is more than 8 MB, you can increase the file size upto 100 MB by modifying the below properties. Restart of services are required once the file size is updated.

mmg-ui/conf/application.properties:

spring.servlet.multipart.max-file-size=8MB

spring.servlet.multipart.max-request-size=8MB

mmg-service/conf/application.properties:

spring.servlet.multipart.max-file-size=8MB

spring.servlet.multipart.max-request-size=8MB

9. Click **Import**. A new model is created by importing the model data dump of another model.

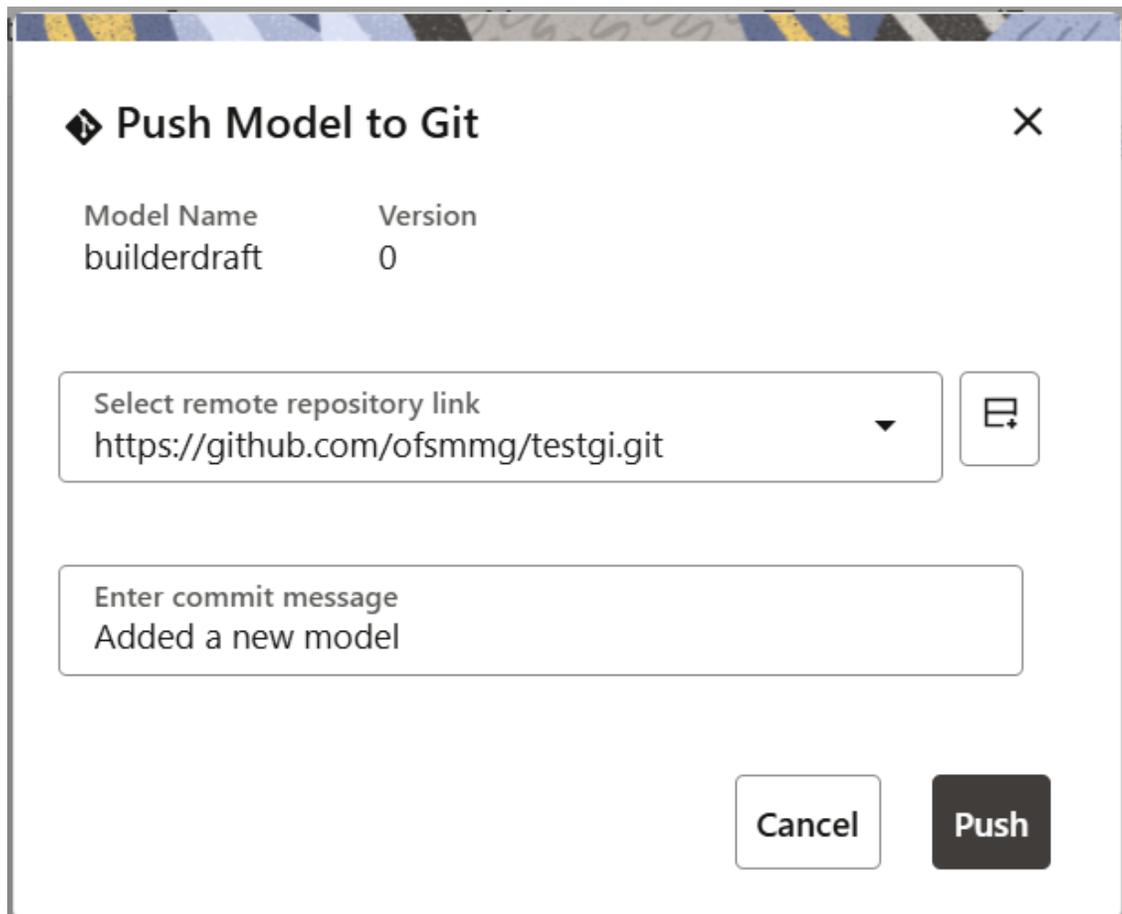
Git Functionalities

This section describes the following Git functionalities:

- Git Push
- Git Pull
- Git Clone

Git Push: This feature allows users to save and upload (push) changes made to the model to the remote Git Repository, but only on their assigned working branch that matches their environment (For example: Dev, QA). Users can commit their changes locally and then push them remotely with a single option. If the model is not yet linked to any repository, the system will create a new local Git repository and connect it to a remote one using a repository URL provided by the user, ensuring the model is properly tracked and synchronized.

Figure 11-115 Git Push



Push Model to Git

Model Name	Version
builderdraft	0

Select remote repository link
https://github.com/ofsmmg/testgi.git

Enter commit message
Added a new model

Cancel Push

Git Pull: This feature allows users to fetch and load a Model Pipeline from a remote Git Repository into their environment. Users can select from any linked environment branches and also choose a specific commit version to pull, giving them flexibility to work with different stages or snapshots of the model as needed.

Figure 11-116 Git Pull

Pull Model From Git X

Model Name	Version
devTest0	0

Select remote repository link
https://github.com/ofsmmg/testgi.git

Connect ✓

Select Environment
dev

Select Commit
V.2. python changes

Cancel Pull

Git Clone: This feature allows users to clone a model from a remote Git Repository by selecting the model, its environment branch, a specific Git commit version, and the corresponding MMG version, enabling precise replication of the desired model state locally.

Figure 11-117 Git Clone

Clone Model From Git ✕

Select remote repository link
https://github.com/ofsmmg/testgi.git

Connect ✓

Select Environment
dev

Select Model
DEMODRAFT

Select Model Version
0

Select Commit
V.2. second commit with pyt

Cancel Clone

Execute Models using Scheduler Service

The models that you have created require that they are executed using Scheduler Service before they can be available to the users of OFSAA applications. For more information on this, see the [Scheduler Service](#).

Define a Task

The following three Components are supported during the task creation:

- Model
- Populate Workspace
- Custom

Figure 11-118 Create Task

Create Task Save Close

▼ Task Details

Task Name *

Task Description

Task Type

* Components

Batch Service URL

Task Service URL

Enter the following details in Task Details section:

- **Task Name:** Enter the task name.

Note

The Task Name must be alphanumeric and should not start with a number. The Task Name should not exceed 60 characters in length. The Task Name should not contain any special characters except underscore (_).

- **Task Description:** Enter the task description. No special characters are allowed in Task Description. Words like Select From or Delete From (identified as potential SQL injection vulnerable strings) should not be entered in the Description
- **Task Type:** Select the task type from the drop-down list. The options are REST and SCRIPT. You can enter Shell script for Model, Populate Workspace, Custom components. Status key in the curl command should be in uppercase as STATUS.
- **Batch Service URL:** Select the required Batch Service URL from the drop-down list. This can be blank, and you can provide the full URL in the Task Service URL field.
- **Task Service URL:** Enter task service URL if it is different from Batch Service URL.

Note

Task Parameters will vary based on the selected Component.

When Component is Model

The following window shows the Task Parameters for Model Component.

Figure 11-119 Task Parameters

Task Parameters				
Parameter	\$BATCHDATE\$	Value	Batch Date	<input checked="" type="checkbox"/>
Parameter	\$BATCHRUNID\$	Value	BATCHRUNID	<input checked="" type="checkbox"/>
Parameter	Global Filter	Value	<input type="text"/>	<input type="checkbox"/>
Parameter	Table Filters	Value	<input type="text"/>	<input type="checkbox"/>
Parameter	Additional Parameters	Value	<input type="text"/>	<input type="checkbox"/>

Note

Fields marked with * are mandatory fields.

- **Batch Date:** Shows the batch execution date. You can enable or disable this parameter.
- **Batch Run ID:** Shows the batch execution run ID. You can enable or disable this parameter.
- **Objective:** Select the Object which you want to use for execution. For more information, see the Create Objective (Folders) section. The Sub Objective is displayed with path. For example, if Test1 is Objective and Test11 is Sub Objective, and you want to use Test11 Objective for execution, then select this field as Test1/Test11.
- **Model:** Select the Model of selected Objective. It can be any specific model of Objective or All models of Objective.
 - If the ALL_CHAMPION is selected here, then Objectives with no Champion Model is skipped, and the Objectives with Champion Models gets executed.
 - If CHAMPION is selected, and no Champion is present, then Model Execution gets fail.
- **Link Type:** Select the link type for execution. For example: Training, Scoring, or Training+Scoring. For more information, see the Links in the Pipeline Designer section.

- **Synchronous Execution:** You can set this parameter to Yes or No.
 - If Synchronous Execution is set to Yes, then execution will wait for the notebook execution status.
 - If Synchronous Execution is set to No, then execution will not wait for the notebook execution status, it will trigger the notebook and update task status as successful in batch monitor.
- **Optional Parameters:** This is used pass the parameters dynamically.
For example:
`model_group_name=LOB1,benford_flag=Y,benford_digit=1,from_date=01-Jul-2020,to_date=31-Jul-2021`
Model_group is parameter defined in model and value can be passed here.

The Create Task window also shows the following Header Parameters, which are not editable:

- **User:** logged in user name
- **Workspace:** shows the launched workspace name
- **Locale:** shows the locale. For example: en_US

When Component is Populate Workspace

If you select Component as Populate Workspace, then to use the data population, enter the following parameters.

- **Additional Parameters**
- **Data Filters - Global level**
- **Data Filters - Table level**
- **Data Filters - Hint**

For more information, see the [Populating Workspace](#) section.

Figure 11-120 Populate Workspace screen

Parameter	Value	Checked
\$BATCHDATES	Batch Date	<input checked="" type="checkbox"/>
\$BATCHRUNID\$	BATCHRUNID	<input checked="" type="checkbox"/>
Global Filter	<input type="text"/>	<input checked="" type="checkbox"/>
Table Filters	<input type="text"/>	<input checked="" type="checkbox"/>
Additional Parameters	<input type="text"/>	<input checked="" type="checkbox"/>
Header Parameters		
user	aaaiuser	<input checked="" type="checkbox"/>

SS \$RUNSK support for processing Delta Data by Data Pipeline and Match Rules

A new parameter to pass RUNSK is being implemented in the Scheduler UI.

Dynamic Parameters screen for new batches are created.

1. RUNSKEY will be present in Task Parameters while copying task for existing batches.
2. RUNSKEY should be visible in View Execution Parameters - Task Parameters in Monitor screen.
3. RUNSKEY will change just as BatchRunID in "Rerun" Case.
4. RUNSKEY will remain same just as BatchRunID in "Restart" Case.
5. RUNSKEY will be stored for batches in "AAICL_SS_BATCH_RUN" in column "N_RUNSKEY".

Export/Import Models via Utility

You can import and export the models between Sandbox to Production or vice versa using the Utility.

Prerequisite:

Before you import, ensure the Model artifacts are available in the <installed path>/ftpshare.

To import the models:

- Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:

```
./model_export_import_utility.sh IMPORT WORKSPACE LOGIN_USER LOCALE FILE_NAME
Y/N
```

Example:

```
./model_export_import_utility.sh IMPORT CS SAUSER en_US CS_1638105398036_0.zip
Y
```

Table 11-29 Fields and icons on the Sandbox Summary page

Field	Description
Import	Command to import the Model into Workspace
Workspace	The name of the Workspace to which you are importing the Model
Login_User	The logged in user name.
Locale	The application language preferences For example: en_US
File_name	The file name with the following format: Workspace name_Model ID_Model version.zip For example: CS_1638105398036_0.zip

Note

If you enter input as **Y**, the utility imports the models in the approved status.
If you enter as **N**, the models are imported but not in the approved status.

To export the models, perform the following steps:

Prerequisite:

Before you export, ensure the Models and Drafts are available in the UI / setup.

1. Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:
./model_export_import_utility.sh EXPORT WORKSPACE LOGIN_USER LOCALE
MODELID MODEL_VERSION

Example:

```
./model_export_import_utility.sh EXPORT CS SAUSER en_US 1638105398036 0
```

Table 11-30 Fields and icons on the Sandbox Summary page

Field	Description
Export	Command to export the Model from the Workspace
Workspace	The name of the Workspace from which the model is exported
Login_User	The logged in user name.
Locale	The application language preferences. For example: en_US
Model ID	The Model ID
Model Version	The version of the Model

Note

Currently, if the user gives Production Workspace name during the import of model dump through the utility, the utility migrates the dump to the Production Workspace as well. This should be restricted, as the models in sandbox may or may not be promoted, and the invalid status will be shown in the Production Workspace.

View Models

The View Models feature launches the OFS Data Studio window. You can view models on this window.

To use View Models:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.

The window displays folders that contain models and model records in a table.

3. Click **Action** next to corresponding Model and select Open in Pipeline Designer.

See the [Create Paragraphs in Pipeline Designer](#) section for details on how to use the OFS Data Studio.

Edit Models

The editing of models created versions that are different from the previously saved model and the cycle of [Model Governance](#) applies to any edited model. You can edit models from the OFS Data Studio window using Python scripting language.

You can edit the script of version 0 only even in Pipeline Designer and Studio. It is not possible in other versions.

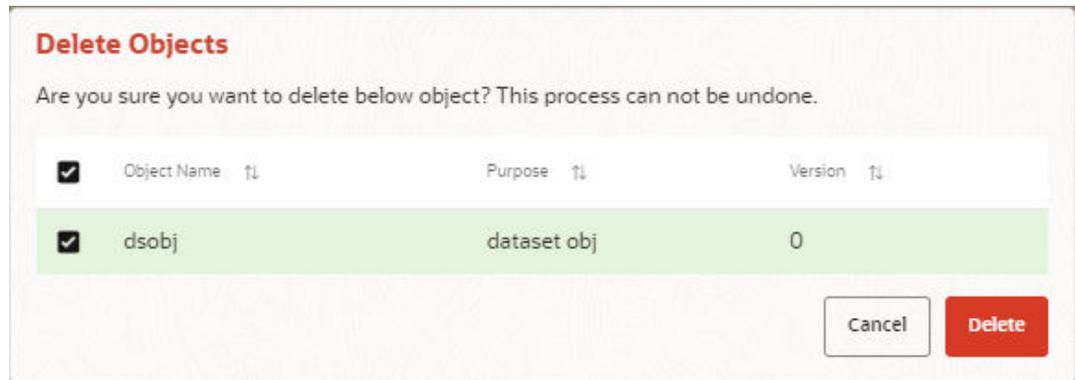
Delete Objectives and Draft Models

To delete Objectives and Models that exist in the Objectives, follow these steps:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click **Model Pipelines** to display the **Model Pipelines** window.

The window displays objectives that contain models and model records in a table.

3. Click **Action** next to corresponding record (objective/draft) and select **Delete**.
4. You can select two or more objectives or models from the records.
5. Click **Delete** to display the Delete dialog box.
6. Click **Delete** to delete or click **Cancel** to cancel.

Figure 11-121 Delete Objects screen**Note**

If the Model is promoted to Production, you cannot delete the model even it is retired.

Programmatic Dynamic Forms and Fixed Dynamic Forms

The Programmatic Dynamic Forms and Fixed Dynamic Forms scripts are supported.

A dynamic form is a user input field that is generated from the code of a paragraph. Dynamic forms allow users to bind free variables in a paragraph.

For Example:**Textbox**

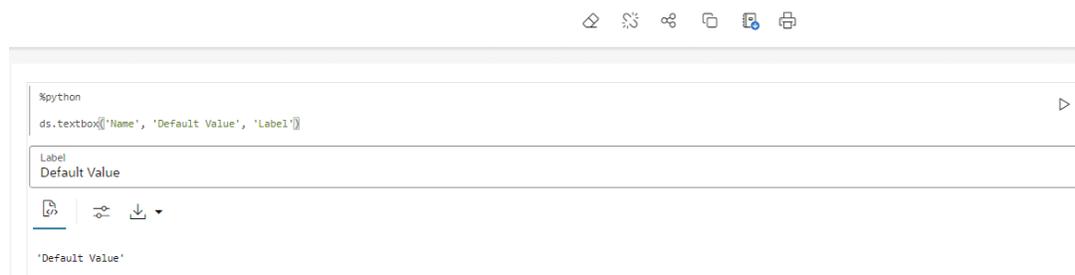
The Textbox dynamic form allows users to input any string of characters. Its format is as follows:

- Click **Add Python Paragraph** in Model Studio to add a Paragraph in the Notebook and fetch the runtime parameters supported in Model Studio runtime as shown in the following example. This is available for all the interpreters.

For example, enter as follows:

```
%python
ds.textbox('Name', 'Default Value', 'Label')
```

Click **Play** to run the script and display.

Figure 11-122 Run the script and display tab

Note

In the current release, the MMG Studio supports Textbox, Select, Slider, Checkbox, Date Picker, Time Picker and DateTime Picker dynamic forms.

Session Management in Model Pipeline

Overview

The application now supports enhanced session management for executing model pipelines. This feature resolves earlier limitations where pipeline executions were bound to an implicit session, with no visibility or control for users. The update brings consistency between pipeline and notebook behavior. Each session has its own context, keeping pipeline and notebook outputs in sync.

Figure 11-123 Model Pipeline Session screen

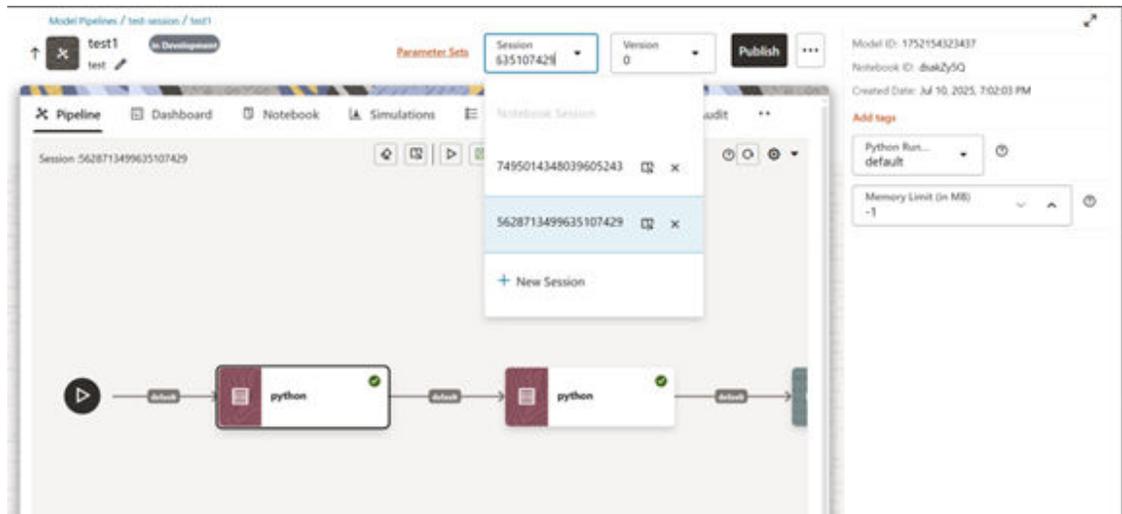
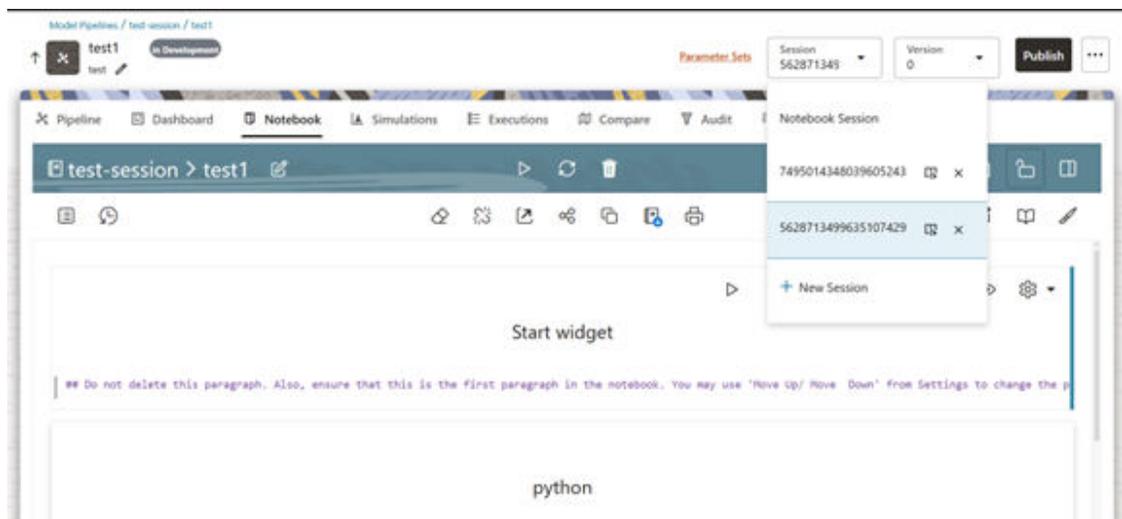


Figure 11-124 Session drop-down screen with multiple sessions shared between Pipeline Canvas and Notebook



Key Features

- Session dropdown available in both pipeline and notebook tabs.
- Default state is 'No session selected' users must create or choose a session.
- A prompt appears if execution starts without a selected session.
- Sessions can be created, switched, reinitialized, or cleared.
- Selected session syncs automatically across notebook and pipeline.
- Pipeline outputs and notebook results are isolated per session.

Note

- Session selection is mandatory before running pipelines.
- Sessions help isolate workflows, making debugging and reruns easier.

Creating a Session

Go to the pipeline or notebook tab, open the Session drop-down, and select 'Add Session'.

