

Oracle Financial Services Model Management and Governance User Guide



Release 26.1.0
G52879-05
April 0

ORACLE®

Copyright © 2014, 2026, 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	3
Graphs	4
Scheduler Service	4
Audit Trail	4
Data Studio Options	4
Object Migration	5
Model Libraries	12
Model Techniques	12
Model Catalog	12
Data Management	12
Managing User Groups	12
User Groups	12
Role Mapping	15
Function-Role Matrix for Conda CRUD Operations	17
Sign in	18

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
MMG Schema Sync Utility	31
Use the MMG Schema Sync Utility	31

7 File Manager

8 Administration

Data Stores	2
Adding a Data Store	4
Viewing the Data Store Details	9
Conda Environments	9
Register a Conda Environment	12

Create a Conda Environment	16
Edit a Conda Environment	16
View a Conda Environment	17
Deregister a Conda Environment	18
Inspect a Conda Environment	19
Remove Environment	19
Clone Environment	20
Export Environment	20
Edit a Conda Environment Description	21
Add or Remove Environment Tags	21
Search All Packages	21
View Package Properties	22
Update Package	22
Remove Package	22
Clone Environment	23
Export Environment	24
Modify Conda Settings	24
View Conda Properties	25
Using the .condarc Conda Configuration File	25
API Registry	29
Configure an API	30
View an API Registry	33
Edit an API Registry	33
Delete an API Registry	34
Kafka Topics	35
Create a Kafka Cluster	36
Delete a Kafka Cluster	37
Register a Kafka Topic	37
View a Kafka Topic	39
Edit a Kafka Topic	40
Delete a Kafka Topic	40
OFSA Environment	41
Data Studio Options	42
Configure Interpreters	42
Manage Tasks	43
Manage Tasks	43
Manage Credentials	44
Configure Templates	44
Audit Trail	44
Identity Management	45

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 Catalog	43
View a Model Catalog	46
Delete a Model Catalog	62
Pipelines	62
Prerequisites	62
Access Model Pipelines	62
Manage Models	69
Create Objective (Folders)	70
Create Draft Models	71
Create Seeded Models	77
Pipeline Designer	78
Pipeline Canvas	79
Parameter Set	116
Import/Export Parameter Sets using Utility	117
Scheduler Configuration UI	118
Publish Model Draft	119

Scope Detail	120
View Model Details	120
Model Governance	121
Request Model Acceptance	123
Review Models and Move to Approve or Reject	124
Approve Models and Promote to Production	124
Deploy Models in Production and Make it a Global Champion	125
Import a Model Data into a New Model	125
Git Functionalities	127
Execute Models using Scheduler Service	129
Define a Task	129
When Component is Model	131
When Component is Populate Workspace	132
SS \$RUNSK support for processing Delta Data by Data Pipeline and Match Rules	133
Data Pipeline Execution	133
Export/Import/Retire Models via Utility	134
View Models	136
Edit Models	136
Delete Objectives and Draft Models	136
Programmatic Dynamic Forms and Fixed Dynamic Forms	137
Session Management in Model Pipeline	137
Creating a Session	139
Clearing a Session	140
Reinitializing a Session	140
Switching Between Notebook and Pipeline	141
Graphs	142
Accessing the Graph Summary Page	144
Adding a Graph Pipeline	145
Create a Node	146
Create an Edge	151
Zoom In and Zoom Out	157
Edit a Graph Pipeline	157
Delete a Graph Pipeline	158
Edit a Node	158
Delete a Node	159
Schedule Details	160
Audit History	165
Execution History	165
Analyses	166

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	27
Batch to User Configuration	27
General Configuration	28
External Interface Component Scheduler Service	30
Batch Utilities	42
MMG Scheduler Shell Utility (mmg_scheduler.sh)	42
Scheduler Configuration / Admin User Interface	46

13 Model and AI Risk Management (MARM)

Getting Started with Model and AI Risk Management	1
Model Registration in MARM	1
Access and Security	2
MARM Dashboard	3
Model Inventory Management	4
Validation Workflow	5

14 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
About Data Transformation Platform	18
Getting Started with Data Transformation Pipelines	19
Downloading DTP Pipelines	19
Importing DTP Pipelines	20
Copying DTP Pipelines	20
Add DTP Pipelines (MMG)	21
Editing DTP Pipelines	22
Deleting DTP Pipelines	23
Widgets in Data Transformation Pipelines	24
Managing Data Pipelines	49
Widgets in Data Pipelines	49
About Scenario Pipelines	64

Pre-Configured DTP Scenario Pipelines	64
Managing Threshold Sets (MMG)	64
Accessing Threshold Sets	65
Creating a Threshold Set	65
Copying a Threshold Set	66
Editing Threshold Sets	66
Deleting Threshold Sets	67
About Threshold Simulator	67
Using Jobs (MMG)	71
Creating Jobs	71
Editing Jobs	72
Deleting Jobs	72
Viewing Execution Monitor	73
User Group and Roles Mapping	74
Managing Batches	79
Pre-Configured Batches	80
Purge Batch Details	81
Administrative Tasks	83
Viewing Audit History	83
Creating Focus Types (MMG)	85
Creating Event Types (MMG)	86

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.