Figure 11-125 Add session

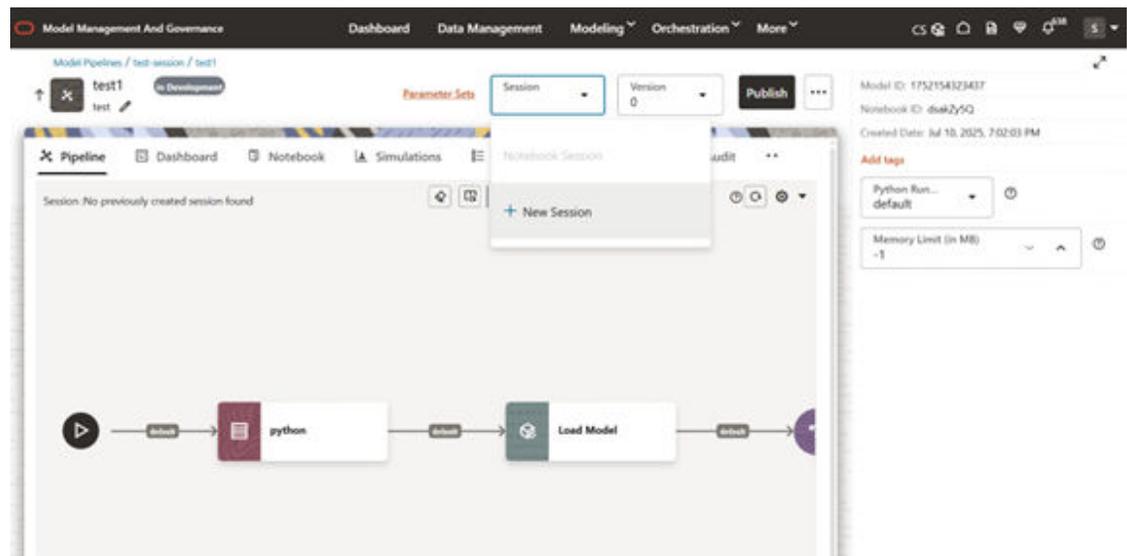
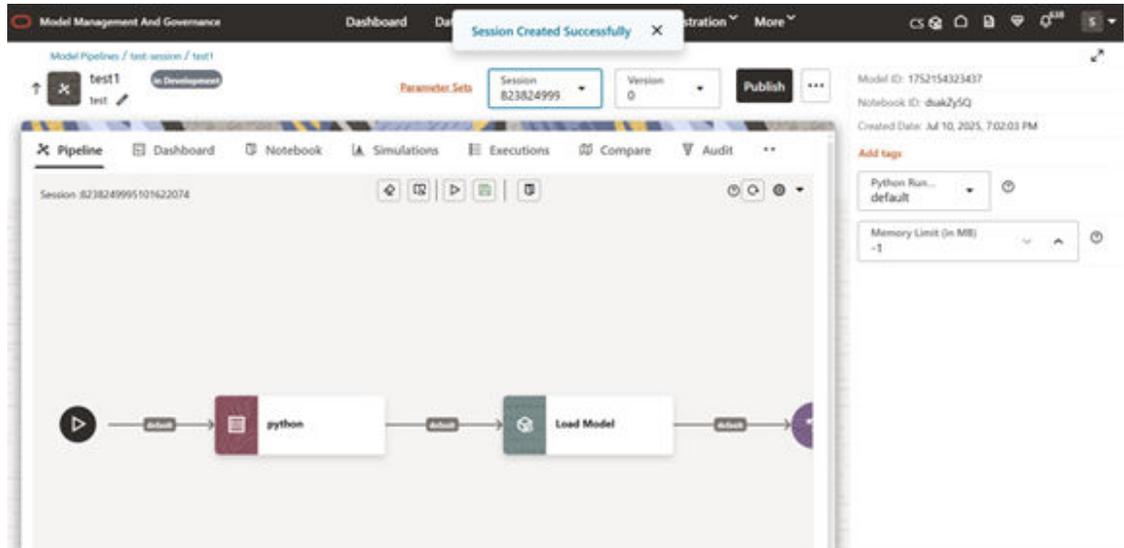


Figure 11-126 Add session created



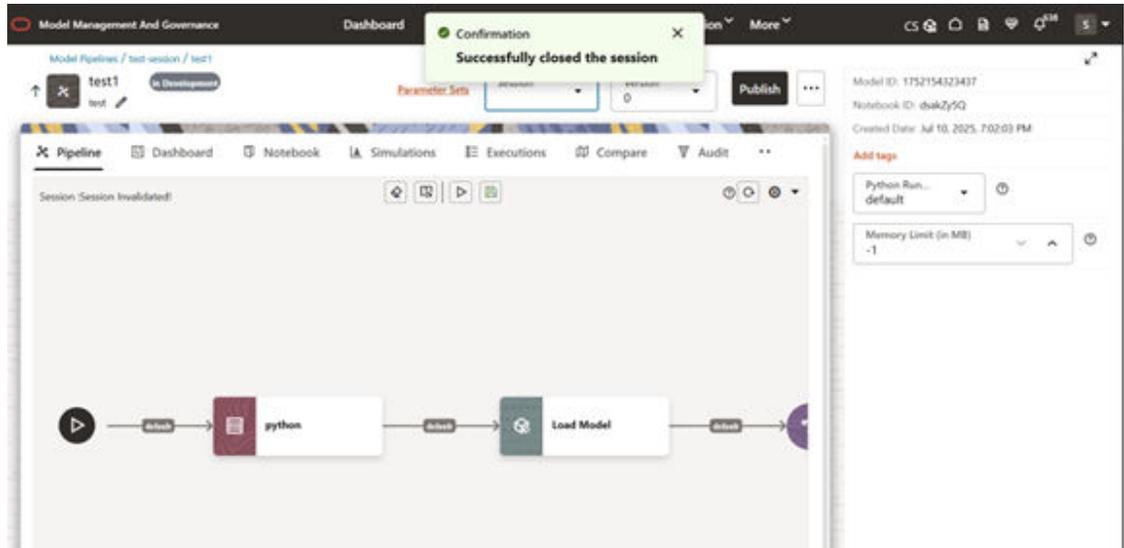
Clearing a Session

Clearing a Session In the Session drop-down, click the cross icon beside the session to delete it.

Figure 11-127 Clearing a Session screen



Figure 11-128 Closing the Session screen



Reinitializing a Session

Click the re-initialize icon next to a session in the drop-down to reload its state.

Figure 11-129 Reinitializing a Session

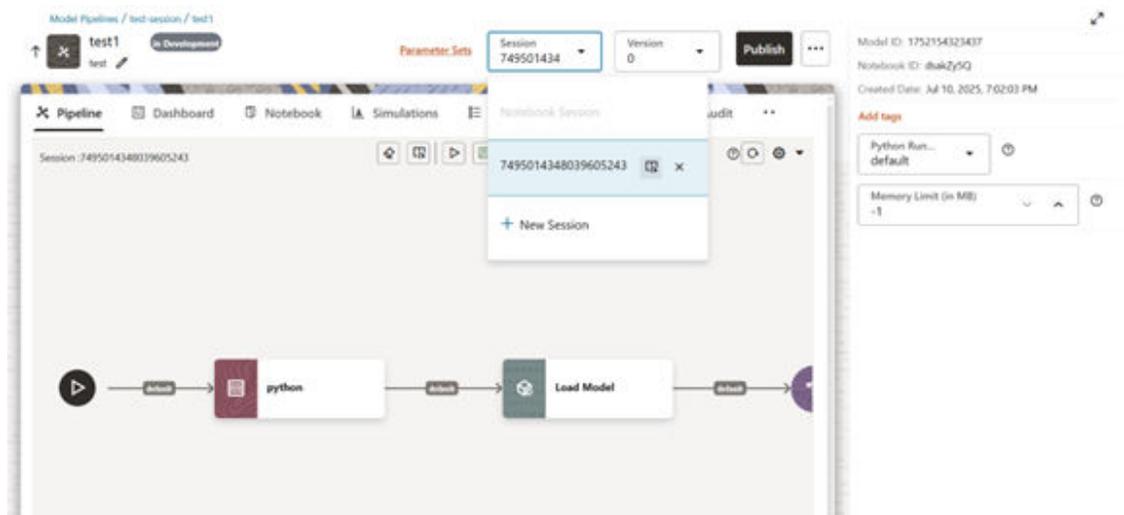
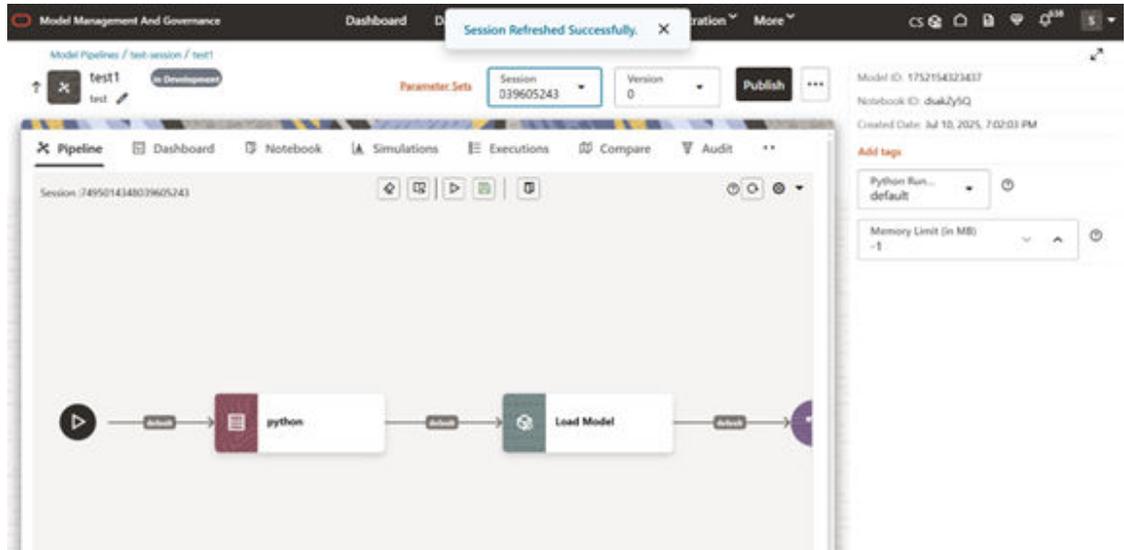


Figure 11-130 Refreshing a Session



Switching Between Notebook and Pipeline

Once a session is selected, you can move between Notebook and Pipeline tabs while keeping outputs in sync.

Figure 11-131 Switching between Notebook and Pipeline

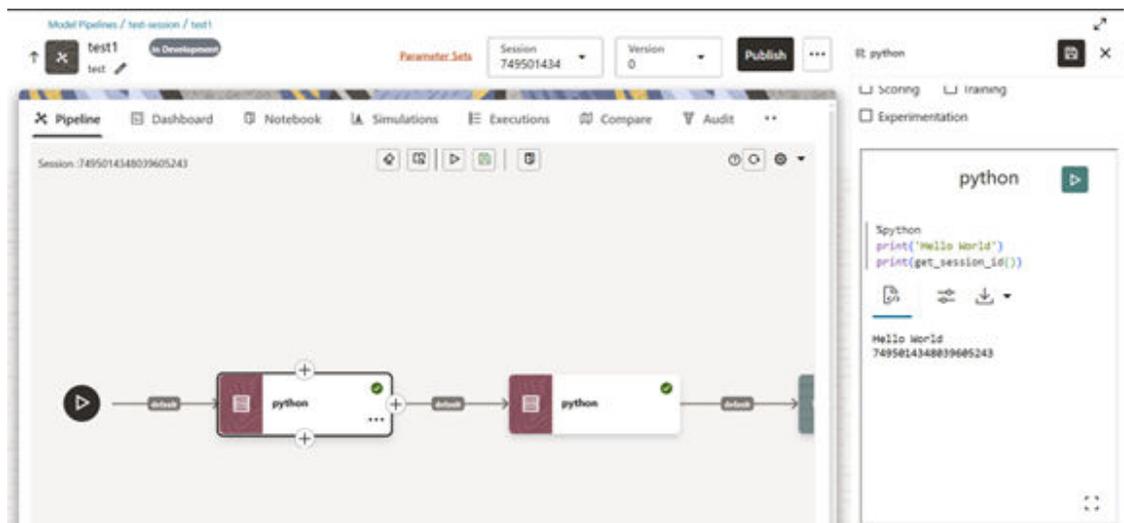
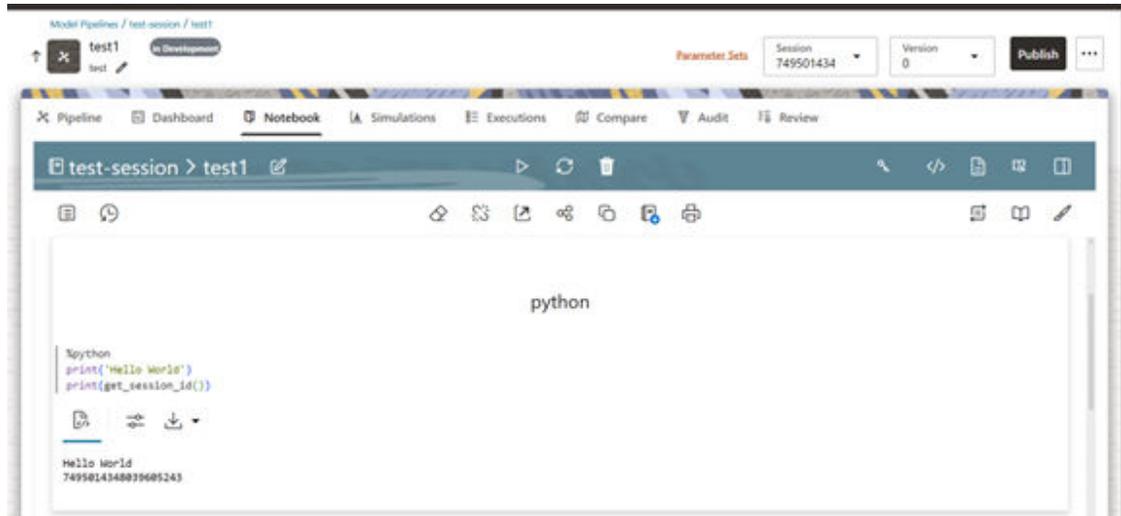


Figure 11-132 Notebook screen



Graphs

The Graph Pipeline feature enables you to transform data in relational tables into a graph.

This feature uses the latest technology to harness the power of Graph Analytics to give Financial Institutions the ability to monitor the data financial institutions effectively. The data is organized as nodes, edges, and properties (property data is stored on the nodes or edges). The results of analytics algorithms are stored as transient properties of nodes and edges in the Graph.

Users can harness the power of Graph Analytics using our in-built in-memory Oracle Graph Analytics Engine (PGX).

The Graph Pipeline functionality allows users to define graphs easily, attach underlying data, and match pipelines to populate data in the graph.

The Graph Pipeline functionality allows users to quickly create and configure a graph for use in advanced analytics. It can also manage and schedule the tasks required to run populate the graph on a periodic basis.

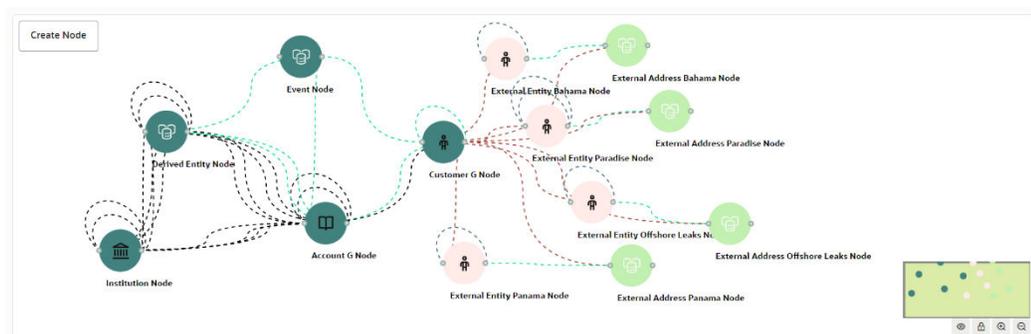
The Graph Pipeline functionality allows you to:

- Creating Graph Model from their existing relational data model
- Configuring the Graph Model through Pipeline UI
- Creating and Scheduling the data pipeline and matching (creation of similarity relationships) for created Graph Model
- Adding new sources and contextualizing the links quickly
- Standardizing the data before pushing it to Graph Model
- Using the following data source for Graph Model:
 - Oracle
 - File System (CSV)
- Using BD data source definitions for pre-seeded Workspace

- Using pre-configured Financial Crime Graph Model pipelines
- Using pre-seeded mappings of Graph Model properties with BD Data source properties
- Scheduling to create Graph by running pre-configured batches
- Blending data and getting insights from data quickly
- Load data into elastic search indexes so matching and entity resolution can occur in the graph
- Define matching rules for the generation of similarity edges in the graph.

An example of a preconfigured Financial Crime Graph Model with Nodes and Edges is provided here.

Figure 11-133 Example of Graph Model



The Graph Model defines the nodes and the relationship between them:

- **Node:** Represents a single entity with its attributes. The entity can be a single table or combination of tables merged on some common feature, it could be from a file, or any generalized component of multiple substructures merged for a specific reason/objective. For example, Customer.

A customer could be a single entity.

Customer, Salesperson, Manager, and so could be individual entities that can be structured together as a “Person” entity with a set of attributes persisted across individual entities.

- **Edge:** A connector between nodes or the same Node itself. Each edge can have a set of attributes associated with it. These edges will map to ‘join’ between two entities that could be direct or transitive.

- **Modeler Attributes:** Properties of the Node/Edge, and these are mandatory inputs. For example:

A Customer Node will have properties like Customer ID, Name, Age, Gender, etc.

A Transaction edge will have properties like Transaction ID, Date, Amount, From/To account, etc.

See the [Adding a Graph Pipeline](#) section for creating Graph, Nodes, Edges, and Scheduling.

Accessing the Graph Summary Page

The Graph Summary page gives access to the various graph pipeline functions such as create, view and delete.

To access the Graph Summary page, follow these steps:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click the **Graphs** icon.

This displays the graph pipeline in a tabular format.

The following table provides descriptions for the fields and icons on the Graph Summary page.

Table 11-31 Graph Summary Details

Field or Icon	Description
Search	The field to search for a graph pipeline. Enter specific terms in the field for which you want to search, and press Enter on the keyboard to display the results.
Name	The name of the graph pipeline.
Number of Nodes	The number of nodes present in the graph pipeline.
Number of Edges	The number of edges present in the graph pipeline.
Owner	The owner of the graph pipeline.
Created Date	The date on which the graph pipeline was created.
Delete	Click to delete multiple graph pipelines.
Add	Click to create a new graph pipeline. See Adding a Graph Pipeline section.
Action	Click Action icon to View/Edit/Delete/Download functions on the selected graph pipeline.

Note

The user is now able to execute/schedule a batch with SYSDATE-X days functionality.

Adding a Graph Pipeline

To add a graph pipeline:

1. Navigate to Graph Summary page.

This page displays the graph summary records in a table.

2. Click **Add**.

The **Add Graph** pop-up window is displayed. Figure: Add Graph window.

Figure 11-134 Add Graph window

The screenshot shows a dialog box titled "Add Graph". In the top right corner, there is a close button (X) and a toggle switch labeled "Import dump". Below the toggle are two text input fields: "Graph Name" and "Graph Description", both with "Required" labels to their right. At the bottom right of the dialog are two buttons: "Cancel" and "Add".

3. In the **Graph Name** field, enter a name for the graph. The name must be unique to a particular workspace.
4. In the **Graph Description** field, enter the description for the graph and click Add.

OR

You can import an existing Graph. Use the toggle button to select **Import Dump**.

The following page is displayed.

Browse the file and click **Import**. Once the Import is successful, the Graph Model page is displayed.

The Maximum Age of Old session for a graph is 7 days by default. The Maximum age of old session of 7 days specifies that graph would be retained for a period of 7 days. You can modify the description and Max Age of Old session by clicking on the **Setting** icon. If you want the batch to be created in read-only mode for scheduler screen, enable the option.

5. Click **Save**.
The user session of the Graph Pipeline will get refreshed after the set timeline.
6. You can perform the following:
 - a. [Create a Node](#)
 - b. [Create an Edge](#)
 - c. Toggle drawing which will enable or disable the user from dragging components
 - d. [Zoom In and Zoom Out](#)

Create a Node

To create a Node:

1. In the Graph Model page, click Create Node.

OR

Select the graph for which you want to create a Node, click on the Action button, and select the **Edit** option. In the Graph Model page, click **Create Node**.

The **Node Details** screen is displayed.

Figure 11-135 Node Details

Node Details [X]

Basic Information

Logical Name [Required]

Logical Description [Required]

Node Modeler Attributes Section

Search [+]

Logical Name	Data Type	Format	Action
No data to display.			

Page 1 (0 of 0 items) | < 1 >

Advanced Features

Manage Pipeline(s)

Search [Create] [Attach]

Pipeline Name	Action
No data to display.	

Page 1 (0 of 0 items) | < 1 >

- In the **Logical Name** field, enter a name for the Node or click **Setting** . The Setting pop-up window is displayed.

Figure 11-136 Setting

Setting [Select Icon] [Select Color] [X]

[Icons]

[Cancel] [OK]

3. Select the required representation from the above Node icon.
For example: Person, Institution, Account etc.
Hover over the icon to view the definition of the node.
4. Based on your selection, the related Nodes are displayed. Select the Node, and you can choose the required color for the Node using the **Select Color** option.
5. The selected Node is displayed on the graph. To add attributes to the Node, click **Add** .
Modeler attributes are the properties of Node/Edge, and these are mandatory inputs.
For Example:
Customer Node will have properties like Customer ID, Name, Age, Gender, etc.
The **Add Node Modeler Attribute(s)** screen is displayed.

Figure 11-137 Add Node Modeler Attribute(s) screen

Add Node Modeler Attribute(s)

Required

Required

Required

6. In the **Logical Name** field, enter the logical name for the modeler attribute.
This field is mandatory.
7. Select the data type based on your requirements in the **Data Type** field drop-down.
The available data types are Text, Number, and Date. This field is mandatory.
Based on the data type selection, the following input field is displayed.
For example, the **Maximum Length** of Text field is displayed if the data type is selected as Text.

Note

Ensure the Maximum Length attribute is set to at least the field's length from which the data is to be populated.

8. Enter the data and click **OK**.

The Node attributes are displayed on the Node Details page.

Figure 11-138 Search

Node Modeler Attributes Section

+

Logical Name ⌵	Data Type ⌵	Format ⌵	Action
No data to display.			

You can add any number of modeler attributes based on your requirements by clicking on the Add icon.

9. Under the **Advanced Features** group, select the modeler attribute from the Column Identifier drop-down. All the created modeler attributes are displayed in this list.

Figure 11-139 Advanced Features

🔍 Node Details
×

> 📄 Basic Information

▾ 📄 **Advanced Features**

Column Identifier
▾

Required

> 🔄 Manage Pipeline(s)

Under the **Manage Pipeline(s)** group, you can either create a new data pipeline or attach the existing pipeline.

The application uses data pipelines to prepare filtered data which can be used to create and run scenarios. Data pipelines prepare data by selecting and joining data sources to create virtual data tables, adding derived attributes to data, running derivations on the data to determine the risk associated with the entity, and so on.

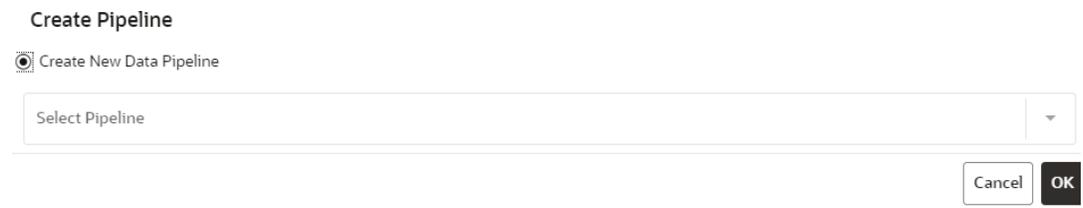
Note

The newly created data pipeline is not automatically added to the Node. You must attach the pipeline using the **Attach** drop-down option.

The selected data pipeline is displayed under the Manage Pipeline(s).

Figure 11-140 Manage Pipeline(s)

- Click **Create** drop-down and select **Data Pipeline**.
The **Create Pipeline** window is displayed.

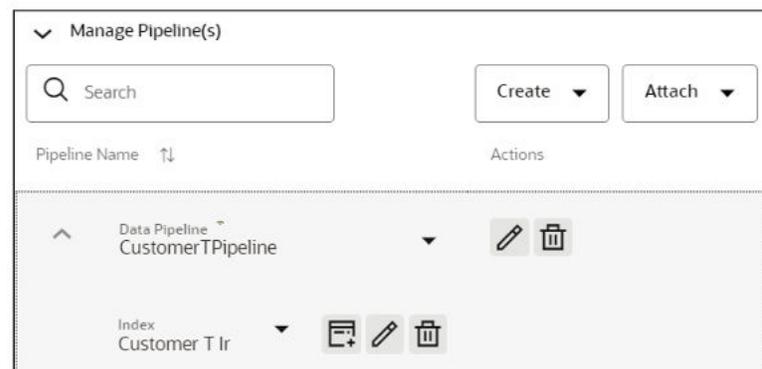
Figure 11-141 Create Pipeline

- Select the **Create New Data Pipeline** option and then click **Ok**.
The Pipeline Designer page is displayed.

- To create a data pipeline, see the [Managing Data Pipeline](#) section.
OR
- To clone from an existing data pipeline, see the Cloning from an Existing Data Pipeline section.

Note

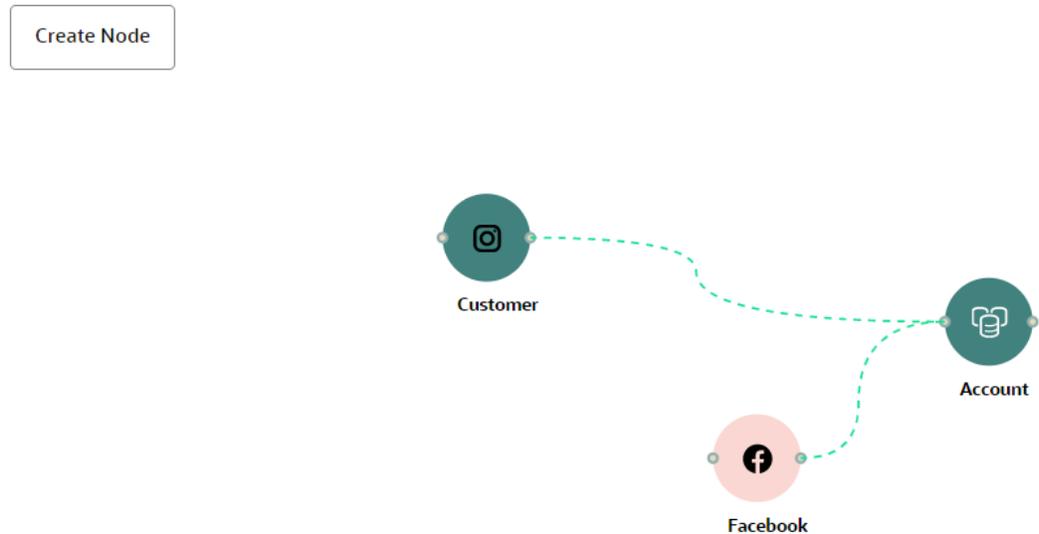
The Persist of the Data pipeline of the corresponding node should be defined with the prescript. See the Prescript Condition section for more details.

Figure 11-142 Create an Index

- To define an index for the data pipeline, see the Defining an Index section.
12. By clicking on the **Attach** drop-down, you can add any number of data pipelines to the Node-based on your requirements.
 13. Click **OK**.

The Node is created with pipeline Facebook and an account attached to it as shown in the below example.

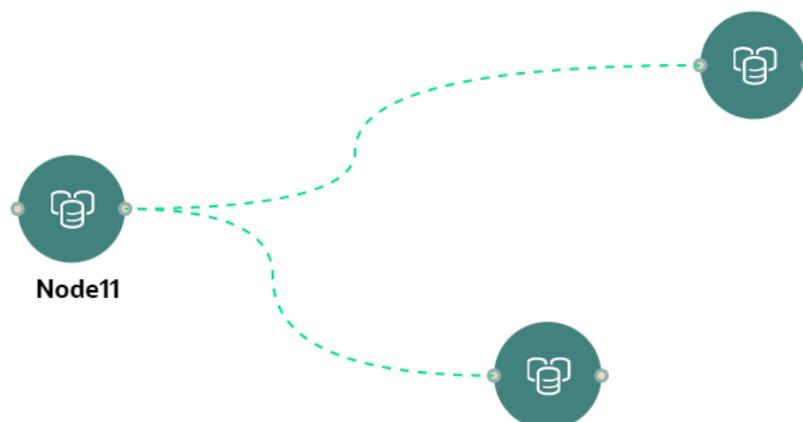
Figure 11-143 Graph with Node and Edge



Create an Edge

An Edge can be used to connect two different nodes or connect to itself. When compared to nodes, edges also capture the basic information like name, description, attributes to an edge, and possible pipelines that can be attached to an edge.

Figure 11-144 Edge



To create an edge, perform these steps:

1. Hover on the Node to create a relationship between them.
The **Edge Details** screen is displayed.

Figure 11-145 Edge Details

Edge Details

Basic Information

Logical Name Required

Logical Description Required

Edge Modeler Attributes Section Is Similarity Edge

Search

Logical Name	Data Type	Format	Action
No data to display.			

(0 of 0 items) | < 1 >

Advanced Features

Manage Pipeline(s)

Search

Create Attach

Pipeline Name	Action
No data to display.	

2. In the **Logical Name** field, enter the Node's name or click the **Setting** icon.
The **Setting** pop-up window is displayed.

Figure 11-146 Setting

Setting

Select Edge

Select Color

Cancel OK

3. Select the required edge from the options displayed and choose the required color for the edge using the **Select Color** option.

4. Click **OK**.
The selected edge is displayed on the graph pipeline.
5. To add attributes to the edge, click the **Add** icon.

Figure 11-147 Add Edge Modeler Attribute(s) screen

Add Edge Modeler Attribute(s)

Required

Required

Required

Cancel
OK

The **Add Edge Modeler Attribute(s)** screen is displayed.

6. In the **Logical Name** field, enter the logical name for the edge attribute.
This field is mandatory.
7. In the **Data Type** drop-down, select the data type based on your requirements.
The available data types are Text, Number, and Date. This field is mandatory.
Based on the data type selection, the following input field is displayed.
For example, the **Maximum Length of Text** field is displayed if the data type is selected as **Text**.
8. Enter the data and click **Ok**.
The edge attribute is displayed on the **Edge Details** page.

Figure 11-148 Search

+

Logical Name	Data Type	Format	Action
Test	Text v	Maximum Length of Text 30	

By clicking on the **Add** icon, you can add any number of edge attributes based on your requirements.

OR

Figure 11-149 Edge Details

Edge Modeler Attributes Section Is Similarity Edge

Logical Name ⌵	Data Type ⌵	Format ⌵
Edge ID	Number	22
Source ID	Text	100
Target ID	Text	100
Label	Text	100
Source	Text	100

Click the **Is Similarity Edge** toggle button the **Edge Modeler Attributes** section to add the pre-defined set of attributes available in the edge.

Similarity Edges in the graph are created by Match Rules. If the edge is defined using Match Rule (under Manage Pipeline, by selecting Match Rule from drop down), the toggle button should be enabled to make it a Similarity Edge.

- Under the **Advanced Features** group, you can specify the Edge Retention Policy.

For example, An Edge Retention Policy of 7 Days specifies that the current edge would be retained in the physicalized Graph for a period of 7 Days.

- Enter the retention count and select the retention duration from the drop-down.

The available retention durations are Days, Weeks, and Months.

Figure 11-150 Retention Durations tab

Retention Policy ?

Retention Count
5

Retention Duration
Days

- Enable the **Pluggable Attribute**, then the corresponding drop-down will contain the list of attributes of the Edge.

Pluggable edges are those edges whose data can be directly plugged in to the graph. In Financial Crime and Compliance domain, Transaction is a pluggable edge, which is generally large and is difficult to find out difference/ delta between two subsequent loads. Hence, it's data is provided for a range of dates which can be directly plugged in to the PGX memory while refreshing the graph.

Any pluggable edge should have an identifier attribute, which is of “Date” data type. Figure: Pluggable Attribute.

Figure 11-151 Pluggable Attribute

Pluggable Attribute

Is Pluggable

Pluggable Attribute Required

- Under the **Key Attributes** field, select the Column, Start, and End identifier from the respective Identifier drop-down. Without these attributes, an edge cannot be created. All the created edge attributes are displayed under this drop-down.

Figure 11-152 Key Attributes

Key Attributes

Column Identifier Required

Start Identifier Required

End Identifier Required

- Under the **Manage Pipeline(s)** group, you can either create a new data pipeline if the edge is coming from source data and Match Rules, if a similarity edge is to be created between nodes or attach the existing pipeline and Match Rules.

Figure 11-153 Manage Pipeline(s) for Edge

Manage Pipeline(s)

Search

Pipeline Name

No data to display.

Action

Create Attach

Data Pipeline

Match Rules

- Click **Create** drop-down and then select the Data Pipeline. The **Create Pipeline** window is displayed.

Figure 11-154 Create Pipeline

Create Pipeline

Create New Data Pipeline

Clone From an Existing Data Pipeline

Select Pipeline

Cancel OK

15. Select the **Create New Data Pipeline** option and then click **OK**.

The Pipeline Designer page is displayed.

- To create a data pipeline, see the Creating a New Data Pipeline section.
OR
- To clone from an existing data pipeline, see the Cloning from an Existing Data Pipeline section.

Note

Define persist of the Data pipeline of the corresponding edge with the prescript. See the Prescript Condition section for more details.

16. Click **Create** drop-down and then select the **Match Rules** from the Manage Pipeline(s) group.

The **Create Pipeline** window is displayed.

Figure 11-155 Create Pipeline



17. Select the **Create New Match Rule** option and then click **Ok**.

The Create Match Ruleset window is displayed

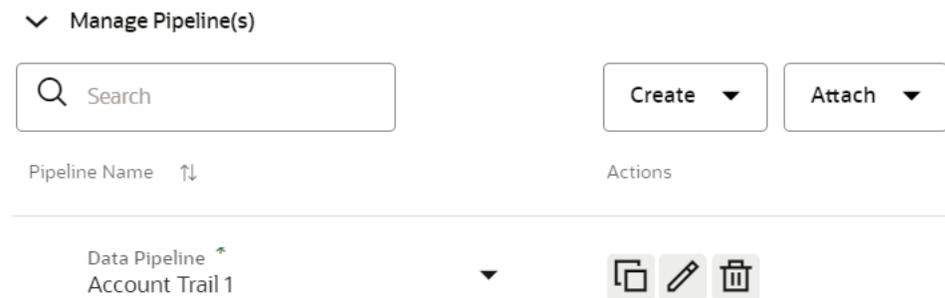
- To create a Match Rule, see the Creating Match Ruleset section.
OR
- To clone from an existing match rule, see the Cloning Ruleset (Match) section.

Note

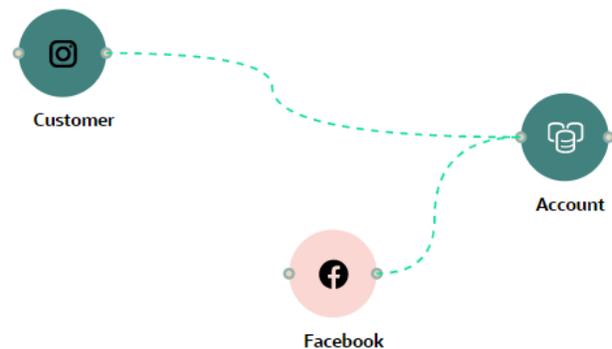
The newly created data pipeline and match rule are not automatically added to the edge. You must attach using the Attach drop-down option.

By clicking on the **Attach** drop-down, you can add any number of data pipelines/match rules to the Node-based on your requirements.

The selected data pipeline/match rule is displayed.

Figure 11-156 Create Pipeline**18. Click OK.**

An Edge is created between the Customer, Facebook, and Account as shown in the below example.

Figure 11-157 Graph page

Zoom In and Zoom Out

You can zoom in or zoom out of the graph pipeline and view the data pipelines using the diagram present in the bottom right corner of the page.

Edit a Graph Pipeline

To edit a graph pipeline, perform these steps.

1. In the **Graph Summary** page, select the graph pipeline you want to edit, click the Action icon, and select **Edit**.
2. In the **Graph Name** field, modify the graph pipeline name.
The name must be unique to a particular workspace.
3. Click the **Setting** icon.
4. In the **Graph Description** field, enter the description for the graph pipeline and click **Save**.

The following page is displayed.

Figure 11-158 Graph page

The screenshot shows a 'Graph Details' dialog box with the following elements:

- Graph Name:** Graph001
- Graph Description:** Graph001 Test Delta
- Should batch be created in read-only mode for scheduler screen ?** (Toggle switch is turned on)
- Max age of old session:**
 - Retention Count:** 7
 - Retention Duration:** Days (dropdown menu)

The Maximum Age of Old session for a graph is 7 days by default. The Maximum age of old session of 7 days specifies that graph would be retained for a period of 7 days. You can modify the description and Max Age of Old session by clicking on the **Setting** icon. If you want the batch to be created in read-only mode for scheduler screen, enable the option.

5. Click **Save**.

The user session of the Graph Pipeline will get refreshed after the set timeline.

Delete a Graph Pipeline

To delete a graph pipeline:

- On the **Graph Summary** page, select the graph pipeline you want to delete, click the **Action** icon, and select **Delete**.

To delete multiple graph pipelines, select the graph pipelines which you want to delete and click **Delete** icon in the Header.

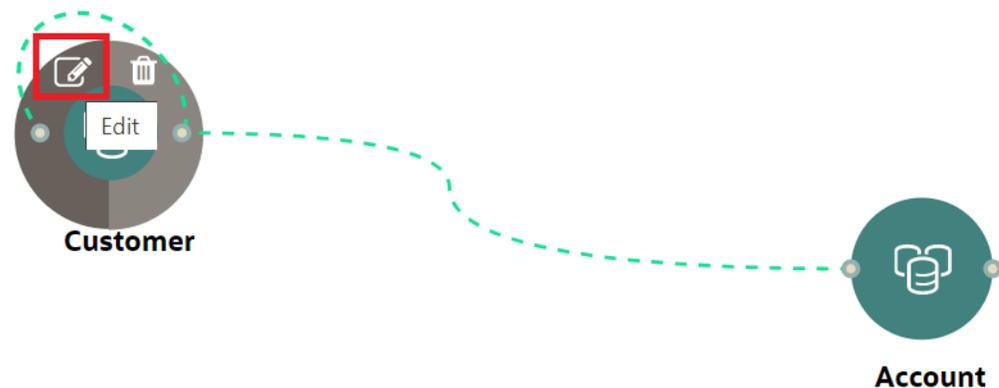
Edit a Node

To edit a node, perform these steps.

1. On the **Graph Summary** page, select the graph pipeline you want to edit, click the **Action** icon, and select **Edit**.
2. Hover over the graph pipeline and click **Edit** icon.

The **Node Details** page is displayed.

Figure 11-159 Node Details page



3. Perform the steps from 3 to 12 in the [Creating a Node](#) section.

ⓘ Note

If you delete any attribute in the Node Modeler Attribute section of the Node Details page, you must delete the same attribute in the match rule details. For example:

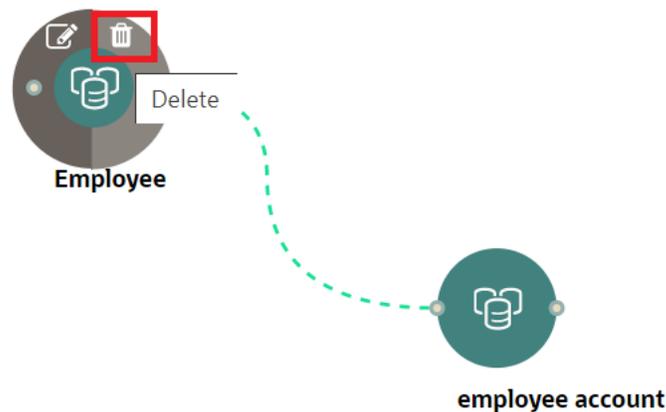
If the edge has two match rules and one is attached, and another match rule is not attached to the edge details. You need to remove attributes in both math rules for this case. To remove the deleted attribute from the Node Details page in the unattached match rule, perform the following:

- a. Navigate to **Edge Details > Manage Pipeline(s) > Attach**. Attach the new match rule, which is not attached to the node details.
- b. Click **Edit** and delete the attribute in the Ruleset Details, which is removed from the Node Details page.
- c. Click **Save**. The attribute is removed.
- d. Click **Delete** in Manage Pipeline(s) of the Edge Details page to remove the attached match rule.
- e. After saving in the Edge Details page, navigate to the Node Details page and delete the attribute again.
- f. Click **OK**. The attribute is removed from the node.

Delete a Node

To delete a node:

1. On the **Graph Summary** page, select the graph pipeline you want to edit, click the **Action** icon, and select **Edit**.
2. Hover over the graph pipeline and click **Delete** icon.

Figure 11-160 Delete Node

A confirmation dialog is displayed.

3. Click **Delete**.

The Node is deleted.

Schedule Details

Note

Currently, when you execute a Graph pipeline for the first time and navigates to the "Monitor Batch" screen, two "run ids" are generated for that batch.

To create a Graph Refresh Schedule:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click the **Graphs** icon.
This displays the graph pipeline in a table.
3. Click on the graph for which you want to create refresh schedule.
 - The Graph Model page is displayed.
4. Navigate to **Schedule Details** page and click **Add Schedule**.
The **Manage Schedule** screen is displayed.

Figure 11-161 Manage Schedule screen

Manage Schedule
✕

Basic Information

Schedule Name Required

Schedule Type

Once
 Daily
 Weekly
 Monthly
 Cron

Start Date
 02/26/2023 📅

Run Time
 12:00 AM

Select Data Pipelines

Selected Data Pipelines Required

Select Load Index

Select Load Index

5. In the **Schedule Name**, enter the desired name.
6. Select the **Schedule Type** from the options.
Based on the schedule type, the following input field is displayed.

Note

For more details and expressions of schedule type, click the **help** icon.

7. Click inside the **Select Task** field and select the tasks. You can select multiple tasks.
8. Click **Next** and add the values for the selected task, and the values are mandatory fields.

Figure 11-162 Add Optional Parameters

Add Schedule Cancel Previous Add

Selected Task(s)
TestCustomer

Optional Parameters +

Task : TestCustomer | Populates : TestCustomer | Type : datapipeline

Key	Value	
GraphDB	GraphDB	
BDATM	BDATM	
\$RUNTYPES\$	NULL	
\$BATCHRUNTYPES\$	NULL	
\$BATCHTYPES\$	NULL	

- Click the + icon to add optional parameters for the selected task.

For the Pre-configured Data pipeline, respective Key fields will be displayed. You can configure the respective key-value in the Value field as required and specified in the following table:

Table 11-32 Optional Parameters for Data Pipeline

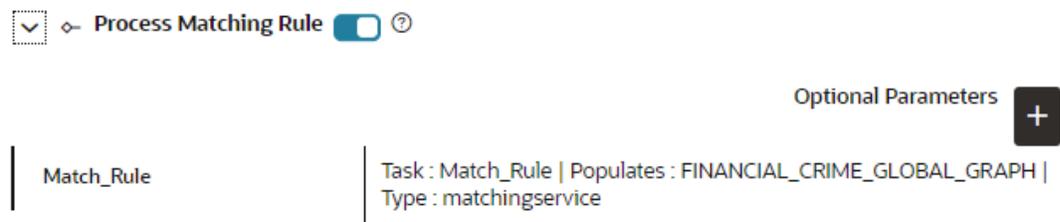
Name	Description	Default Value
\$GRAPH_SCHEMA	Oracle wallet alias of graph schema. Navigate to <OFS_COMPLIANCE_STUDIO>/wallet/ tnsnames.ora and check the Graph Schema alias name. For example, graph_schema_alias	
\$BD_SCHEMA	Oracle wallet alias of BD schema. Navigate to <OFS_COMPLIANCE_STUDIO>/wallet/ tnsnames.ora and check the BD Schema alias name. For example, bd_schema_alias	
\$RUNTYPES\$	Indicates Run type. It can be either a production batch or a test batch. If it is a production batch, set it as PROD and TEST for the test batch.	PROD

Table 11-32 (Cont.) Optional Parameters for Data Pipeline

Name	Description	Default Value
\$batchRunType\$	Indicates Batch Run type. It can be either run or re-run. If it needs to execute once, set it as RUN and RE-RUN for re-executing after failure.	RUN
\$BATCHTYPE\$	Indicates the batch type. It should be DATA for the data pipeline task.	DATA
\$JOBNAME\$	Name of the data pipeline to be executed.	

10. To add Match Rules, enable the Process Matching Rule option and click + icon to add optional parameters.

For more information on Match Rule, see the Creating Match Ruleset section.

Figure 11-163 Process Matching Rule

For the Pre-configured Match Rules, respective Key fields will be displayed. You can configure the respective key-value in the Value field as required and specified in the following table:

Table 11-33 Optional Parameter for Match Rules

Name	Description	Default Value
datasource	Oracle wallet alias of graph schema. Navigate to <OFS_COMPLIANCE_STUDIO>/wallet/ tnsnames.ora and check the Graph Schema alias name. For example, graph_schema_alias	-
loadType	Execution Load Type. Available types are full and delta. In this release, only the full mode is supported.	full
processingGroupName	Logical processing group name for the batch. You should set it as MAN.	MAN

Table 11-33 (Cont.) Optional Parameter for Match Rules

Name	Description	Default Value
basedOnPipelineId	Match rules can be executed based on pipeline Ids or ruleset Ids. The value should be true while running from the Graph.	true
runSkey	A numeric value indicates the runskey of the execution.	100
runType	Indicates matching job that can be executed as part of graph or Entity Resolution. It should be set as Graph.	graph

11. Enable the **Refresh Graph** option and click **Add** to refresh the Graph to add optional parameters.

Note

The Key and Value parameters are not required.

Figure 11-164 Refresh Graph

Note

If you do not enable the Refresh Graph option, the pipeline will add nodes and edges to the Graph Schema but does not load into PGX.

Note

The user is now able to execute/schedule a batch with SYSDATE-X days functionality.

Audit History

At any time, you can audit the Graphs from the Audit History page. This page provides the events on the Graph. This shows the information such as, when the Graph was created, who created the Graph and its status, and so on.

The sequence of actions performed on the Graph is listed in the table view and the timeline view (graphical representation).

To view the audit history of the Graph, follow these steps:

1. Click **Launch Workspace** next to corresponding Workspace to Launch Workspace to display the **MMG Dashboard** window with application configuration and model creation menu.
2. Click the **Graphs** icon.
This displays the graph pipeline in a table.
3. Click on the graph for which you want to view the audit history.
The **Graph Model** page is displayed.
4. Navigate to **Audit History** page to view the details.

Figure 11-165 Audit History

Status	Type	Name	User	Comments	Time
Created	Graph	FINANCIAL_CRIME_GLOBAL_GRAPH	MMGSAML	Graph FINANCIAL_CRIME_GLOBAL_GRAPH (0) was created by MMGSAML	2 May 2023, 17:40:12

Click for **Audit Timeline View** for Timeline View and **Audit Table View** to switch to the regular view. Click to **Refresh** to refresh the Audit History.