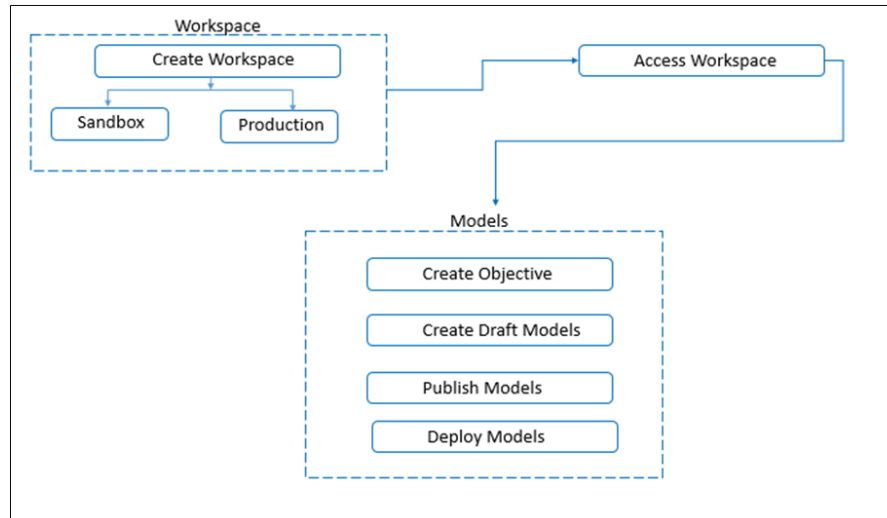
About Oracle Financial Services Model Management and Governance

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

When you save a model pipeline in the **Pipeline Builder**, you can enter comments in the **Add** comments text box that appears in the **Save Pipeline** dialog box. Saved comments are available for review in the **Audit** tab (Audit panel) of the **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" } }

Table 2-1 (Cont.) 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
2	Invoke Export	Headers: ofs_workspace_id ofs_remote_user locale (default set as en-US)	POST	/v1/public/migrate/export/{code}		{ "response": { "message": "Object Migration Triggered Successfully" } }
3	Save Import Definition	Headers: ofs_workspace_id ofs_remote_user locale (default set as en-US)	POST	/v1/public/migrate/import/outline	{ "migrationCode": "import1_exp41", "fileName": "TEST_TESTEXP41_06082024_680336", "code": "import1_exp41" }	{ "response": { "message": "Migration outline definition saved successfully." } }
4	Invoke Import	Headers: ofs_workspace_id ofs_remote_user locale (default set as en-US)	POST	/v1/public/migrate/import/{code}		{ "response": { "message": "Object Migration Triggered Successfully" } }

All APIs should be invoked through Gateway port, if enabled, and UI port, if not enabled. Auth should be the bearer token, which is same as OFS AAI.

Use Bearer Token Authentication

We can execute these APIs: `https://[<mmg_ui_gateway_url/ui_url>/<contextname>/rest-api/v1/token|https://ofss-mum-1035.snbomprshared1.gbucdsint02bom.oraclevcn.com:6773/mmg8127/rest-api/v1/token]`

Basic Auth: ofsauser/secret (configured as properties in the app.property file)

Response:

```
{ "token_type": "Bearer", "expires_in": 3600, "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJqdGkiOiJhZTZNKTUyMC0zYTQwLTRhMjctYTYgMTA0OjYzFmY2FjN2YzMWQwLjEiLCJpc3MiOiJPRlNBQSIsInN1YiI6Im9mc2F1c2VyIiwic3ViamVjdFR5cGU0IjoiJ2JibGlbnRJRCI9ImhhdCI6MTc0MTA4MjQ4MCwiZXhwIjojNzQxMDg2MDgwfwQ.XqbtbZYqKnY1RQv4EW371iZHxrpNWBUrNMx8ef161AhwKozc3qfYgL3fXYfwrQhT9VdSufogu1-stdTYWIs203RtWBniuYyjrK5ghXCyWkzyxpmlALx6v8H3bl1g248GrNPZHTyt1tgKfhYbDM-nCl5f_ztZh8I9IAPa6XImRHfisD5MRv0dVvNn3vsdJd7onLgEQwnCEBGqT6HQ20fuAECEulv5V4qt0lc
```

```
Aib8BWokJM-NCVPiov5uaDPY0klyMljdvA6z5Cmr6-
g0MWIjyX16GLtBSDgBAvsEopLznwnmiQLSEPwAnp8Y3KJZwQQVlS2zY033XDv9B8eufXke0A"}

# save export definition eg:-[https://<mmg_ui_gateway_url/ui_url>/<contextname>|
https://host/]/v1/public/migrate/export/outline input json:

{ "migrationCode":"testexp42", "exportObjects":[ \
{"objectCode":"TEST_population", "objectType":"BATCH" } ]} 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 definition saved successfully." } } output failed -> \
{ "migrationCode": "testexp42", "errorMessage": "objectCode/Type missing",
"statusCode": "INVALID_INPUT", "status": "FAILED" }
```

Invoke Export

For example:[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

For example:[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

For example:[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" } }

```
}
```

Object Migration REST APIs (MMG)

Object Migration enables moving MMG objects between environments (for example, DEV to UAT to PROD) to support controlled deployments and environment separation.