Figure 11-166 Audit History window Timeline View with Horizontal Time Axis



Execution History

To view the Graph Execution History:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click the **Graphs** icon.
This displays the graph pipeline in a table.

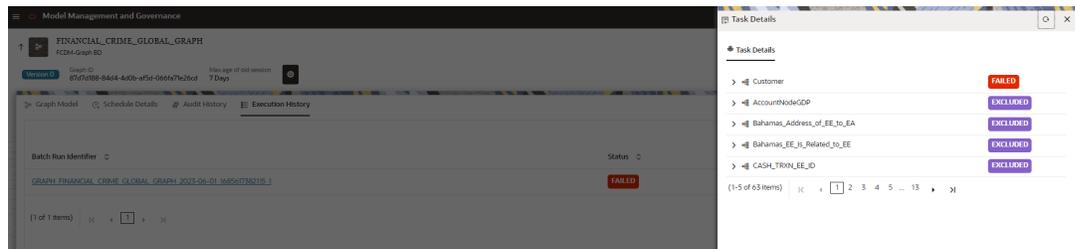
- Click on the graph for which you want to view the execution history.

The **Graph Model** page is displayed.

- Navigate to **Execution History** page to view the details.

This window displays the details such as Batch Run Identifier, Status, Start and end time of the execution. Click on the Batch Run Identifier to view the individual task details.

Figure 11-167 Execution History page



Analyses

You can analyse, search, add, and delete a notebook from this page.

Note

In the current release, features such as Edit, Clone, and Credential version are not supported.

To analyse the Graph:

- Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
- In the Mega menu, click **Modeling > Graphs**.
This displays the graph pipeline in a table.
- Click on the graph for which you want to view the execution history.
The **Graph Model** page is displayed.
- Navigate to **Analyses** page to view the details.

12

Orchestration

Orchestration covers the following:

- **Scheduler Dashboard:** Get an overview of the scheduled tasks and processes.
- **Define Batch:** Manage and configure batch definitions.
- **Define Tasks:** Create tasks, configure parameters, and set execution dependencies within a batch process.
- **Schedule Batch:** Set execution schedules for your batch processes.
- **Monitor Batch:** Track and monitor batch process executions.

Scheduler Service

The Scheduler Service automates the running of models in the Compliance Studio application.

The Scheduler Service contains a graphical user interface and a single control point for defining and monitoring background executions.

The Scheduler Service is a service in the Infrastructure system that automates behind-the-scenes work that is necessary to sustain various enterprise applications and functionalities. This automation helps the applications to control unattended background jobs.

The Scheduler Service contains a graphical user interface and a single point of control for the definition and monitoring of background executions.

Following are the concepts or terminologies in the Scheduler Service:

- **Batch:** Date and time-based execution of the background tasks based on a defined period during which the resources were available for batch processing.
- **Job:** A batch job is a piece of a program meant to meet specific and business-critical functions. The program is a RESTful API used in a batch.
- **Job Dependency:** When the batch job is submitted, it is moved to the job queue until the system is ready to process. The system process the job based on chronological order or priority in case if more jobs are required to be executed in the job queue.
- **Schedule:** Batch jobs are used to automate the tasks that require to be performed on a regular basis but don't necessarily need to occur during the day or have an employee interacted with the system are batch schedule. Jobs that happen on a regular basis are incorporated into batch schedules.

API Endpoints for Export/Import Scheduler Objects

Header Parameters for each API:

- ofs_tenant_id - Tenant ID of the Application
- ofs_service_ID - Service ID of the Application
- ofs_workspace_ID - Workspace ID of the Application

- ofs_remote_user - Used ID of the User. This parameter should be mapped to proper function.
- Content-Type: application/JSON
- 1. Batch Export API:
HTTP Method - GET
URL - /v1/rest/batch/:code
- 2. Batch Import API:
HTTP Method - POST
URL - /v1/rest/batch
Request Body:

```
{
  "objectcode": "<<BATCH_NAME>>",
  "objecttype": "BATCH",
  "objectsubtype": "",
  "overwrite": "Y",
  "objectdefinition":""
}
```
- 3. Batch Group Export API:
HTTP Method - GET
URL - v1/rest/batchgroup/:code
- 4. Batch Group Import API
HTTP Method:
POST URL- /v1/rest/BatchGroup
Request Body:

```
{
  "objectcode": "<<BATCHGROUP_NAME>>",
  "objecttype": "BATCHGROUP",
  "objectsubtype": "",
  "overwrite": "Y",
  "objectdefinition":""
}
```
- 5. Schedule Export API:
HTTP Method- GET
URL- v1/rest/schedule/:code
- 6. Schedule Import API HTTP Method - POST URL - /v1/rest/schedule
Request Body:

```
{
  "objectcode": "<<SCHEDULE_ID>>",
  "objecttype": "SCHEDULE",
```

```
"objectsubtype": "",
"overwrite": "Y",
"objectdefinition":""
}
```

Note

1. OverWrite flag can be **Y** or **N**
2. Objectdefinition will be obtained in export API same need to be paste during import in request body
3. In Schedule Export API code it is Schedule ID.
4. During import of schedule dependent batch or if BatchGroup is not migrated, the user have to use batch and BatchGroup import API to migrate updated definition in the Target environment.

Batch Import Export Utility

This utility is used to export and import batch objects like batch, batchgroups and schedules. This script works in two modes Export and Import. The first parameter of the script decides which mode it is. The Export mode requires exactly five parameters, while the Import mode requires seven parameters. Sequence of the parameters is mentioned below:

For Batch Export:

```
./batch_export_import_utility.sh EXPORT WORKSPACE LOGIN_USER OBJECT_CODE
OBJECT_TYPE
```

For Batch Import:

```
./batch_export_import_utility.sh IMPORT WORKSPACE LOGIN_USER OBJECT_CODE
OBJECT_TYPE OVERWRITE FILE_NAME
```

Table 12-1 Parameter Description

Parameter	Description	Example
WORKSPACE	Workspace from which/into which the batch object must be exported/imported.	CS
LOGIN_USER	Any valid user which has access to the system.	SAUSER
OBJECT_CODE	Code of the batch object being exported/imported. If object is of schedule type, please use schedule id instead of code.	BATCH_1
OBJECT_TYPE	Type of the object. e.g., batch, batchgroup, schedule. Must be given in lowercase.	batch
OVERWRITE	'Y' if you want to overwrite existing object while importing, 'N' if you don't want to.	Y/N

Table 12-1 (Cont.) Parameter Description

Parameter	Description	Example
FILE_NAME	The format is <workspace>/<object_type>/<object_code>	CS_batch_BATCH_1

Scheduler Service Dashboard

1. Click **Launch Workspace** next to corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. In the LHS menu, click **Scheduler Service**.

This displays the executed runs in a table.

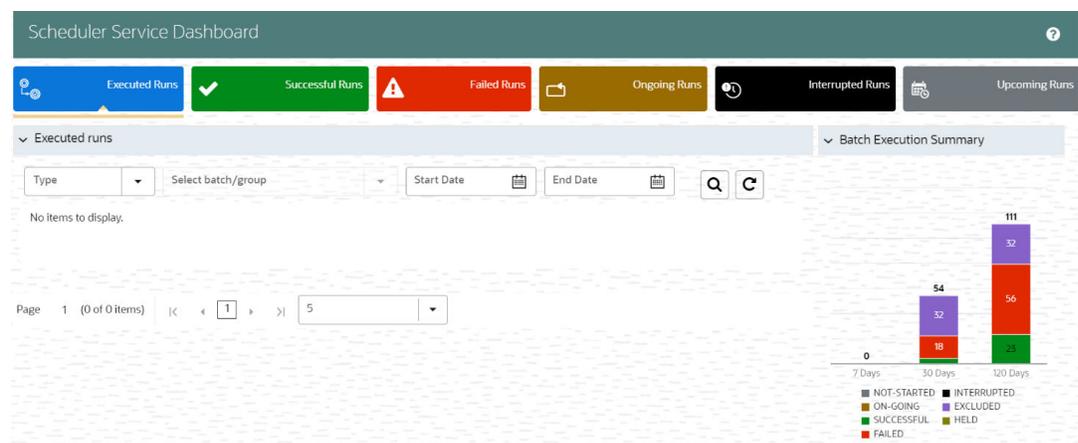
To view the demonstration of the Dashboard window, see the [Scheduler Service Introduction video](#).

In the Scheduler Service window, you can view the following details:

- The Executed Runs, Successful Runs, Failed Runs, Ongoing Runs, Interrupted Runs, and Upcoming Runs tabs. You can click the tabs to view the details of the Batches based on their status. For example, click **Ongoing Runs** to view the details of the batches that are currently running.
- The Batches that were executed within the last 7 or 30 days contain details such as Batch Name, Batch Run ID, and Run Time. Click **30 days** to view the batches that were executed within the last 30 days.
- The **Batch Execution Summary** pane displays the count of total batches executed that were executed within the last 7 days, 30 days, and 120 days. Additionally, you can see the separate count of successful batches, failed batches, interrupted batches, ongoing batches, and the batches which are yet to start, by hovering your mouse over the batches.

You can filter the executed runs based on Batch type, Batch/ group, start date and end date by selecting the corresponding options.

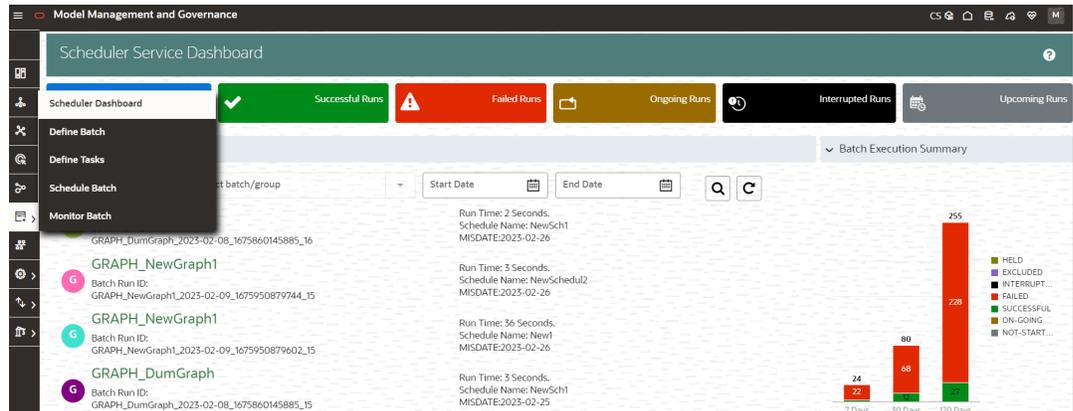
Figure 12-1 Scheduler Service Dashboard



On the **Scheduler Service** window, you can see:

- Menu Navigation: Shows the menu options available in Scheduler Service.
- Quick Actions: Quick actions can be performed by using the buttons in this section.
- User Details: View the logged in user's details, help, and set your profile preferences.

Figure 12-2 Header Details



On the left side of the window, click the **Scheduler Service** icon to see available options in the Scheduler Service.

Configure Batch and Batch Groups

This section describes about Configure Batch and Batch Groups.

Create a Batch

The **Define Batch** window displays the details of an existing batches such as Batch ID, Batch Name, Batch Description, Last Modified By and Last Modified Date. You can create, edit, copy, and delete batches. You can also create a [Batch Group](#), a list of batches that you have selected and grouped, and execute it.

To create a new batch and schedule and monitor the batch that you create:

1. Click **Define Batch** from the Header in the Dashboard window. After selecting the batch, click corresponding to the batch to proceed to create or edit tasks.
2. In the Define Batch window, click **Add**.
The **Create a New Batch** window is displayed.
3. Specify the details as described in the following table.

Table 12-2 Fields in the Create a New Batch window and their Description

Field	Description
Batch Name	<p>The Batch Name is generated based on the values provided by you.</p> <p>NOTE:</p> <ul style="list-style-type: none"> The Batch Name should be unique across the Information Domain. The Batch Name must be alphanumeric and should not start with a number. The Batch Name should not exceed 60 characters in length. The Batch Name should not contain any special characters except “_”.
Batch Description	<p>Enter a description for the Batch based on the Batch Name.</p> <p>NOTE: The Batch description should be alphanumeric. The allowed special characters are: and <blank space>, along with spaces and alpha-numeric. It should not exceed 200 characters in length.</p>
Service URL Name/ Service URL Notify on Mail	<ul style="list-style-type: none"> Select the Service URL name from the drop-down list, if it is available. The Service URL is displayed in the Service URL field. To add a new service URL, enter a name to identify it in the Service URL Name field and enter the proper URL in the Service URL field. You can give partial URL here and the remaining URL in the Task Service URL. <p>Choose this option to notify by an e-mail the batch execution status depending on the property selected. E-mail for the email notification is fetched from the IDCS configuration for the logged-in user.</p> <p>NOTE: E-mail is a mandatory field in IDCS.</p> <p>The BATCH_NOTIFY_FUNCT function has to be enabled to access this feature.</p> <p>If it is not available for the logged-in user, this field is grayed out.</p> <ul style="list-style-type: none"> Every Time: An e-mail is triggered irrespective of the batch status. Never: No e-mail will be triggered. On Error only: An e-mail is triggered only when the batch execution has failed/Error. On Interrupt only: An e-mail is triggered if the batch is successfully interrupted. <p>NOTE: Never is the default option.</p>

4. Click **Save**.

The new Batch is created and displayed in the **Define Batch** window.

Manage Batch

You can manage a batch by editing, copying, or delete them as required.

- Edit:** Use the **Edit Batch** icon to edit the Batch Description, Service URL Name, Service URL. You can also add a new batch parameter. You cannot edit seeded batches.
- Copy:** Use the **Copy Batch** option to copy a Batch that you want to clone or create instances in the system from the **Define Batch** window.

- **Delete:** Use the **Delete** icon to delete a Batch that is no longer required in the system. You cannot delete seeded batches.

Create a Batch Group

You can create a new batch group in the Define Batch window and schedule and monitor the batch group that you created.

To create a new Batch Group:

1. In the **Define Batch** window, click **Add**.

The **Create a New Batch** window is displayed.

2. Select **Batch Group** option.

3. Specify the following fields:

- Name
- Description
- Add Batches: This is a multi-select field. You can select the batches that you want to add to the group using this field.
- Notify on mail

Note

The Add Batches is a multi-select field, you can select the batches that you want to add to the group using this field.

Choose the desired option to notify by an e-mail the batch execution status depending on the property selected. E-mail for the email notification is fetched from the IDCS configuration for the logged in user.

Note

E-mail is a mandatory field in IDCS.

The BATCH_NOTIFY_FUNCT function must be enabled to access this feature.

If it is not available for the logged in user, this field is grayed out.

- Every Time: An e-mail is triggered irrespective of the batch status.
- Never: No e-mail will be triggered.
- On Error only: An e-mail is triggered only when the batch execution has failed/Error.
- On Interrupt only: An e-mail is triggered if the batch is successfully interrupted.

Note

Never is the default option.

4. Click **Save**.

The new Batch Group is created and displayed in the **Define Batch** window.

Manage Batch Group

You can manage a batch group by editing, copying, or delete them as required.

- **Edit:** Use the edit icon to edit the Batch Group name, Added Batches, and Batch Group Description.
- **Copy:** Use the copy option to copy a batch group that you want to clone or create instances in the system.
- **Delete:** Use the delete icon to delete a batch group that is no longer required in the system.

Define Batch

The Define Batch window displays the details of all existing [Batch](#), like Batch ID, Batch Name, Batch Description, Last Modified By and Last Modified Date. This window allows you to create a new, edit, copy, and delete the batches. You can also create a [Batch Group](#) and execute the Batch Group which has the list of batches that you have selected and grouped.

To navigate to the Define Batch window, click Define Batch option from the Header in the Dashboard window. After selecting the batch, you can select the **Monitor** icon corresponding to the batch to proceed to create or edit tasks.

Batch

Batch is a process of execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing.

Edit a Batch

The **Edit Batch** option allows you to edit the Batch details such as Batch Description, Service URL Name and Service URL and also add a new Batch Parameter.

Note

Seeded batches cannot be edited.

To modify a Batch, perform the following steps:

1. In the Define Batch window, click **Edit** corresponding to the Batch you want to modify.

The **Edit Batch** window is displayed.

2. Modify the required Batch details.

For more information, see Create a Batch section.

3. Click **Save**.

The edited batch is saved and displayed in the Define Batch window.

Copy a Batch

The **Copy Batch** option allows you to copy a Batch that you want to clone or create instances in the system from the **Define Batch** window.

To copy a Batch, perform the following steps:

1. In the **Define Batch** window, click **Copy Batch** corresponding to the Batch that you want to copy.
The **Copy Batch** window is displayed.
2. Specify the Batch details as you want to clone and copy the existing batch.
For more information, see [Create a Batch](#) section.
3. Click **Save**.
The copied batch is saved and displayed in the **Define Batch** window.

Delete a Batch

The Delete Batch option allows you to delete a Batch that are no longer required in the system from the Define Batch window.

Note

Seeded batches cannot be deleted.

To delete a Batch, perform the following steps:

1. From the **Define Batch** window, click **Delete** corresponding to the Batch you want to delete.
2. Click **OK** in the confirmation dialog to confirm deletion.
If the batch has any active schedules a warning is displayed.
Upon confirmation, all schedules of the batch are also deleted.

Batch Group

Batch Group is a process of grouping the batches that are required to be execute together for execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing.

Edit a Batch Group

The **Edit Batch Group** option allows you to edit the Batch Group details such as Batch Group name, Added Batches, and Batch Group Description.

To modify a Batch Group, perform the following steps:

1. In the **Define Batch** window, click **Batch Group** option to list the batch groups.
2. Click **Edit** corresponding to the Batch Group you want to modify.
The **Edit Batch** window is displayed.
3. Modify the required Batch Group details.
For more information, see [Create a Batch Group](#) section.
4. Click **Save**.
The edited batch group is saved and displayed in the **Define Batch** window.

Copy a Batch Group

The **Copy Batch** option allows you to copy a Batch that you want to clone or create instances in the system from the Define Batch window.

To copy a Batch Group, perform the following steps:

1. In the **Define Batch** window, click Batch Group option to list the batch groups.
2. Click **Copy Batch** corresponding to the Batch Group that you want to copy.

The Copy Batch window is displayed.

3. Specify the Batch Details as you want to clone and copy the existing batch.

For more information, see Create a Batch Group section.

4. Click **Save**.

The copied batch group is saved and displayed in the **Define Batch** window.

Delete a Batch Group

The **Delete Batch** option allows you to delete a Batch that are no longer required in the system from the **Define Batch** window.

To delete a Batch Group, perform the following steps:

1. From the **Define Batch** window, click **Batch Group** option to list the batch groups.
2. Click **Delete** corresponding to the Batch Group you want to delete.
3. Click **OK** in the confirmation dialog to confirm deletion.

Define Tasks

The **Define Tasks** window displays the list of tasks associated with a specific Batch definition. You can create new tasks, edit the existing tasks or delete unwanted tasks. Additionally, you can specify task precedence for each task in Task Precedence window and click the Schedule icon to Schedule the batch.

The **Define Tasks** window allows you to perform the following task operations for your batch and batch group:

- Batch
- Batch Group

Batch

Batch is a process of execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing. You can perform the following operation for the batch based on the task.

- Add a Task
- Modify a Task
- Define Task Precedence
- Delete a Task

Modify a Task

Modifying a task option allows you to modify the details of existing tasks of a Batch definition such as Task Description, Task Type, Batch Service URL and Task Service URL. You can also add a new task parameter and enable or disable already existing task parameters.

To modify a Task, perform the following steps:

1. From the **Define Task** window, select the Batch whose task details you want to modify, from the **Select** drop-down list.
2. Click **Edit** corresponding to the Task whose details you want to modify.
The **Edit Task** window is displayed.
3. Modify the required Task Details.
For more information, see Add a Task section.
4. Click **Save**.

Delete a Task

You can remove a task from a Batch definition which are no longer required in the system by deleting it from the **Define Task** window.

To delete a Task, perform the following steps:

1. From the **Define Task** window, select the Batch whose task details you want to delete from the **Select** drop-down list.
2. Click **Delete** corresponding to the Task you want to delete.
3. Click **OK** in the confirmation dialog to confirm deletion.

Batch Group

Batch Group is a process of grouping the batches that are required to be execute together for execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing. You can perform the following operation for the batch based on the task.

- Define Task Precedence

Define Task Precedence

Task Precedence indicates the execution-flow of a Batch. Task Precedence value facilitates you to determine the order in which the specific Tasks of a Batch are executed.

For example, consider a Batch consisting of 4 Task. The first 3 Task does not have a precedence defined and hence will be executed simultaneously during the Batch execution. But, Task 4 has a precedence value as task 1 which indicates that, Task 4 is executed only after Task 1 has been successfully executed.

You can set Task precedence between Tasks or define to run a Task after a set of other tasks. However, multiple tasks can be executed simultaneously, and cyclical execution of tasks is not permitted. If the precedence for a Task is not set, the Task is executed immediately on Batch execution.

To define the task precedence in the Define Task window, perform the following steps:

1. Click the **Menu** button corresponding to the task for which you want to add precedence task.

The **Task Precedence Mapping** window is displayed.

The Task Precedence option is disabled if a batch has only one task associated.

- a. Select the batch that you want to execute before the current task, from the Available Tasks pane and click **Move**. You can press Ctrl key for multiple selections.
 - b. To select all the listed batches, click **Move All**.
 - c. To remove a batch, select the task from the Selected Tasks pane and click **Remove**.
 - d. To remove all the selected batches, click **Remove All**.
2. Click **Save** to update Task Precedence in the batches.
 3. Click **Preview** to view the Precedence information.

Configure Tasks

The **Define Tasks** window displays the list of tasks associated with a specific Batch definition. You can create new tasks, edit existing tasks, or delete unwanted tasks. Additionally, you can specify task precedence for each task in the Task Precedence window and schedule the batch.

Use the **Define Tasks** window to perform task operations for your batch and batch group.

Add Task to a Batch or Batch Group

You can add new tasks to a selected Batch or Batch Group definition.

To add new task:

1. Click **Define Tasks** from the Header panel.

The **Define Task** window is displayed.

2. Select the Batch or Batch Group for which you want to add new task from the Select drop-down list.
3. Click **Add**.

The **Create a New Task** window is displayed.

When creating a task, there are multiple types of components to choose from:

Model

- **Model:** Select the Model of selected Objective. It can be any specific model of Objective or All models of Objective. If the ALL_CHAMPION is selected here, then Objectives with no Champion Model is skipped, and the Objectives with Champion Models gets executed. If CHAMPION is selected, and no Champion is present, then Model Execution gets fail.
- **Objective:** Select the Object which you want to use for execution. For more information, see the Create Objective (Folders) section. The Sub Objective is displayed with path. For example, if Test1 is Objective and Test11 is Sub Objective, and you want to use Test11 Objective for execution, then select this field as Test1/Test11.
- **Link Types:** You can set this parameter to **Yes** or **No**. If Synchronous Execution is set to **Yes**, then execution will wait for the notebook execution status. If Synchronous Execution is set to **No**, then execution will not wait for the notebook execution status, it will trigger the notebook and update task status as successful in batch monitor.
- **Synchronous Executions:** You can set this parameter to **Yes** or **No**. If Synchronous Execution is set to **Yes**, then execution will wait for the notebook execution status. If

Synchronous Execution is set to **No**, then execution will not wait for the notebook execution status, it will trigger the notebook and update task status as successful in batch monitor.

- **Optional Parameters:** This is used pass the parameters dynamically.

Populate Workspace

- **Global Filter:** Provided input will be applied as a data filter on all the tables selected for data sourcing.
 - **Table Filter:** Users can provide data filters individually on the tables here. Please note that you may provide multiple table names for the same SQL filter. Global Filters will not be applicable for those tables on which filters have been applied individually. In case the same table name is provided in more than one rows here, the filter condition will be generated as a conjunction of all the provided filters.
 - **Additional Parameters:** Additional parameters include source/target pre-scripts, fetch size, and commit size to optimize data processing and control execution behavior.
4. Enter the details as given here:

Table 12-3 Fields in the Create a New Task window and their Description

Field	Description
Task Name	<p>Enter the task name.</p> <p>NOTE:</p> <ul style="list-style-type: none"> • The Task Name must be alphanumeric and should not start with a number. • The Task Name should not exceed 60 characters in length. • The Task Name should not contain any special characters except underscore (_).
Task Description	<ul style="list-style-type: none"> • Enter the task description along with spaces and alphanumeric. No special characters are allowed in Task Description. • Words like Select From or Delete From should not be entered in the Description.
Task Type	<p>Select the task type from the drop-down list:</p> <p>Rest: Used to invoke RESTful APIs. Only POST method is supported commonly used to trigger downstream services (For example: models, graphs and so on) or workflows.</p> <p>Executable: Used to run shell scripts (.sh) or other system-level executables.</p>
Batch Service URL	<p>Select the required Batch Service URL from the drop-down list. This can be blank, and you can provide the full URL in the Task Service URL field.</p> <p>WORKSPACE_URL: Seeded batch URL base endpoint of the batch service responsible for managing and orchestrating batch or batch group executions related to workspace APIs.</p> <p>MMG_SERVICE_URL: Seeded batch URL base endpoint of the batch service responsible for managing and orchestrating batch or batch group executions related to MMG APIs (For example: Model Execution).</p> <p>CS_SERVICE_URL: Seeded batch URL base endpoint of the batch service responsible for managing and orchestrating Batch or Batch Group executions related to CS APIs.</p>

Table 12-3 (Cont.) Fields in the Create a New Task window and their Description

Field	Description
Component	These are preceded metadata types, which can be executed through Scheduler. The user is allowed to select the required components with regards to the select Bacth URL, or the user can create their own CUSTOM URLs by selecting the CUSTOM component.
Task Service URL	Enter task service URL if it is different from Batch or Batch Group Service URL. Batch URL: This is the base endpoint of the batch service responsible for managing and orchestrating batch or batch group executions. Users can append the task end points in the Define Task. There is nothing as Batch group service URL. Task Service URL: This is the endpoint for executing individual tasks (For example, REST calls). Users can either provide an end point path to be appended to the Batch URL provided in the batch definition or specify a full REST URL for independent execution.

5. From the Task Parameters pane, click **Add** to add a new Task Parameter. By default, all Batch level parameters are added and enabled as task parameters. To disable, deselect the check box corresponding to the task parameter ()
 - a. Enter the Parameter name in the **Param Name** field.
 - b. Enter the Parameter value in the **Param Value** field.
You can delete a parameter by clicking **Delete** corresponding to the parameter.
6. Click **Save**.

 **Note**

Header Parameter: This parameters are passed in the request header of the execution API to provide metadata like workspace, service ID and so on.

Manage Tasks

You can manage tasks by modifying or deleting tasks and also define task precedence.

- **Modify:** Use the edit icon to modify details of existing tasks of a Batch definition such as Task Description, Task Type, Batch Service URL, and Task Service URL. You can also add a new task parameter and enable or disable existing task parameters.
- **Delete:** Use the delete icon to remove a task from a Batch definition from the Define Task window.
- **Define Task Precedence:** Task Precedence specifies the execution-flow of a Batch. Task Precedence value facilitates you to determine the order in which the tasks of a Batch are executed.

For example, consider a Batch consisting of 4 tasks. The first 3 tasks do not have a precedence defined and hence will be executed simultaneously during the Batch execution. But, task 4 has a precedence value as task 1 which indicates that, task 4 is executed only after task 1 has been successfully executed.

You can set Task precedence between Tasks or define to run a Task after a set of other tasks. However, multiple tasks can be executed simultaneously; cyclical execution of tasks is not permitted. If the precedence for a task is not set, the task is executed immediately on execution of the Batch.

To define the task precedence in the Define Task window:

1. Click **Menu** button corresponding to the task for which you want to add a precedence task.
The **Task Precedence Mapping** window is displayed.

Note

The Task Precedence option is disabled if a batch has only one task associated.

- a. Select one or multiple tasks you want to execute before the current task, from the **Available Tasks** pane and click >. Press Ctrl key to select multiple items.

2. Click **Save**.
3. Click **Preview** to view the precedence information.

Define Task Precedence

Task Precedence specifies the execution-flow of a Batch. Task Precedence value facilitates you to determine the order in which the specific Tasks of a Batch are executed.

For example, consider a Batch consisting of 4 Task. The first 3 Task does not have a precedence defined and hence will be executed simultaneously during the Batch execution. But, Task 4 has a precedence value as task 1 which indicates that, Task 4 is executed only after Task 1 has been successfully executed.

You can set Task precedence between Tasks or define to run a Task after a set of other tasks. However, multiple tasks can be executed simultaneously, and cyclical execution of tasks is not permitted. If the precedence for a Task is not set, the Task is executed immediately on Batch execution.

To define the task precedence in the Define Task window:

1. Click the **Menu** button corresponding to the task for which you want to add precedence task.

The **Task Precedence Mapping** window is displayed.

Note

The Task Precedence option is disabled if a batch has only one task associated.

- a. Select the Task you want to execute before the current task, from the **Available Tasks** pane and click **Move**. You can press Ctrl key for multiple selections.
 - b. To select all the listed Tasks, click **Move All**.
 - c. To remove a Task, select the task from the Selected Tasks pane and click **Remove**.
 - d. To remove all the selected Tasks, click **Remove All**.
2. Click **Save** to update Task Precedence.
 3. Click **Preview** to view the Precedence information.

Exclude or Include Tasks

You can exclude tasks or include the excluded tasks during Batch Group Execution. The excluded task components are therefore executed in the normal process assuming that the excluded task have completed execution.

To exclude/include tasks, perform the following steps:

1. In the **Schedule Batch** window, click **Exclude Tasks**.
The **Select Tasks** window is displayed.
2. To exclude tasks:
 - a. Select the required task from the **Included Tasks** list and click **Move**. You can press Ctrl key for multiple selections.
 - b. To exclude all tasks, click **Move All**.
3. To include the excluded tasks:
 - a. Select the required task from the **Excluded Tasks** list and click **Remove**. You can press Ctrl key for multiple selections.
 - b. To include all excluded tasks, click **Remove All**.
4. Click **Save**.

Schedule Batch or Batch Group

Use the *Schedule Batch / Batch Group* feature to run, schedule, re-start, re-run batches in the Scheduler Service. After you upload the data in the required format into Object Storage, you must load the data into the system using the Scheduler Service. You can schedule them to run in a required pattern and view the run time status of the scheduled services using the Monitor Batch feature.

You can perform operations on batches and batch groups. Information in the following sections is applicable to both batches and batch groups. Where there are unique instructions for batches or batch groups, they are called out clearly.

Execute a Batch/Batch Group

Use the Execute batch option to run a batch/batch group instantaneously.

To execute a batch:

1. Click **Schedule Batch** from the Header panel.
The **Schedule** window is displayed.
2. Select the Batch Name from the **Select Name** drop down menu.
For example, AMLDataLoad.
You can preview the schedule of a Batch/Batch Group by clicking on Preview button.
3. Click **Execute**.
The Execution Status Dialog Box is displayed with the Batch executed successfully message.
This indicates the unique identification reference number for the batch and date of the batch execution.

4. In the **Execution Status** Dialog Box, click **Monitor** to monitor the batch.
5. If you want to exclude/include some tasks, click **Exclude Tasks**.
For more information, see [Exclude/Include Tasks](#) section.
6. If you want to hold/release some tasks, click **Hold Tasks**.
For more information, see [Hold/Release Tasks](#) section.
7. If you want to edit the dynamic parameters of the batch, click **Edit Dynamic Parameters** .
For more information, see [Edit Dynamic Parameters](#) section.

Schedule a Batch/Batch Group

You can schedule a Batch to run just for Once, Daily, Weekly, Monthly or a Cron Expression for scheduling the batches. You can also have a user defined schedule to schedule and run a batch/batch group.

Schedule Once

To schedule a Batch to run once:

1. Click **Schedule Batch** from the Header panel.
The **Schedule Batch** window is displayed.
2. In the Schedule Batch window, click **Once**.
3. Select the Batch or Batch Name you want to schedule for once from the **Select** list.
4. Enter a **Schedule Name**.
5. Click the date and time picker icons and select the date and time when you want to run the batch.
6. Click **Schedule**.

Schedule Daily

To schedule a Batch to run daily:

1. In the **Schedule Batch** window, click **Daily**.
2. Select the Batch or Batch Name you want to schedule daily from the **Select** list.
3. Enter a **Schedule Name**.
4. Click the date and time picker icons and select the date and time when you want to run the batch.
5. Click **Schedule**.

Schedule Weekly

To schedule a Batch to run weekly:

1. In the **Schedule Batch** window, click **Weekly**.
2. Select the batch or batch name you want to schedule weekly from the **Select** list.
3. Enter a **Schedule Name**.
4. Click the date and time picker icons to select the start date, end date, and time when you want to run the batch.

5. Select the days of the week when you want to run the batch from the **Select Days of the Week** list.
6. Click **Schedule**.

Schedule Monthly

To schedule a Batch to run monthly:

1. In the **Schedule Batch** window, click **Monthly**.
2. Select the Batch or Batch Name you want to schedule monthly from the Select list.
3. Enter a **Schedule Name**.
4. Click the date and time picker icons to select the start date, end date, and time when you want to run the batch.
5. Select the days of the week when you want to run the batch from the **Select Days of the Week** list.
6. Click **Schedule**.

Schedule Cron Expression

To run a Batch in a user-defined schedule, you can have custom schedule with the help of Cron Expression. A Cron Expression is a string comprised of 6 or 7 fields separated by white space. Fields can contain any of the allowed values, along with various combinations of the allowed special characters for that field. For more information, click the information icon.

To create a customized schedule for a batch using a Cron Expression:

1. In the **Schedule Batch** window, click **Cron Expression**.
2. Select the batch or batch name you want to schedule from the Select list.
3. Enter a **Schedule Name**.
4. Enter the Cron Expression for your schedule. For more information about the Cron Expression, click the **Information** icon.
5. Click **Schedule**.

Pause a Batch

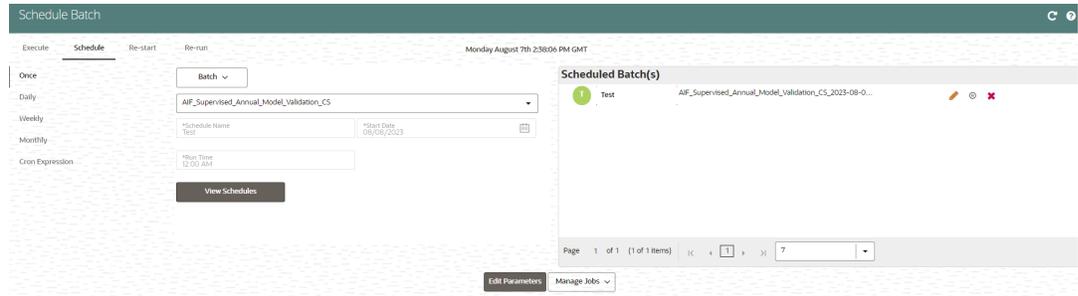
You can pause a batch that has been executed. User can click on pause button and can pause schedule for between scheduled dates. Time is not supported for pause activity, if any schedule is paused for current date, it will be paused immediately. If pause start date is passed, user cannot change pause start date, rest fields can be manipulated and will be valid one minute before end of day.

If pause is expired, user cannot edit and delete but will be shown in UI.

To pause a batch:

1. From the Header panel, click **Schedule Batch** and then click the **Schedule** tab.
2. Select the batch you want to pause from the list.
3. For the batch you have selected to pause, select the scheduling type (Once, Daily, Weekly, Monthly, or Cron Expression).
4. Click **View Schedules**.

The scheduled batches are displayed on the right pane.

Figure 12-3 Scheduled Batches

5. Click the pause icon for the batch you want to pause.
6. In the **Manage Pause** screen, click **Add Pause**.
7. Select the **Pause Start** and **End Date**, add comments if required, and click +.
The Pausing schedule appears in the Pause Summary.
8. Repeat the steps if you want to add multiple pausing schedules for the same batch.

Re-start a Batch/Batch Group

You can restart a Batch which has not been executed successfully or which has been explicitly interrupted, or cancelled, or put on hold during the execution process. By restarting a Batch, you can continue Batch execution directly from the point of interruption or failure and complete executing the remaining tasks.

To re-start a batch:

1. Click **Schedule Batch** from the Header panel.
The **Schedule** window is displayed.
2. From the **Schedule** window, select **Re-start** tab.
3. Select the Batch Group you want to re-start from the Select Name drop down menu.
4. Select the Batch Run ID.
5. Click **Re-start**.

Re-run a Batch/Batch Group

You can re-run a Batch which has previously been executed. Rerun Batch facilitates you to run the Batch irrespective of the previous execution state. A new Batch Run ID is generated during the Rerun process and the Batch is executed as similar to the new Batch Run.

To re-run a batch:

1. Click **Schedule Batch** from the Header panel.
The **Schedule Batch** window is displayed.
2. In the **Schedule Batch** window, select **Re-run** tab.
3. Select the Batch or Batch Name you want to re-run from the **Select Name** list.
4. Select the **Batch Run ID**.
5. Click **Re-run**.

Edit Dynamic Parameters

Dynamic Parameters facilitate modification of dynamic parameters for the batch. You can change the param value and save the changes to the batch. The **Edit Dynamic Parameters** option is available in all the tab in the *Schedule Batch* window.

To edit the dynamic parameters for a batch/batch group:

1. In the **Schedule Batch** window, click **Edit Dynamic Parameters**.
2. In the **Edit Dynamic Params** window, modify the values as required.
3. Use the **SYSDATE-x** option to reference relative dates execute or schedule a batch. In the **Edit Dynamic Parameter** window, enter a value, for example; *SYSDATE-2*, to represent two days before the execution date/time. You must ensure that the value for x is a non-negative integer
4. Click **Save**.

The modified parameters are applied to the Batch.

Task Definitions of a Batch/Batch Group

You can modify the task definition state in the Batch Execution window to exclude or hold the defined task in a Batch from execution. The excluded tasks are therefore assumed to have completed execution and get excluded during the Batch Run.

While executing or scheduling a batch group from the Schedule Batch window, you can:

- Exclude a task or include the excluded task.
- Hold a task or release the held task.

Exclude or Include Tasks

You can exclude tasks or include the excluded tasks during Batch/Batch Group Execution. The excluded task components are executed in the normal process assuming that the excluded tasks have completed execution.

To exclude/include tasks:

1. In the **Schedule Batch** window, click **Exclude Tasks**.
The **Select Tasks** window is displayed.
2. To exclude tasks:
 - a. Select one or multiple tasks from the Included Tasks list and click > or >> respectively. Press the Ctrl key and click to select multiple items.
3. To include the excluded tasks:
 - a. Select one or multiple tasks from the **Excluded Tasks** list and click < or << respectively. Press the Ctrl key and click to select multiple items.
4. Click **Save**.

Hold or Release Tasks

You can hold tasks or release the held tasks during Batch Group Execution. The tasks which are on hold along with the defined components are skipped during execution. However, at least one task should be available in a Batch without being held or excluded for Batch execution.

To hold/release tasks:

1. In the **Schedule Batch** window, click **Hold Tasks**.
The **Select Tasks** window is displayed.
2. To hold tasks:
 - a. Select the required task from the **Released Tasks** list and click **Move**. You can press Ctrl key for multiple selections.
 - b. To hold all tasks, click **Move All**.
3. To release held tasks:
 - a. Select the required task from the **Held Tasks** list and click **Remove**. You can press Ctrl key for multiple selections.
 - b. To release all held tasks, click **Remove All**.
4. Click **Save**.

Pre-Conditions for a Batch Group

You can schedule the batches and set the pre-conditions within a Batch group with frequency as Weekly, Monthly, or based on an Interval for scheduling the batches. The batch that satisfies the configured pre-conditions are executed as part of the schedule.

Note

Pre-Conditions can only be applied when using the Schedule option in the Schedule Batch window.

Weekly

To set the pre-conditions to the batches in a batch group weekly, perform the following steps:

1. In the **Schedule Batch** window, you can select either Once, Daily, Weekly, Monthly, or Cron Expression option based on the schedule that you want to run the batch group.
2. Select the Batch Group you want from the Select drop down menu.
3. Enter a Schedule Name.
4. Specify the other details displayed when you are selecting Once, Daily, Weekly, Monthly, or Cron Expression.
5. Click Pre-Conditions.
6. In the Pre-Conditions window, specify the Batch from the drop down and from the Frequency drop down and select Weekly.
7. Select the days from the Select Days drop down that you want to schedule the batch run within the selected week.
8. Click **Add** to add the specified entry.

Note

Pre-Conditions can be added only to one batch at a time.

9. Click **Save**.
The batch is executed based on the configured pre-conditions.

Monthly

To set the pre-conditions to the batches in a batch group monthly, perform the following steps:

1. In the **Schedule Batch** window, you can select either Once, Daily, Weekly, Monthly, or Cron Expression option based on the schedule that you want to run the batch group.
2. Select the Batch Group you want from the Select drop down menu.
3. Enter a Schedule Name.
4. Specify the other details displayed when you are selecting Once, Daily, Weekly, Monthly, or Cron Expression.
5. Click Pre-Conditions.
6. In the Pre-Conditions window, specify the Batch from the drop down and from the Frequency drop down and select Monthly.
7. Select the days from the Select Days drop down that you want to schedule the batch run within the selected week.
8. Click **Add** to add the specified entry.

Note

Pre-Conditions can be added only to one batch at a time.

9. Click **Save**.
The batch is executed based on the configured pre-conditions.

Interval

To set the pre-conditions to the batches in a batch group based on an interval, perform the following steps:

1. In the **Schedule Batch** window, you can select either Once, Daily, Weekly, Monthly, or Cron Expression option based on the schedule that you want to run the batch group.
2. Select the Batch Group you want from the Select drop down menu.
3. Enter a Schedule Name.
4. Specify the other details displayed when you are selecting Once, Daily, Weekly, Monthly, or Cron Expression.
5. Click Pre-Conditions.
6. In the Pre-Conditions window, specify the Batch from the drop down and from the Frequency drop down and select Interval.
7. Select the interval from the Custom Recurrence (Repeat every) Days drop down that you want to schedule the batch run within the selected week.

Note

The Custom Recurrence can be set maximum to 60 days.

8. Click **Add** to add the specified entry.

Note

Pre-Conditions can be added only to one batch at a time.

9. Click **Save**.
10. The batch is executed based on the configured pre-conditions.

Edit Dynamic Parameters

Dynamic Parameters facilitates you to the modify the dynamic parameters for the batch. You can change the param value from the Edit Dynamic Params window and save the changes to the Batch. The Edit Dynamic Parameters option is available in all the tab in the **Schedule Batch** window.

To edit the dynamic parameters for a batch group, perform the following steps:

1. In the **Schedule Batch** window, click **Edit Dynamic Parameters**.
The **Edit Dynamic Params** window is displayed.
2. In the **Edit Dynamic Params** window, modify the values as required.
3. Click **Save**.
The modified parameters are applied to the Batch.

Accessing Scheduler Optional Parameters Inside Studio Paragraphs

The Scheduler Service automates the running of models in the application. The Scheduler Service contains a graphical user interface and a single control point for defining and monitoring background executions.

For more details on the Scheduler Service, see [Scheduler Service](#) section.

To access the optional parameters passed from the Scheduler during execution of any models, run the following script:

```
To print
print('BATCHRUNID value is : ${BATCHRUNID$}')
print('TASKID value is : ${TASKID$}')
print('FICMISDATE value is : ${FICMISDATE$}')
```

To access the optional parameters passed during execution of any models, run the following scripts:

```
%python
print('threshold value is : ${threshold}')
```

Note

Threshold is the optional parameter passed during the model execution.

From scheduler task/edit dynamic parameters, the aparameter set to a model can be passed in optional parameter textbox as parameterSets=Set1 where parameterSets is the key for paramter sets and Set1 is the actual set value.

From EICS it is as below:

```
./EICSchedulerService.sh -o 1 -u mmgadmin -w CS -c ofsauser -s secret -x 232 -
b BatchModel -t rest -v '{"batchParams":
{"$FICMISDATE$": "2021-11-30", "$BATCHRUNID$": "BATCHRUNID"}, "taskRuntimeParams":
{"Task1": {"optionalparams": "parameterSets=Set1"}, "Task2":
{"optionalparams": "parameterSets=Set2"} }
```

Monitor Batch or Batch Groups

Use the Monitor Batch feature to view the status of executed batch/batch group along with the task's details. You can track the issues if any, on regular intervals and ensure smoother batch/batch group execution. A visual representation as well as tabular view of the status of each task in the batch/batch group is available.

You can monitor operations for batches and batch groups using the Monitor Batch feature.

To monitor a batch/batch group:

1. Click **Monitor Batch** from the Header panel.
The **Monitor** window is displayed.
2. Select the batch/batch group from the **Select** drop-down and then select the **Batch Run ID** from the **Run ID** list.
3. Click **Start Monitor**.
The result is displayed in **Visualization** and **List View** tabs.
 - On the Visualization tab are charts and in List View are details in a tabular format with the following details:
 - a. i. Batch Status: Displays the batch status—NOT-STARTED, ON-GOING, SUCCESSFUL, FAILED, INTERRUPTED, EXCLUDED, HELD, and UNDEFINED.
 - b. Batch Start Time: The batch start time.
 - c. Batch End Time: The batch end time.
 - d. Task Details: Mouseover the task to display task status and additional details.
 - e. More Information: The message returned by the Rest Service.
4. Select **Stop Monitor** if you wish to stop monitoring. You can also specify the Start and Stop Monitor options along with refresh interval by providing the Refresh every seconds and minutes information.

Note

- You can select the refresh interval and the duration of the auto refresh. The default refresh interval is 5 seconds and default duration 5 minutes. That is, data is refreshed every 5 seconds for the next 5 minutes.
- The interval input range must be between 5 to 60 seconds and the duration input range between 5 to 180 minutes.
- You can use the **Stop Monitor** button to stop the auto refresh.

5. To restart, rerun, or interrupt the monitoring, select the Restart, Rerun, or Interrupt buttons, respectively.

- To view log information, click the log icon in the **List View** tab. The Log Viewer window appears displaying the log details.
- Click the download icon to download the log, or click the close icon to close the log viewer.

Note

Downloads PDF button has been introduced in the batch monitor.
Downloaded PDF is available in: <ftpsharepath>printservice/pdfs/
scheduler/<DATE>

Batch Group

Batch Group is a process of grouping the batches that are required to be execute together for execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing.

To monitor a batch group:

- Click **Monitor Batch** from the Header panel.
The **Monitor** window is displayed.
- Select the **Batch Group from the Select** drop-down and then select the **Batch Run ID from the Run ID** drop-down.
- Click **Start Monitor**.

The result is displayed in Visualization and List View tabs. Details of these tabs are as follows:

- The Visualization tab displays the details in the form of a chart represented with the following details:
 - Batch Status: Displays the batch status, the different batch status are NOT-STARTED, ON-GOING, SUCCESSFUL, FAILED, INTERRUPTED, EXCLUDED, HELD, and UNDEFINED.
 - Batch Start Time: Displays the batch start time details.
 - Batch End Time: Displays the batch end time details.
 - Batch Details: Mouseover the task to display its status and details.
 - The **List View** tab displays the details in a tabular form with the following details:
 - Batch Status: Displays the batch status, the different batch status are NOT-STARTED, ON-GOING, SUCCESSFUL, FAILED, INTERRUPTED, EXCLUDED, HELD, and UNDEFINED.
 - Batch Start Time: Displays the batch start time details.
 - Batch End Time: Displays the batch end time details.
 - Batch Details: Mouseover the task to display its status and details.
 - More Information: The message returned by the Rest Service.
- If you wish to stop the monitoring, select Stop Monitor. You can also specify the Start and Stop Monitor options along with refresh interval in the Refresh every seconds and minutes fields.

Note

- You can select the refresh interval and the duration for the auto refresh. The refresh interval is defaulted to 5 seconds and duration is defaulted to 5 minutes. That is the refresh happens every 5 seconds for next 5 minutes.
- Range of interval input must be between 5 to 60 seconds and range of duration.
- Input should be between 5 to 180 minutes.
- You can use the Stop Monitor button to stop the auto refresh.

5. To restart the Batch Group, select **Restart**.
6. To rerun the Batch Group, select **Rerun**.
7. To interrupt the Batch Group, select **Interrupt**.

To view the log information about the batch group, click **View Log** in the **List View** tab.

Scheduler Configuration

Scheduler Configuration UI allows you to manage the scheduler service configuration parameters. It also allows you to map specific user to batch and batch groups for email notifications.

To access the Scheduler Configuration UI in the Scheduler Service page, follow these steps:

1. Go to the Home page of the application.
2. Navigate to **Batch Administration** in the left Navigation pane.
3. Under **Batch Administration**, click on **Scheduler Configuration**.

Note

Ensure you have the BATCH_ADMIN function code to access the Scheduler Configuration page.

Batch to User Configuration

Batch to User Configuration menu allows you to map users to specific batch/batch group processes. This assignment ensures that emails are sent only to the specific users associated with each batch.

To access the Batch to User Configuration page, select **Batch to User Configuration** in the Scheduler Configuration UI.

Note

Click the **eye** icon on the **Batch to User Configuration** tile to view the page.

To search for a specific batch/batch group, enter the keywords in the Search field and click **Search**. You can search based on Batch Name, Batch Code, and Batch type.

Note

The system automatically sends an email to all users mapped to the BATCH_NOTIFY_FUNT function and BATCH_NOTIFY_ROLE role. If a batch/batch group is mapped to a user (and their email ID), the batch email notifications will be sent only to that configured batch user. If no batch-to-user configuration is provided or updated, email notifications will be sent to users with the above function and role.

Perform the following steps to add specific user(s) to the batch/batch group:

1. In Batch to User Configuration menu, click **Add** to add new batch/batch groups.
2. Select the batch type from the dropdown menu.
3. Select the required batch/batch group from the dropdown menu.
4. Select the required users from the dropdown menu.
5. Click **Create**, the *Batch User mapping is created successfully* message is displayed.

Action menu

Batch to user mapping configuration page lists all the batch/batch groups which are mapped to specific user(s).

1. Select the desired batch/batch group from the Batch to user mapping configuration page. Click **Action Menu** to view, modify, or delete batch/batch groups.
 - a. **View**
Clicking **View** allows users to see detailed information on the batch/batch group user mapping.
 - b. **Edit**
Edit the batch/batch group to user mapping configuration. You can either update the user details or remove the user.
 - c. **Delete**
The **Delete** option allows you to remove the user to batch/batch group mapping from the system.

General Configuration

The **General Configuration menu** allows you to configure the parameters related to a scheduler service.

1. Under **Scheduler Configuration** menu, select **General Configuration**.

Note

Click the **eye** icon on the **General Configuration** tile to view the page.

2. In Scheduler Service General Configuration Screen, click **Edit** to modify the configuration settings.
3. Modify the required details, refer to the **General configuration parameters** table below.

Table 12-4 General configuration parameters

Parameter	Description	Default Value
Enable/disable general email notification	Enable/disable general email notifications to automatically alert users when a batch job is successful, failed, or is interrupted.	Enable
Enable/disable in-app notification	Enable/disable in-app notifications to alert users within the application when a batch job is successful, failed, or is interrupted.	Enable
Enable/disable notification for threshold email	Enable/disable email notifications triggered when a batch execution exceeds its expected execution time threshold. For example, if a batch typically completes in 1 hour but now takes longer, an email is sent as its taking longer than expected time to complete.	Enable
Threshold email time notification percentage criteria	<p>Defines the additional execution time (in percentage) allowed beyond the last successful batch completion time before sending a threshold notification email.</p> <p>For example: If a batch previously completed successfully in 1 hour and the threshold is set to 20%, a threshold notification will be triggered if the batch exceeds 1 hour and 12 minutes (i.e., 60 minutes + 20%).</p> <p>If multiple notifications are allowed (as per the Threshold email Notify Limit), the time for subsequent notifications will be calculated from the last notification time, adding the same threshold percentage again.</p> <p>For instance, the next notification would be triggered after 20% of 72 minutes (i.e., 86.4 minutes), and so on.</p>	20

Table 12-4 (Cont.) General configuration parameters

Parameter	Description	Default Value
Threshold email Notify Limit	Defines the maximum number of email notifications that can be sent when a batch exceeds its execution time threshold during a single run.	5
	<div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>Note</p> <p>If general email notification is disabled and only if the threshold email notification is enabled, scheduler sends only the threshold email.</p> </div>	
Batch to User mapping upper limit	Specifies the maximum number of users that can be mapped to a batch/batch group. You can configure a maximum of five user mappings per batch or batch group.	10

4. Click **Save** to save the modified configurations.
OR

Click **Cancel** to discard the changes and revert to the previous settings.

External Interface Component Scheduler Service

Use the **EICSchedulerService.sh** utility located in the `<installed path> mmg-home/bin` folder to perform basic scheduler operations.

Prerequisites:

- Ensure public and private keys are present in the conf folder and the corresponding path is present in `application.properties` of mmg UI service.
- You should be aware of the `clientId/secret` which will be used for basic authentication. (Currently, these values are stored in `application.properties` as `token.clientid` and `token.secret`).

Table 12-5 List of possible flagged arguments to be passed while executing EICSchedulerService.sh

Flag	Description	Comments
-o	Operation Type: The operation the user wants to perform.	A mandatory argument for all type of requests. Possible Values are: 1 – To trigger an object 2 – To get the current status of a run. 3 – To re-start an execution. 4 – To re-run an execution. 5 – To interrupt an execution.
-u	ofs_remote_user	A mandatory argument for all the type of requests
-w	Workspace Id	A mandatory argument for all the type of requests
-c	Client Id for authentication	A mandatory argument for all the type of requests Currently, in mmg-ui application.properties the parameter token.clientid is set to ofsuser
-s	Client Secret for authentication	A mandatory argument for all the type of requests Currently, in mmg-ui application.properties the parameter token.secret is set to secret
-b	Object Name	Batch or Batch Group Name.
-t	Object Type	For Batch, it is set to rest and for Batch Group, it is set to group.
-r	Batch Execution Id	Execution ID of the Batch or Batch Group.
-x	External Unique Id	The unique ID for every execution. Mandatory for execute and rerun operations.
-i	Included Jobs	In case of trigger, included jobs can be provided as comma separated values using this flag.
-e	Excluded Jobs	In case of trigger, excluded jobs can be provided as comma separated values using this flag.
-h	Held Jobs	In case of trigger, held jobs can be provided as comma separated values using this flag.
-v	Dynamic parameter list	This is an optional parameter to support passing of dynamic runtime parameters for Batch/ Task.

Table 12-6 List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
Trigger	<ul style="list-style-type: none"> • Execute batch “batch1”: <ul style="list-style-type: none"> – Syntax: <code>. / EICSchedulerService.sh -o 1 -u < ofs_remove_user > -w < Workspace Id > -c < Client Id for authentication > -s < Client Secret for authentication > -x < External Unique Id > -b < Object Name > -t < Object Type ></code> – Example: <code>./ EICSchedulerService.sh -o 1 -u scheduler -w WS1 -c ofsouser -s secret -x batch1_1001 -b batch1 -t rests</code> • Execute batch “batch1” with the Dynamic Parameter list 	To execute a batch or a batch group	-4	Invalid arguments passed in request/not enough params in Request body
			-5	Invalid Request Headers
			-9	Duplicate External Unique Id
			-3	Batch or Batchgroup does not exist.
			-6	No executable job is present. Batch with no task.
			0	Batch or BatchGroup execution is success

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
	<ul style="list-style-type: none"> - Syntax: . / EICSched ulerServ icerService.sh - o 1 -u < ofs_remo te_user > -w < Workspac e Id > - c < Client Id for authenti cation > -s < Client Secret for authenti cation > -x < External Unique Id > -b < Object Name > - t < Object Type> -v <Dynamic Runtime paramete r list> - Example: ./ EICSched ulerServ icerService.sh - o 1 -u mngadmin -w CS -c ofsouser -s secret - x 232 -b BatchMod el -t rest -v '{"batch Params": {"\$FICMI 		-1	Batch or BatchGroup execution is failed.

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
		<pre> SDATE\$: "2021-11 -30", "\$B ATCHRUNI D\$": "BAT CHRUNID" , "param_ 1": "qwer t", "para m_2": "ab cde"}, "t askRunti meParams ": { "Task1" : { "p1": "1 1102", "p 2": "2220 2", "abc" : "123342 ", "extra P": "wert 2y"}, "Ta sk2": { "abc": " 1233", "p 3": "3333 ", "extP2 ": "0983" }}}' </pre>		

i Note

The Task / Batch p

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
		<p> a r a m e t e r s s h o u l d b e t h e a c t u a l p a r a m e t e r k e y .</p> <ul style="list-style-type: none"> • Execute batch “batch1” with “task1” and “task2” included: <ul style="list-style-type: none"> – Syntax: . / EICSched ulerServ ice.sh - o 1 -u < ofs_remo te_user > -w < Workspac 		

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
		e Id > - c < Client Id for authenti cation > -s < Client Secret for authenti cation > -x < External Unique Id > -b < Object Name > - t < Object Type> -I < Included Jobs with comma separate d values>		
		- Example: ./ EICSched ulerServ ice.sh - o 1 -u scheduse r -w WS1 -c ofsauser -s secret - x batch1_1 001 -b batch1 - t rest - i task1,ta sk2		
		• Execute batch "batch1" with "task1" as excluded and		

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
	<p>"task2" as held:</p> <ul style="list-style-type: none"> - Syntax: ./EICSchedulerService.sh -o 1 -u <ofs_remote_user> -w <Workspace Id > -c <Client Id for authentication > -s <Client Secret for authentication > -x <External Unique Id > -b <Object Name > -t <Object Type> -e <Excluded Jobs> -h <Held Jobs> - Example: ./EICSchedulerService.sh -o 1 -u scheduler -w WS1 -c ofsauser -s secret -x batch1_1001 -b batch1 - 			

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
	t rest - e task1 -h task2			
Status	<ul style="list-style-type: none"> • Get status of execution having runid = MMG_R1_202 2-09-07_16625 57327886_1: <ul style="list-style-type: none"> – Syntax: ./EICSchedulerService.sh -o 2-u <ofs_remove_user> -w <Workspace Id> -c <ClientId for authentication> -s <ClientSecret for authentication> -r <BatchExecution Id> – Example: ./EICSchedulerService.sh -o 2 -u scheduser -w WS1 -c ofsouser -s secret -r MMG_R1_2022-09-07_1662557327886_1 	To check the status	-3	Batch or Batchgroup Run Id does not exist.
			-4	Invalid arguments passed in request/not enough params in Request body
			-5	Invalid Request Headers
			0	Batch or batchgroup status is success
			-1	Batch or batchgroup status is failed or API failed internally to return valid response status code.
			-2	Batch or batchgroup status is interrupted.
			1	Batch or batchgroup status is not started.
			2	Batch or batchgroup status is ongoing.
			5	Batch or batchgroup status is held.

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
Restart	<ul style="list-style-type: none"> • Restart execution for batch "MMG_R1" having runid = MMG_R1_2022-09-07_1662557327886_1: – Syntax: ./EICSchedulerService.sh -o 3 -u <ofs_remote_user> -w <Workspace Id> -c <Client Id for authentication> -s <Client Secret for authentication> -b <ObjectName> -r <Batch Execution Id> – Example: ./EICSchedulerService.sh -o 3 -u scheduler -w WS1 -c ofsouser -s secret -b MMG_R1 -r MMG_R1_2022-09-07_1662557327886_1 	To restart failed execution	-4	Invalid arguments passed in request/not enough params in Request body
			-5	Invalid Request Headers
			-3	Batch or Batchgroup Run Id does not exist.
			-6	No executable job is present. Batch with no task.
			0	Batch or BatchGroup restart is success
			-1	Batch or BatchGroup restart is failed.

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
Rerun	<ul style="list-style-type: none"> • Rerun execution for batch "batch1" having runid = batch1 -r batch1_2022-09-01_1662011065557_1: <ul style="list-style-type: none"> – Syntax: ./EICSchedulerService.sh -o 4 -u <ofs_remove_user> -w <Workspace Id> -c <ClientId for authentication> -s <ClientSecret for authentication> -x <ExternalUnique Id> -b <ObjectName> -r <BatchExecution Id> – Example: ./EICSchedulerService.sh -o 4 -u scheduser -w WS1 -c ofsouser -s secret -x batch1_1002 -b 	To rerun the batch execution	-4	Invalid arguments passed in request/not enough params in Request body
			-5	Invalid Request Headers
			-3	Batch or BatchgroupId or Run Id does not exist.
			0	Batch or BatchGroup rerun is success
			-1	Batch or BatchGroup rerun is failed.

Table 12-6 (Cont.) List of possible exit codes returned by "EICSchedulerService.sh"

API	Syntax	Description	Status Code	Code Description
	batch1 - r batch1_2 022-09-0 1_166201 1065557_ 1		-9	Duplicate External Unique Id
Interrupt	<ul style="list-style-type: none"> • Interrupt execution for batch "batch1" having runid = batch1_2022-09-07_1662530601924_1: <ul style="list-style-type: none"> – Syntax: ./EICSchedulerService.sh -o 5 -u <ofs_remove_user> -w <Workspace Id> -c <ClientId for authentication> -s <ClientSecret for authentication> -b <ObjectName> -r <BatchExecution Id> – Example: ./EICSchedulerService.sh -o 5 -u scheduler -w WS1 -c ofsouser 	To Interrupt the ongoing execution	-3	Batch or Batchgroup Run Id does not exist.
			-4	Invalid arguments passed in request/not enough params in Request body
			-5	Invalid Request Headers
			0	Batch or BatchGroup interrupt is success
			-1	Batch or BatchGroup interrupt is failed

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	-s secret - b batch1 -r batch1_2 022-09-0 7_166253 0601924_ 1		-7	Batch or BatchGroup status is not ongoing or already interrupted.

Note

- The status code **-10** can be a general code if the scheduler service APIs are unreachable.
 - On successful request for execute/rerun, the batchrunid corresponding to external unique id will be stored in the AAICL_SS_EXT_BATCH_RUN_ID_MAPPING table.
 - External Unique ID is required only in case of execute/rerun request.
 - Invalid UserID status for EICSchedulerService.sh Utility
- If the EICSchedulerService.sh utility fails with status as -5, then its due to invalid user. The error can be observed in the mmg-service.logs.

Batch Utilities

A new utility, `batch_export_import_utility.sh`, to export and import batch/batchgroups/schedules is introduced in the following path: `OFS_MMG/bin/batch_export_import_utility.sh`.

The # `FTPSHARE` value in the utility must be manually updated before triggering the utility file.

The # `Export` value can be triggered by using the following command:

```
## ./batch_export_import_utility.sh EXPORT <workspace> <mmguser> <batchcode>
batch/schedule/batchgroup ##
```

For example: `./batch_export_import_utility.sh EXPORT CS mmgadmin BatchRest batch`

The # `Import` value can be triggered by using the following command:

```
##./batch_export_import_utility.sh IMPORT <workspace> <mmguser> <batchcode>
batch/schedule/batchgroup <OVERWRITE_flag> <filename without .txt extension>
```

For example: `./batch_export_import_utility.sh IMPORT DEMO7 mmgadmin BatchRest batch`
N CS_batch_BatchRest

13

More

More feature covers:

- **Export Objects:** Utilize Object Migration feature to efficiently export and save objects.
- **Import Objects:** Import previously exported objects into the system.
- **Model Actions:** Review and approve model deployments in bulk.
- **Audit Trail:** Monitor all system activities and changes.
- **Data Pipelines:** Create and manage data pipelines for efficient data flow and processing.

Object Migration

Object Migration is the process of migrating or moving objects between environments.

You may want to migrate objects for reasons such as managing global deployments on multiple environments or creating multiple environments so that you can separate the development, testing, and production processes.

Prerequisites to Migrating Objects

In order to migrate objects, users must be mapped to the **Object Migration Admin Group**.

Note

Identity Administrator Group users cannot migrate objects if they are not mapped to the **Object Migration Admin Group**.

Migration Object Types

You can migrate (import/export) the following object types by clicking the Object Migration icon on the left menu.

- **Schedule:** Provides the instruction to schedule the execution of defined processes. When a schedule is migrated, the associated batch is also migrated.
- **Batch:** A group of jobs that are scheduled to automatically execute at a preset interval of time without any user intervention. When a batch is migrated, the batch and the associated pipeline information are also migrated.
- **Batch Group:** A set of individual batches are consolidated to form a single Batch Group. When you migrate a batch group all the batches, tasks, and pipeline information associated with that batch group are also migrated.
- **Pipeline:** A pipeline is an embedded data processing engine that filters, transforms, and migrates data on-the-fly. Pipelines consist of a set of data processing elements called widgets connected in series, where the output of one widget is the input to the next element.

- **Threshold:** The threshold limit associated with values of set variables for scenarios in FCCM Cloud Service. These threshold values are set when scenarios are created or installed and can be changed, if required.
- **Job:** Jobs provide a set of instructions to execute Workflow Pipelines, based on the set threshold values.
- **PMF Process:** PMF Processes are defined to sequence the Workflow Pipelines, the applications, and to design the artifacts that participate in the pipelines, to implement the pipelines. When exporting a PMF Process, dependent metadata such as data fields, transition rules associated with the PMF process are taken care of.
- **Roles:** Roles are used to map functions to a defined set of groups to ensure user access system security.
- **Groups:** Groups are used to map Roles. Specific User Groups can perform only a set of functions associated with that group.

Export Objects

To access the Object Export Summary page:

1. Click **Launch Workspace** next to corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click the **Object Migration** icon and select **Export Objects**.

The Object Export Summary page containing the records is displayed with the following details.

Figure 13-1 Object Export Summary page

Name	Status	Last Modified By
EXP5	Success	mmgtest
EXP4	Success	mmgtest
EXP3	Success	mmgtest
EXP2	Success	mmgtest
EXP1	Success	mmgtest

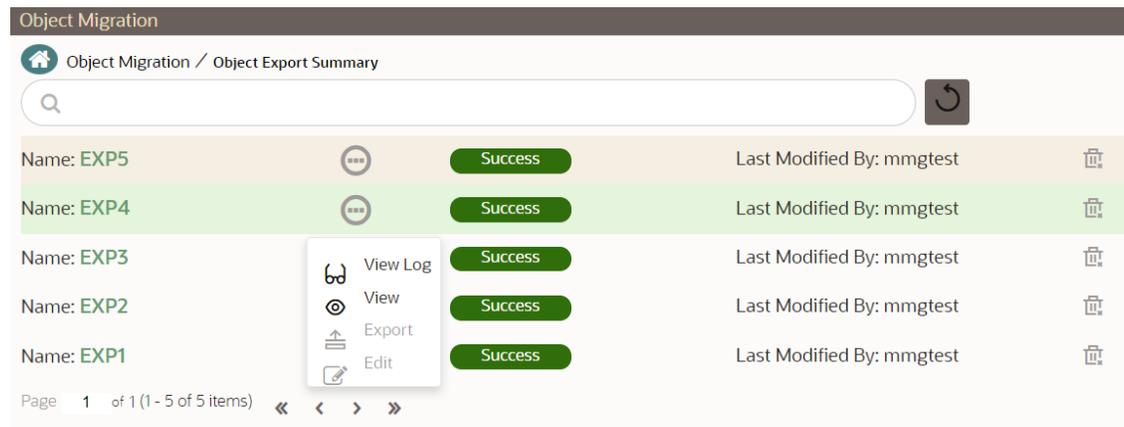
- **Name:** The unique migration name assigned to the collection when the migration definition was created.
- **Object Migration Status:** The migration status of the record corresponding to the specified Definition Name. The three migration status values are as follows:
 - **Success** - Set to Success, when the object export is completed successfully.
 - **Saved** - Set to Saved, when the migration definition is ready for export and needs to import.
 - **Failed** - Set to failed, when the migration definition is not exported successfully.

- **Last Modified By:** The ID of the Last Modified by user who has modified the record.

Search Objects to Export

To search for a specific migration definition, type the first few letters of the user name that you want to search in the **Search** box and click the **Search** icon.

Figure 13-2 Search Objects to Export



Use the controls at the bottom of the page, to set the number of entries displayed per page and to navigate between pages.

Create Migration Export Object Definitions

You can create Migration Export object definitions for the following object types.

- Batch Group
- Threshold
- PMF Process
- Batch
- Pipeline
- Schedule
- Job

To create a definition for export of objects to be migrated, do as follows:

1. Click **Add** in the **Object Export Summary** page to display the **Migration Definition** window.

Figure 13-3 Migration Definition window

Migration Definition

Migration Name Required

File Name

2. In the **Migration Definition** window, enter the details for the following:
 - **Migration Name:** Enter the code of the export of objects to be migrated definition. This is a unique identifier.

- **File Name:** The system auto-creates the file name of the objects that can be used to export the definition in the following format:
 - **For Business Objects:** Migration Name_BO_Time Stamp (MMDDYYYY HHMMSS)
 - **For Identity Objects:** Migration Name_IDM_Time Stamp (MMDDYYYY HHMMYY)
3. Click **Apply** to save the details.
The Object Selection Page is displayed.
 4. In the Object Selection page, select the required Object Type from the Object Types drop-down list. The object types listed for the System Configuration tab are, Schedule, Batch group, Batch, Pipeline, Threshold, Job and PMF process. For more information about the object types, refer [Migration Object Types](#).

You can also enter the first few letters in Search to add a particular object from a selected Object type.

The list of objects is displayed.
 5. Select the objects to be added to the Migrate Definition.
The selected objects are added to the respective object type branch.
 6. Click **Save** to create the Migrate Definition.
A confirmation message is displayed, when the definition is saved successfully.

The new migration definition is listed in the Object Export Summary page and the status is set to Saved.

Note

If the migration definition object is not created successfully and the status is set to Failed. Contact [My Oracle Support](#) for more information.

View Migration Objects to be Exported

You can view and edit the migration objects from the **Object Export Summary** page.

To view the list of migration export objects associated with a migration definition, highlight the migration definition and click **Menu** button.

You can:

Table 13-1 View Migration Export Objects

Field	Description
View Log	View the migration log details of the selected Migration Definition. For more information, refer to View Log of Migration Export Objects .
View	View the Object Details for a specific Migration Definition. Click an object to view more details. For more information, refer to Viewing Export Migration Object Details .

Table 13-1 (Cont.) View Migration Export Objects

Field	Description
Export	<p>Click Export to initiate Object Migration (Export) for a specific Migration Definition.</p> <p>When the migration is completed, the status will change from Saved to Success.</p> <p>For more information, refer to Exporting Migration Objects.</p>
Edit	<p>Click Edit to view and edit the objects linked to a Migration Definition.</p> <p>For more information, refer to Editing Migration Export Definitions.</p>

View Log of Migration Objects Exported

To view the log details of object with migration status Success or Failed, follow these steps. The view log facilitates you to view the log information of the definition for export of objects to be migrated with its status.

1. Highlight the migration definition and click Menu button and select View Log.

The View Log page is displayed.

The status of the export migration is displayed with the following details.

Table 13-2 View Log of Migration Export Objects

Field	Description
Object Migration ID	The migration ID associated with the export object.
Object Type	The object type of the export object.
Object Code	The object code associated with the export object.
Creation Date	The date of creation of the export object.
Created By	The User Id of the User who created the export object.
Status	<p>The migration status of the export object.</p> <ul style="list-style-type: none"> • Success - Indicates that the export migration was completed successfully. • Failed - Indicates that the export migration did not complete

Note

The View Log page for a migration object with status Saved will be empty.

2. Click **OK** to close the window.

View Details of Exported Migration Objects

To view the list of objects added to a specific Migration Definition:

1. Highlight the migration definition and click **Menu** button and select **View**.
The list of migration objects added to the definition is displayed.
2. Double-click an object to view the object attribute details.

Export Migration Objects

To export the list of objects added to a specific Migration Definition:

- Highlight the migration definition and click **Menu** button and select **Export**.
The status is indicated as either Success or Failed.

Note

Use Object Migration (export) to export a set of objects within the same setup or across different setups.

Edit Exported Migration Definitions

You can edit the migration export objects that are not exported and their status is Saved or Failed. If the object is already exported and the status is set to Success, you cannot edit the object details.

To edit a record of the definition of export of objects to be migrated, follow these steps. You can add more objects to export or remove existing objects.

1. Highlight the migration definition and click **Menu** button and select **Edit**.
The Object Selection window is displayed.
2. Update the required details.
3. In the Object Selection window, select the required Object Type from the Object Types drop-down list.

The object types listed for the System Configuration tab are, Batch_Group, PMF_Process, Batch and Schedule. For more information about the object types, refer [Migration Object Types](#). You can also enter the first few letters in Search to add a particular object from a selected Object type.

The list of objects is displayed.

4. Select the objects to be added to the Migrate Definition or to be deleted from the Migration Definition.
The selected objects are added/deleted to the respective object type branch.
5. Click **Save** to edit the Migrate Definition.
A confirmation message is displayed, when the definition is saved successfully.
The edited migration definition is listed in the Object Export Summary page and the status is set to Saved.
6. Click **Save** to update the changes.

Delete Exported Migration Definitions

You can only delete records in the Saved or Failed state; records in Success state cannot be deleted.

To delete a migrate export object definition, highlight the record to be deleted and click the delete button. Click Yes to confirm and proceed with the deletion.

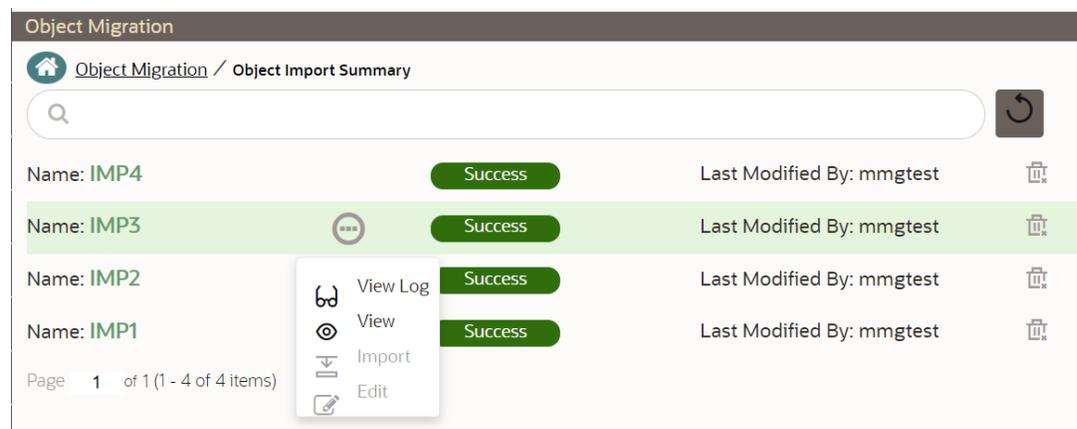
Import Objects

To access the Object, Import Summary page:

1. Click **Launch Workspace** next to corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click the **Object Migration** icon and select **Import Objects**.

The Object Import Summary page containing the records is displayed with the following details.

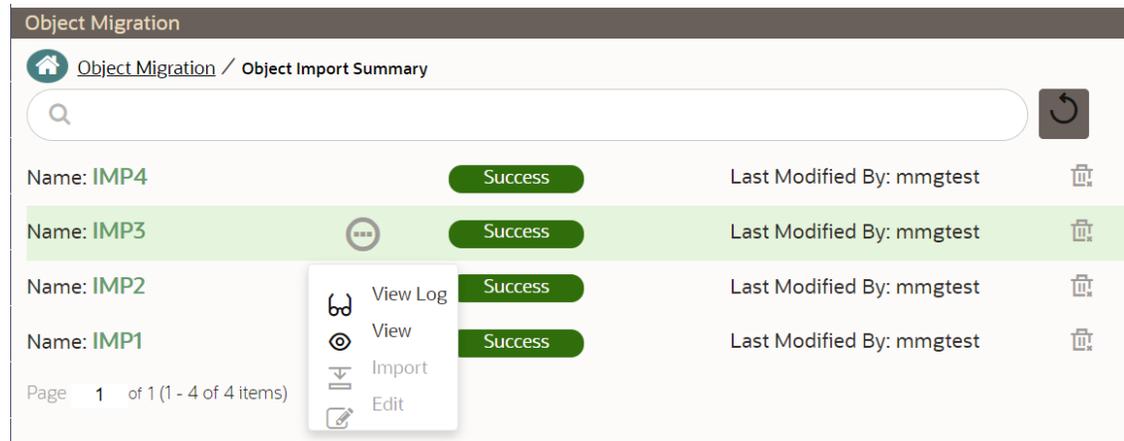
Figure 13-4 Object Import Summary page



- **Name:** The unique migration name assigned to the collection when the migration definition was created.
- **Object Migration Status:** The migration status of the record corresponding to the specified Definition Name. The three migration status values are as follows:
 - **Success** - Set to Success, when the object import is completed successfully.
 - **Saved** - Set to Saved, when the migration definition is ready for import and needs to import.
 - **Failed** - Set to failed, when the migration definition is not imported successfully.
- **Last Modified By:** The ID of the Last Modified by user who has modified the record.

Search Objects to Import

To search for a specific migration definition, type the first few letters of the user name that you want to search in the **Searchbox** and click search icon. The search results display the names that consist of your search string.

Figure 13-5 Object Migration

At the bottom of the page, you have controls to navigate the search entries and also specify the number of entries available on a single page.

Create Import Object Definitions for Migrations

You can create Import Object definitions for migration for the following object types.

- Batch Group
- Threshold
- PMF Process
- Batch
- Pipeline
- Schedule
- Job

To create a definition for import of objects to be migrated, do as follows:

1. Click **Add** in the **Object Import Summary** page to display the **Migration Definition window**.

Figure 13-6 Migration Definition window

The screenshot shows the "Migration Definition" window with the following fields and values:

Field	Value
Id	[Empty]
Dump File	MIG1_exp1_05312022_150459
Import All	Yes
Fail on Error	Yes
Over Write	Yes

At the bottom of the window, there are two buttons: a green "Apply" button with a checkmark icon and a white "Close" button with a circular arrow icon.

2. In the **Migration Definition window**, enter the details for the following:
 - **ID:** Enter a valid name for the new migration import definition. This is a unique identifier.
 - **Dump File:** Select the dump file to be utilized for creating the Migration Import Definition.

3. **Import All:** Select an option to import the nodes that are associated with the selected object type.

You can edit this option if required, in the **Object Selection** page.

Figure 13-7 Object Selection page



- **Yes** - Imports all the nodes that are included in the dump file.
 - **No** - Imports only those nodes that you can select in the Object Selection page.
4. **Fail on Error:** Select an option to proceed with the definition creation in case of an error.
You can edit this option if required, in the **Object Selection** page.
 - **Yes** - Stops the creation process, if error is generated.
 - **No** - Creates the import definition even when error is generated. The node with the error is not included in the object creation.
 5. **Overwrite:** Select an option to overwrite the existing definition file.
You can edit this option if required, in the **Object Selection** page.
 - **Yes** - Replaces the existing Import definition.
 - **No** - Creates a new Import definition.
 6. Click **Apply** to save the details.
The **Object Selection** page is displayed.
 7. In the **Object Selection** page, click **Add Members**, to add objects associated with specific object types.
 8. In the **Object Selection** page, select the required **Object Type** from the **Object Types** drop-down list.

The object types listed are, Schedule, Batch group, Batch, Pipeline, Threshold, Job and PMF process. For more information about the object types, refer [Migration Object Types](#).

You can also enter the first few letters in **Search** to add a particular object from a selected Object type.

The list of objects is displayed.
 9. Select the objects to be added to the Migrate Definition.
You can select the objects only if you select **No** for **Import All**.

The selected objects are added to the respective object type branch.
 10. Click **Save** to create the Migrate Definition.

A confirmation message is displayed, when the definition is saved successfully. The new migration definition is listed in the Object Import Summary page and the status is set to **Saved**.

Note

If the migration definition object is not created successfully and the status is set to **Failed**, contact [My Oracle Support](#) for more information.

View Migration Objects to be Imported

You can view and edit the migration objects from the Object Import Summary Page.

To view the list of imported migration objects associated with a migration definition, highlight the migration definition and click **Menu** button.

The following options are displayed.

Table 13-3 Viewing Migration Import Objects

Field	Description
View Log	View the migration log details of the selected Migration Definition. For more information, refer to View Log of Migration Import Objects .
View	View the Object Details for a specific Migration Definition. Click an object to view more details. For more information, refer to Viewing Import Migration Object Details .
Export	Click Export to initiate Object Migration (Import) for a specific Migration Definition. When the migration is completed, the status will change from Saved to Success. For more information, refer to Importing Migration Objects .
Edit	Click Edit to view and edit the objects linked to a Migration Definition. For more information, refer to Editing Migration Import Definitions .

View Log of Migration Objects Imported

To view the log information of a record of the definition for import of objects to be migrated, follow these steps. The view log facilitates you to view the log information of the definition for import of objects to be migrated with its status.

1. Highlight the migration definition and click **Menu** button and select **View Log**.

The **View Log** page is displayed.

The export migration status with the following details is displayed.

Table 13-4 View Log of Migration Import Objects

Field	Description
Object Migration ID	The migration ID associated with the export object.

Table 13-4 (Cont.) View Log of Migration Import Objects

Field	Description
Object Type	The object type of the export object.
Object Code	The object code associated with the export object.
Creation Date	The date of creation of the export object.
Created By	The User Id of the User who created the export object.
Status	The migration status of the export object. <ul style="list-style-type: none"> • Success - Indicates that the export migration was completed successfully. • Failed - Indicates that the export migration did not complete

Note

The View Log Page for a migration object with status Saved will be empty

2. Click **OK** to close the window.

View Details of Imported Migration Objects

To view the list of objects added to a specific Migration Definition:

1. Highlight the migration definition and click **Menu** button and select **View**.
The list of migration objects added to the definition is displayed.
2. Double-click an object to view the object attribute details.

Import Migration Objects

To export the list of objects added to a specific Migration Definition:

- Highlight the migration definition and click **Menu** button and select **Export**.
After you export, the following types of status are displayed:
 - **Success**- When the object is migrated successfully.
 - **Failed**- When the object migration didn't complete.

Note

Object Migration (export) facilitates you to export a set of objects within the same setup or across different setups.

Note

Currently, if the user give production workspace a name during the import, the utility migrates the dump to the production workspace as well. This should be restricted, as the models in sandbox may or may not be promoted and the invalid status will be shown in the production.

Edit Imported Migration Definitions

You can edit the migration export objects that are not exported and their status is Saved or Failed. If the object is already exported and the status is set to Success, you cannot edit the object details.

To edit a record of the definition of export of objects to be migrated, follow these steps. You can add more objects to export or remove existing objects.

1. Highlight the migration definition and click **Menu** button and select **Edit**.

The **Object Selection** window is displayed.

2. Update the required details.

3. In the **Object Selection** window, select the required **Object Type** from the **Object Types** drop-down list.

The object types listed for the System Configuration tab are, Batch_Group, PMF_Process, Batch and Schedule. For more information about the object types, refer [Migration Object Types](#).

You can also enter the first few letters in **Search** to add a particular object from a selected Object type.

The list of objects is displayed.

4. Select the objects to be added to the Migrate Definition or to be deleted from the Migration Definition.

The selected objects are added/deleted to the respective object type branch.

5. Click **Save** to edit the Migrate Definition.

A confirmation message is displayed, when the definition is saved successfully.

The edited migration definition is listed in the Object Export Summary page and the status is set to Saved.

6. Click **Save** to update the changes.

Delete Imported Migration Object Definitions

You can delete only a record that is set to Saved or Failed status and not a record that is in Success status.

To delete a migrate export object definition:

1. Highlight the record to be deleted and click the **Delete** button.
2. Click **Yes** to confirm and proceed with the deletion.

Prerequisites to Migrate Objects

In order to migrate the objects, the users must be mapped to the **Object Migration Admin Group**.

Note

The Identity Administrator Group Users cannot migrate objects if they are not mapped to the **Object Migration Admin Group**.

Migration Object Types

You can migrate (import/export) the following Object Types by clicking the Object Migration icon on the left menu.

- Schedule
- Batch
- Batch Group
- Pipeline
- Threshold
- Job
- PMF_Process
- Roles
- Groups

Schedule

Schedule provides the instruction to schedule the execution of defined processes. When a schedule is migrated, the associated batch is also migrated.

Batch

A batch is a group of jobs that are scheduled to automatically execute at a preset interval of time, without any user's intervention. When a batch is migrated, the batch and the associated pipeline information are also migrated.

Batch Group

A set of individual batches are consolidated to form a single Batch Group. When we migrate a Batch Group all the batches, tasks and pipeline information associated with that Batch Group are also migrated.

Pipeline

A pipeline is an embedded data processing engine that runs inside the application to filter, transform, and migrate data on-the-fly. Pipelines are a set of data processing elements called widgets connected in series, where the output of one widget is the input to the next element.

Threshold

The threshold limit associated with set variables values for scenarios in FCCM Cloud Service. These threshold values are set when scenarios are created or installed and can be changed, if required.

Job

Jobs provide set of instructions to execute Workflow Pipelines, based on the set threshold values

PMF_Process

PMF_Processes are defined to sequence the Workflow Pipelines the applications, and to design the artifacts that participate in the Pipelines, to implement the Pipelines. Export of PMF_Process will take care of dependent metadata, such as data fields, transition rules associated to the PMF process, that are defined in PMF.

Roles

Roles are used to map functions to a defined set of groups to ensure user access system security.

Groups

Groups are used to map Roles. Specific User Groups can perform only set of functions associated with that group.

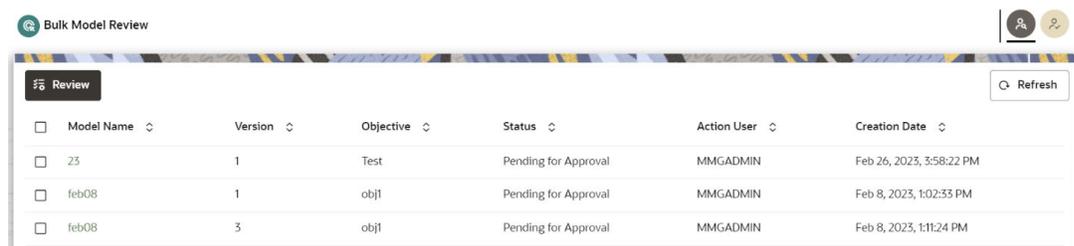
Model Actions

The Model Actions window allows you to view the list of models for which the Workflow action has been taken which requires review or approve across the workspace.

For example, when any Model User sends and Model for review or approval, the user receives the bulk Models in Model Actions window for review or approval. This helps when lot of models are received for review before deployment or some actions need to be performed on many models, such as making multiple models champion locally or globally, and so on.

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. Click Model Actions to view the list models pending an action such as approval.

Figure 13-8 Review Models screen



The screenshot shows the 'Bulk Model Review' interface. At the top, there is a 'Review' button and a 'Refresh' button. Below is a table with the following columns: Model Name, Version, Objective, Status, Action User, and Creation Date. The table contains three rows of data, all with a status of 'Pending for Approval' and action user 'MMGADMIN'.

Model Name	Version	Objective	Status	Action User	Creation Date
25	1	Test	Pending for Approval	MMGADMIN	Feb 26, 2023, 3:58:22 PM
feb08	1	obj1	Pending for Approval	MMGADMIN	Feb 8, 2023, 1:02:33 PM
feb08	3	obj1	Pending for Approval	MMGADMIN	Feb 8, 2023, 1:11:24 PM

If you are a reviewer, Review Models screen is displayed and if you are an approver, Approve Models screen is displayed. This can be toggled in the header of the screen using Reviewer and Approver icon.

3. To review the model, follow these steps:
 - a. Select the Model using the corresponding check box and click **Review**.
 - b. Enter **Comments**, if any, and click **Review**.

Figure 13-9 Review Models window

Review Models ×

Are you sure you want to review the below models for their respective requests?

<input checked="" type="checkbox"/>	Model Name	Version	Status
<input checked="" type="checkbox"/>	23	1	Pending for Approval
<input checked="" type="checkbox"/>	feb08	1	Pending for Approval

Comments

4. To approve or reject the models, follow these steps:
 - a. Select the Model using the corresponding check box and click **Approve** or **Reject**.
 - b. Enter the comments in Comments field of the **Approve Models** window or **Reject Models** window and click **Approve** or **Reject**.

Figure 13-10 Approve Models screen

Approve Models ×

Are you sure you want to approve the below models for their respective requests?

<input checked="" type="checkbox"/>	Model Name	Version	Status
<input checked="" type="checkbox"/>	23	1	Pending for Approval
<input checked="" type="checkbox"/>	feb08	1	Pending for Approval

Comments

Audit Trail

At any time, you can audit the models from the Audit Trail window. The Audit Trail window provides the complete details of model. This shows the information such as, when Model was created, who created the Model, workflow of Model, when this Model became champion or deployed, and so on.

The sequence of actions performed in the model lifecycle is listed in the table view and the timeline view (graphical representation).

To audit models:

1. Click **Launch Workspace** next to the corresponding Workspace to display the **Dashboard** window with application configuration and model creation menu.
2. In the Mega menu, click **More > Audit Trail** to view the time of the various actions performed on the model in the **Audit Trail** window.

Figure 13-11 Audit Trail window

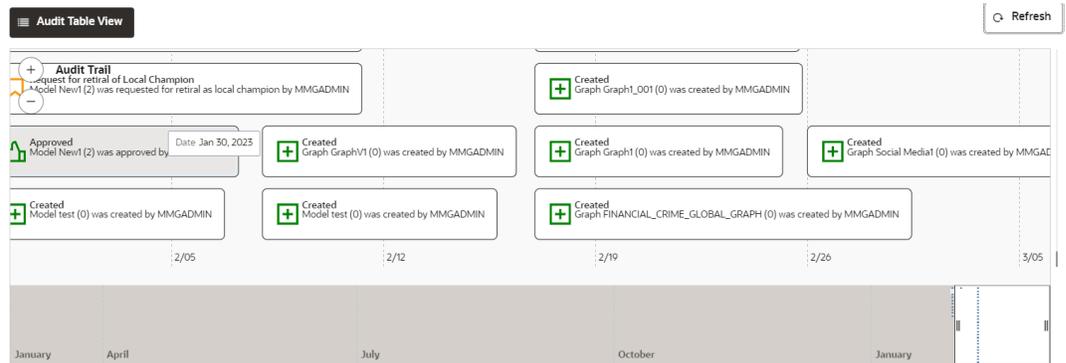
Status	Type	Name	User	Comments	Time
Created	Dataset notebook	XCASCXAS	MMGADMIN	Dataset notebook XCASCXAS (0) was created by MMGADMIN	3 Mar 2024, 13:11:29
Created	Dataset notebook	ZKZ	MMGADMIN	Dataset notebook ZKZ (0) was created by MMGADMIN	3 Mar 2024, 13:02:28
Created	Model	Widget1	MMGSAML	Model Widget1 (1) was created by MMGSAML	29 Feb 2024, 08:56:53
Created	Model	Widget1	MMGSAML	Model Widget1 (0) was created by MMGSAML	29 Feb 2024, 08:52:31
Modified	Model technique	T1	MMGADMIN	Model technique T1 (0) was modified by MMGADMIN	28 Feb 2024, 18:00:52
Modified	Model technique	T1	MMGADMIN	Model technique T1 (0) was modified by MMGADMIN	28 Feb 2024, 17:59:25
Modified	Model technique	T1	MMGADMIN	Model technique T1 (0) was modified by MMGADMIN	28 Feb 2024, 17:57:22
Created	Model technique	T1	MMGADMIN	Model technique T1 (0) was created by MMGADMIN	28 Feb 2024, 17:55:55
Created	Model	df11	MMGADMIN	Model df11 (0) was created by MMGADMIN	28 Feb 2024, 15:31:42
Created	Model	df9	MMGADMIN	Model df9 (0) was created by MMGADMIN	28 Feb 2024, 15:31:27

3. Click for Timeline View.

Click to switch to the regular view. Click to refresh the Audit Trail window.

You can search the models by entering the name in the search box.

Figure 13-12 Audit Trail Window Timeline View with Horizontal Time Axis



A

APIs for Model Pipelines

All the listed APIs/Functions can be utilized by users in Model Pipelines.

Table A-1 Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Set User	Set the user for a session	-	<pre>%python from mmg.constants import * user = "SAUSER" set_user(user)</pre>	user - username to be set	8.1.1
Get User	Get the user that has been set	Set User	<pre>%python from mmg.constants import * get_user()</pre>	-	8.1.1
Attach Workspace	Attach the workspace for a session	-	<pre>%python from mmg.workspac e import attach_workspa ce workspace = "CS" attach_workspa ce(workspace)</pre>	workspace- workspace to be attached	8.1.1
Get Workspace	Get the workspace that has been attached	Attach Workspace	<pre>%python from mmg.workspac e import get_workspace get_workspace()</pre>		8.1.1
List Workspaces	Lists all workspaces available to a user	Set User	<pre>%python from mmg.workspac e import list_workspaces list_workspace s()</pre>		8.1.1

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Check AIF	Check if AIF is set or not	Set User	<pre>%python from mmg.workspac e import check_aif check_aif()</pre>		8.1.1
List All Data Sources	List all data sources available in a workspace for a set user. The default workspace chosen is the one that is attached.	Set User, Attach Workspace	<pre>%python from mmg.workspac e import list_datasources list_datasource s(order=None)</pre>	order - None ensures that all the data sources will be listed for the attached workspace	8.1.1
List All Data Sources of Workspace	List all Data Sources of a specific Workspace that is not the attached workspace for a set user.	Set User	<pre>%python from mmg.workspac e import list_datasources workspace="HE LLO NEW" list_datasource s(workspace,or der=None)</pre>	workspace - Name of workspace for which data sources should be listed order - None ensures that ALL data sources will be listed for that workspace	8.1.1
Fetch First Order Data Source	This method will fetch the data source related to attached workspace with default order 1.	Set User, Attach Workspace	<pre>%python from mmg.workspac e import list_datasources list_datasource s()</pre>	order - 1 by default, if not specified	8.1.1
Fetch First Order Data Source of Workspace	This method will fetch the data source with default order 1 related to specified workspace.	Set User	<pre>%python from mmg.workspac e import list_datasources workspace="CS " list_datasource s(workspace)</pre>	workspace - Name of workspace for which data source should be listed order- 1 by default if not specified	8.1.1

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Fetch Nth Order Data Source	This method will fetch the data source related to attached workspace with given order	Set User, Attach Workspace	%python from mmg.workspace import list_datasources list_datasources(order=2)	order - Order of data source to be fetched	8.1.1
Fetch Nth Order Data Source of Workspace	This method will fetch the data source with given order related to specified workspace.	Set User	%python from mmg.workspace import list_datasources workspace="CS" list_datasources(workspace,order=2)	workspace - Name of workspace for which data source should be listed order = Order of data source to be fetched	8.1.1
Get Connection Object	Returns Oracle/SQLAlchemy Connection Object to Data Source of order 1 in attached workspace.	Set User, Attach Workspace	%python import cx_Oracle from mmg.workspace import get_connection conn=get_connection() if not isinstance(conn, cx_Oracle.connect): print("Not Connection Object") print(conn)		8.1.2.0

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Get Connection Object To Specific Data Source	Returns Oracle/SQLAlchemy Connection Object to specified Data Source	Set User	<pre>%python import cx_Oracle from mmg.workspac e import get_connection datasource="D S001" conn=get_conn ection(datasour ce) if not isinstance(conn, cx_Oracle.conn ect): print("Not Conn Object") print(conn)</pre>	datasource - Name of datasource for which the connection object has to be returned	8.1.2.0
Get Connection Object To Data Source Using Order	Returns Oracle/SQLAlchemy Connection Object to Data Source of specified order in attached workspace.	Set User, Attach Workspace	<pre>%python import cx_Oracle from mmg.workspac e import get_connection datasource=2 conn=get_conn ection (datasource) if not isinstance(conn, cx_Oracle.conn ect): print("Not Conn Object") print(conn)</pre>	datasource - Order number of datasource in attached workspace for which the connection object has to be returned	8.1.2.0

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Test Connection Object	This script can be used to test the connection object to data source	Get Connection Object OR Get Connection Object To Specific Data Source OR Get Connection Object To Data Source Using Order	%python query="SELEC T table_name FROM user_tables" cur = conn.cursor() cur.execute(que ry) print(cur.fetchal l()) if cur: cur.close()		8.1.2.0

Note

The property `RESTRICT_UNMAPPED_DATASTORES_ACCESS` has been introduced in the `NEXTGENEMF_CONFIG` table to control datastore access within notebook sessions. By default, this property is set to `false`, which means access to all datastores is allowed. If a user wants to restrict access so that only the datastores attached to the current workspace are accessible, the value of this property must be set to `true`. When enabled, the `get_conn()` OR `get_connection()` method will only allow connections to those datastores that are mapped to the same workspace as the notebook session. This restriction is effective only when the session-mode in the `application.yml` file is configured as `NOTEBOOK_USER`. The property accepts only two values: `true` or `false`.

Table A-2 PYTHON LINEPARSER APIs (for MMG version 8.1.0)

API Type	Functionality	Notebook Execution Script	Note
Attach Workspace	Attach the workspace for a session	%python mmg.attachWorkspace(workspace) #eg- mmg.attachWorkspace("CS")	workspace - workspace to be attached (String) Sets 'mmg' python variable which contains the information about the attached workspace. Output- Print a boolean in one line on the status of workspace attachment and another line of boolean on status of AIF enabled and attached.

Table A-2 (Cont.) PYTHON LINEPARSER APIs (for MMG version 8.1.0)

API Type	Functionality	Notebook Execution Script	Note
List Workspaces	Lists all workspaces available to a user	%python mmg.listWorkspaces()	Sets 'ofs_wsList' python variable which is the list of all available workspaces. Output- Print the list of workspaces.
Check AIF	Check if AIF is set or not	%python print(aif__isAttached)	Output- Boolean on the status of AIF.
Get Connection Object	Returns cx_Oracle Connection Object	%python mmg.getConnection(schema_name) #eg- mmg.getConnection("OF SAA")	schema_name = name of the schema for which connection is required (String). Sets 'conn' python variable which is the cx_Oracle connection object to be used for firing queries.
Publish	Fetch and sets the Notebook JSON dump	%python mmg.publish()	Sets 'emf_para' python variable which is the JSON dump for the notebook component. Output- Print the set 'emf_para' value.
Register	Register the Notebook	%python mmg.register(register) #eg- mmg.register({"modelName":"model1",\ # "modeldescription":"Model description"}) #eg- mmg.register(emf_para) {After doing mmg.publish()}	register - JSON Stringify object for RegisterNotebookBean Object. Sets 'publishedNotebookId' python variable which is the studio notebook id for the published version. Output- Prints the published notebook id.
Load Flat File	-	%python mmg.loadFlatFile() #eg- mmg.loadFlatFile() Incomplete	
Profile Data	-	%python mmg.profileData(code,title) #eg- mmg.profileData(ABC,XYZ)	

Table A-3 MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
1	Set User	Set the user for a session		<pre>%python from mmg.constants import * user = "SAUSER" set_user(user)</pre>
2	Get User	Get the user that has been set	Set User	<pre>%python from mmg.constants import * get_user()</pre>
3	Attach Workspace	Attach the workspace for a session		<pre>%python from mmg.workspace import attach_workspac e workspace = "CS" attach_workspac e(workspace)</pre>
4	Get Workspace	Get the workspace that has been attached	Attach Workspace	<pre>%python from mmg.workspace import get_workspace get_workspace()</pre>
5	Get Base URL	Get MMG Studio Service Base URL		<pre>%python from mmg.constants import * get_mmg_studio_ service_url()</pre>
6	List Workspaces	Lists all workspaces available to a user	Set User	<pre>%python from mmg.workspace import list_workspaces list_workspace s()</pre>
7	Check AIF	Check if AIF is set or not	Set User	<pre>%python from mmg.workspace import check_aif check_aif()</pre>
8	List All Data Sources	List all data sources available in a workspace for a set user. The default workspace chosen is the one that is attached.	Set User, Attach Workspace	<pre>%python from mmg.workspace import list_datasource s list_datasource s(order=None)</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
9	List All Data Sources of Workspace	List All Data Sources of a specific Workspace that is not the attached workspace for a set user	Set User	<pre>%python from mmg.workspace import list_datasources workspace="HELLO NEW" list_datasources(workspace,order=None)</pre>
10	Fetch First Order Data Source	This method will fetch the data source related to attached workspace with default order 1	Set User, Attach Workspace	<pre>%python from mmg.workspace import list_datasources list_datasources()</pre>
11	Fetch First Order Data Source of Workspace	This method will fetch the data source with default order 1 related to specified workspace	Set User	<pre>%python from mmg.workspace import list_datasources workspace="CS" list_datasources(workspace)</pre>
12	Fetch Nth Order Data Source	This method will fetch the data source related to attached workspace with given order	Set User, Attach Workspace	<pre>%python from mmg.workspace import list_datasources list_datasources(order=2)</pre>
13	Fetch Nth Order Data Source of Workspace	This method will fetch the data source with given order related to specified workspace	Set User	<pre>%python from mmg.workspace import list_datasources workspace="CS" list_datasources(workspace,order=2)</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
14	Get Connection Object	Returns Oracle/SQLAlchemy Connection Object to Data Source of order 1 in attached workspace	Set User, Attach Workspace	<pre>%python import cx_Oracle from mmg.workspace import get_connection conn=get_conne ction() if not isinstance(conn ,cx_Oracle.conn ect): print("Not Conn Object") print(conn)</pre>
15	Get Connection Object To Specific Data Source	Returns Oracle/SQLAlchemy Connection Object to specified Data Source	Set User	<pre>%python import cx_Oracle from mmg.workspace import get_connection datasource="DS" conn=get_conne ction(datasource , conn_type=None) if not isinstance(conn ,cx_Oracle.conn ect): print("Not Conn Object") print(conn)</pre>
16	Get Connection Object To Data Source Using Order	Returns Oracle/SQLAlchemy Connection Object to Data Source of specified order in attached workspace	Set User, Attach Workspace	<pre>%python import cx_Oracle from mmg.workspace import get_connection datasource=2 conn=get_conne ction(datasource , conn_type=None) if not isinstance(conn ,cx_Oracle.conn ect): print("Not Conn Object") print(conn)</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
17	Get Connection Object	Returns Oracle/SQLAlchemy Connection Object to Data Source of order 1 in attached workspace	Set User, Attach Workspace	<pre>%python import oracledb from mmg.workspace import get_conn conn=get_conn() if not isinstance(conn ,oracledb.Connection): print("Not Conn Object") print(conn)</pre>
18	Get Connection Object To Specific Data Source	Returns Oracle/SQLAlchemy Connection Object to specified Data Source	Set User	<pre>%python import oracledb from mmg.workspace import get_conn datasource="DS" conn=get_conn(d atasource, conn_type=None) if not isinstance(conn ,oracledb.Connection): print("Not Conn Object") print(conn)</pre>
19	Get Connection Object To Data Source Using Order	Returns Oracle/SQLAlchemy Connection Object to Data Source of specified order in attached workspace	Set User, Attach Workspace	<pre>%python import oracledb from mmg.workspace import get_conn datasource=2 conn=get_conn(d atasource, conn_type=None) if not isinstance(conn ,oracledb.Connection): print("Not Conn Object") print(conn)</pre>
20	Get Objective Hierarchy	Returns the complete objective hierarchy including pipeline name		<pre>%python print(objective Id)</pre>
21	Fetch FTPSHARE Path	Get complete ftpshare path on the server		<pre>%python from mmg.datasets import fetch_ftpshare fetch_ftpshare()</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
22	Get Pipeline Name	Get the Pipeline Name to which the current notebook is attached	Attach Workspace	<pre>%python from mmg.constants import get_pipeline_name get_pipeline_name()</pre>
23	Get Pipeline ID	Get ID of Pipeline to which the current notebook is attached	Attach Workspace	<pre>%python from mmg.constants import get_pipeline_id get_pipeline_id()</pre>
24	Get Pipeline Version	Get version of Pipeline to which the current notebook is attached	Attach Workspace	<pre>%python from mmg.constants import get_pipeline_version get_pipeline_version()</pre>
25	Get MISDATE	Return MISDATE from pipeline execution if set otherwise takes default value provided	Attach Workspace	<pre>%python misdate=ds.date_picker(name='\$FICMISDATE\$',format='yyyy-MM-dd',default_value='2000-01-01',label='FICMISDATE') print(misdate)</pre>
26	Get BATCHRUNID	Returns BATCHRUNID if set from pipeline, otherwise takes default value		<pre>%python batchrunid=ds.textbox(name='\$BATCHRUNID\$',default_value='Batch_auto',label='BATCHRUNID') print(batchrunid)</pre>
27	Get TASKID	Returns TASKID if set from pipeline, otherwise takes default value		<pre>%python taskid=ds.textbox(name='\$TASKID\$',default_value='task1',label='TASKID') print(taskid)</pre>

Miscellaneous Helper Python Scripts

Table A-4 Miscellaneous Helper Python Scripts

Index	Name	Description	Prerequisite	Script
1	Create Connection Object Using Wallet Alias	This script can be used to test the connection to any Oracle database using wallet alias. Make sure to close connection object after use.		<pre>%python wallet_alias = "WALLET_ALIASE" import oracledb oracledb.init_oracle_client() conn = oracledb.connect(dsn=wallet_alias)</pre>
2	Test Connection Object	This script can be used to test the connection object to data source	Get Connection Object OR Get Connection Object To Specific Data Source OR Get Connection Object To Data Source Using Order	<pre>%python query="SELECT table_name FROM user_tables" cur = conn.cursor() cur.execute(query) print(cur.fetchall()) if cur: cur.close()</pre>
3	Close Connection Object		Connection Object should be created using any of the above methods	<pre>%python query="SELECT table_name FROM user_tables" cur = conn.cursor() cur.execute(query) print(cur.fetchall()) if cur: cur.close()</pre>
4	Close Cursor		Cursor Object should be created using any of the above methods	<pre>%python if cur: cur.close()</pre>
5	Check environment variables	For example: Check wallet alias value of MMG_LIB_WALLET_ALIAS		<pre>%python import os print(os.environ['MMG_LIB_WALLET_ALIAS'])</pre>

Table A-4 (Cont.) Miscellaneous Helper Python Scripts

Index	Name	Description	Prerequisite	Script
6	Save Dataframe to a File	Save dataframe to a file stored on server		<pre> %python from mmg.datasets import fetch_ftpshare, save_dataframe from mmg.workspace import get_workspace from mmg.constants import get_pipeline_ve rsion misdate=ds.date _picker(name='\$ FICMISDATE\$', format='yyyy- MM-dd', default_value=' 2000-01-01', label='FICMISDA TE') batchrunid=ds.t extbox(name='\$B ATCHRUNID\$', default_value=' Batch_auto', label='BATCHRUN ID') taskid=ds.textb ox(name='\$TASKI D\$', default_value=' task1', label='TASKID') filename="df.cs v" filepath=fetch_ ftpshare() + "/mmg/" + get_workspace() + "/pipelines/" + objectiveId + "/" + get_pipeline_ve rsion() + "/" output/" + misdate + "/" + batchrunid + "/" + taskid + "/" + filename res=save_datafr ame(df,filepath </pre>

Table A-4 (Cont.) Miscellaneous Helper Python Scripts

Index	Name	Description	Prerequisite	Script
				<pre>) message=res["messages"] if res["status"]== "ERROR": raise Exception(messa ge) else: print(message)</pre>

Python Linepraser APIs (for MMG version 8.1.0)

Table A-5 Python Linepraser APIs (for MMG version 8.1.0)

Index	API Type	Functionality	Notebook Execution Script	Note
1	Attach Workspace	Attach the workspace for a session	<pre>%python mmg.attachWorks pace(workspace) #eg- mmg.attachWorks pace("CS")</pre>	workspace = workspace to be attached (String) Sets 'mmg' python variable which contains the information about the attached workspace Output- Print a boolean in one line on the status of workspace attachment and another line of boolean on status of AIFenabled and attached
2	List Workspaces	Lists all workspaces available to a user	<pre>%python mmg.listWorkspa ces()</pre>	Sets 'ofs_wsList' python variable which is the list of all available workspaces Output- Print the list of workspaces
3	Check AIF	Check if AIF is set or not	<pre>%python print(aif__isAt tached)</pre>	Output- Boolean on the status of AIF

Table A-5 (Cont.) Python Linepraser APIs (for MMG version 8.1.0)

Index	API Type	Functionality	Notebook Execution Script	Note
4	Get Connection Object	Returns cx_Oracle Connection Object	<pre>%python mmg.getConnection(schema_name) #eg- mmg.getConnection("OFSAA")</pre>	<p>schema_name = Name of the schema for which connection is required (String)</p> <p>Sets 'conn' python variable which is the cx_Oracle connection object to be used for firing queries</p>
5	Publish	Fetch and sets the Notebook JSON dump	<pre>%python mmg.publish()</pre>	<p>Sets 'emf_para' python variable which is the JSON dump for the notebook component Output-Print the set 'emf_para' value</p>
6	Register	Register the Notebook	<pre>%python mmg.register(register) #eg- mmg.register({"modelname": "model1", \ # "modeldescription": "Model description"}) #eg- mmg.register(emf_para) {After doing mmg.publish() }</pre>	<p>register = JSON Stringify object for RegisterNotebookBean Object Sets 'publishedNotebookId' python variable which is the studio notebook id for the published version. Output- Prints the published notebook ID</p>
7	Load Flat File	<pre>%python mmg.loadFlatFile() #eg- mmg.loadFlatFile() Incomplete</pre>		
8	Profile Data		<pre>%python mmg.profileData(code,title) #eg- mmg.profileData(ABC,XYZ)</pre>	<p>code = title =</p>

B

Public APIs for Scheduler Operations

Public APIs for Scheduler Operations

The following APIs are exposed for the Scheduler Operations.

Note

1. All the below APIs are POST requests.
2. All the APIs accept below values in request header:
 - ofs_tenant_id - Optional
 - ofs_service_id - Optional
 - ofs_workspace_id - Respective workspace id where the batch needs to be created
 - ofs_remote_user – MMG login user The ofs_workspace_id and ofs_remote_user are mandatory fields.

Base URL: `http(s)://<MMG_Service_HostName>:<MMG-Service_Port>/<CONETXTNAME>/`

Table B-1 APIs for Scheduler Operations

Functionality	API	Sample Request JSON	Sample Response JSON	Comments
Execute Immediately	rest-api/v1/external/trigger	<pre>{ "batchName": "batch1", "batchType": "rest", "includedTasks": "task1,task2", "excludedTasks": "task1,task2", "heldTasks": "task1,task2" }</pre>	<pre>{ "severity": "info", "summary": "Object triggered successfully with Run Id: batch1_2022-08-17_1660721567845_1", "batchRunId": "batch1_2022-08-17_1660721567845_1", "details": "Object triggered successfully.", "status": "success", "statusCode": "0" }</pre>	<ul style="list-style-type: none">• The includedTasks/excludedTasks/heldTasks are options keys.• The batchType will be "group" for batch group execution.

Table B-1 (Cont.) APIs for Scheduler Operations

Functionality	API	Sample Request JSON	Sample Response JSON	Comments
Execution Status	rest-api/v1/external/status	<pre>{ "batchRunId": "batch1_2022-08-17_1660721567845_1", "tasks": ["task1", "task2"] }</pre>	<pre>{ "severity": "info", "batchRunId": "batch1_2022-08-17_1660721567845_1", "taskStatusList": [{ "taskCode": "task1", "taskStatus": "SUCCESSFUL", "statusCode": "0" }, { "taskCode": "task2", "taskStatus": "EXCLUDED", "statusCode": "4" }], "batchStatusCode": "0", "batchList": [], "batchStatus": "SUCCESSFUL", "status": "success", "statusCode": "0" }</pre>	<ul style="list-style-type: none"> "tasks" is options key. <p>If not passed, then response will contain status all the tasks inside a batch.</p>
Rerun	rest-api/v1/external/rerun	<pre>{ "batchName": "batchgroup1", "batchRunId": "batchgroup1_2022-08-17_1660720814942_1" }</pre>	<pre>{ "severity": "info", "summary": "Object triggered successfully for rerun with Run Id: batchgroup1_2022-08-17_1660730049819_1", "batchRunId": "batchgroup1_2022-08-17_1660730049819_1", "details": "Object triggered successfully.", "status": "success", "statusCode": "0" }</pre>	

Table B-1 (Cont.) APIs for Scheduler Operations

Functionality	API	Sample Request JSON	Sample Response JSON	Comments
Restart	rest-api/v1/external/restart	<pre>{ "batchName": "MMG_R1", "batchRunId": "MMG_R1_2022-07-15_1657867378160_1" }</pre>	<pre>{ "severity": "info", "summary": "Object triggered successfully for restart with Run Id: MMG_R1_2022-07-15_1657867378160_1", "batchRunId": "MMG_R1_2022-07-15_1657867378160_1", "details": "Object triggered successfully.", "status": "success", "statusCode": "0" }</pre>	
Interrupt	rest-api/v1/external/interrupt	<pre>{ "batchName": "B2001", "batchRunId": "B2001_2022-05-22_1653222717896_1" }</pre>	<pre>{ "summary": "Execution interrupted successfully for Run Id: B2001_2022-05-30_1653233511394_1", "severity": "info", "batchRunId": "B2001_2022-05-30_1653233511394_1", "details": "Execution interrupted successfully.", "statusCode": "0", "status": "success" }</pre>	

Table B-2 Failed Response Body

Failed Sample Response JSON	Description
Batch Status Failed	<pre>{ "severity": "info", "batchRunId": "BT-BI-EXCHG_RATES_EOD_2022-05-27_1653662703599_1", "batchStatusCode": "-1", "batchList": [], "batchStatus": "FAILED", "statusCode": "0", "status": "success" }</pre>

Table B-2 (Cont.) Failed Response Body

Failed Sample Response JSON	Description
Object Not found	<pre>{ "severity": "error", "summary": "Object does not exist.", "details": "Object does not exist.", "error": { "errorCode": "OBJECT_NOT_EXIST", "errorMsg": "Object does not exist." }, "statusCode": "-3", "status": "failed" }</pre>

Table B-3 Status Codes in the Response Body

Status Code	Description
0	Success
1	Not Started
2	Ongoing
4	Excluded
5	Held
-1	Failure
-2	Interrupted
-3	Object does not exist
-4	Invalid arguments passed in request/not enough parameters in the Request body
-5	Invalid request headers/request headers missing
-6	No executable job is present
-7	Job is already interrupted