Base URL and Connectivity

All Object Migration APIs must be invoked by using the MMG base URL:

If the gateway is enabled, then use the gateway host and port.

If the gateway is not enabled, then use the MMG UI host and port.

Base URL format: https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/

For example:https://<host>:<port>/mmg8133/

Authentication (Bearer Token same as OFS_AAI)

All migration APIs require Bearer token authentication, using the same token mechanism as OFS_AAI.

Get Bearer Token

Endpoint: POST /rest-api/v1/token

Authentication: Use the basic auth (client/user credentials configured in your deployment)

Example (curl):

```
curl -k -u "<CLIENT_ID>:<CLIENT_SECRET>" -X POST  
"https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/rest-api/v1/  
token"
```

Sample Response:

```
{  
  "token_type": "Bearer",  
  "expires_in": 3600,  
  "token": "<JWT_TOKEN_VALUE>"  
}
```

Use the Token

For all Object Migration APIs, include:

Authorization: Bearer <JWT_TOKEN_VALUE>

Required Headers (All Object Migration APIs)

Include the following headers on every request:

ofs_workspace_id: <WORKSPACE_ID>

ofs_remote_user: <REMOTE_USER>

locale: en-US (default)

Also include standard headers as applicable:

Content-Type: application/json

Authorization: Bearer <JWT_TOKEN_VALUE>

Object Migration Endpoints

Save Export Definition

Creates an export outline definition (objects to export).

Method/Endpoint: POST /v1/public/migrate/export/outline

Request Body (example):

```
{  
  "migrationCode": "testexp42",  
}
```



```
"exportObjects": [  
  {  
    "objectCode": "TEST_population",  
    "objectType": "BATCH"  
  }  
]  
}
```

Success Response (example):

```
{  
  "response": {  
    "message": "Migration outline definition saved successfully."  
  }  
}
```

Failure Response (example):

```
{  
  "migrationCode": "testexp42",  
  "errorMessage": "objectCode/Type missing",  
  "statusCode": "INVALID_INPUT",  
  "status": "FAILED"  
}
```

Example (curl):

```
curl -k -X POST  
"https://<MMG_SERVICE_HOST>:  
<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/v1/public/migrate/export/outline"  
-H "Authorization: Bearer <JWT_TOKEN_VALUE>"  
-H "Content-Type: application/json"  
-H "ofs_workspace_id: <WORKSPACE_ID>"  
-H "ofs_remote_user: <REMOTE_USER>"  
-H "locale: en-US"  
-d '  
{ "migrationCode": "testexp42", "exportObjects": [ { "objectCode":  
"TEST_population", "objectType": "BATCH" }  
]  
'
```

Invoke Export

Triggers export execution for a previously saved export definition.

Method/Endpoint:

POST /v1/public/migrate/export/

Where:

= migrationCode used in Save Export Definition

Success Response (example):

```
{
  "response": {
    "message": "Object Migration Triggered Successfully"
  }
}
```

Example (curl):

```
curl -k -X POST
"https://<MMG_SERVICE_HOST>:
<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/v1/public/migrate/export/testexp42"
-H "Authorization: Bearer <JWT_TOKEN_VALUE>"
-H "ofs_workspace_id: <WORKSPACE_ID>"
-H "ofs_remote_user: <REMOTE_USER>"
-H "locale: en-US"
```

Save Import Definition

Creates an import outline definition (dump file to import and import code).

Method/Endpoint:POST /v1/public/migrate/import/outline

Request Body (example):

```
{
  "migrationCode": "imp1_exp41",
  "fileName": "TEST_TESTEXP41_06082024_680336",
  "code": "imp1_exp41"
}
```

Success Response (example):

```
{
  "response": {
    "message": "Migration outline definition saved successfully."
  }
}
```

```
}  
}
```

Example (curl):

```
curl -k -X POST  
"https://<MMG_SERVICE_HOST>:  
<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/v1/public/migrate/import/outline"  
-H "Authorization: Bearer <JWT_TOKEN_VALUE>"  
-H "Content-Type: application/json"  
-H "ofs_workspace_id: <WORKSPACE_ID>"  
-H "ofs_remote_user: <REMOTE_USER>"  
-H "locale: en-US"  
-d '{  
  "migrationCode": "imp1_exp41",  
  "fileName": "TEST_TESTEXP41_06082024_680336",  
  "code": "imp1_exp41"  
}
```

Invoke Import

Triggers import execution for a previously saved import definition.

Method/Endpoint:POST /v1/public/migrate/import/

Where:

= migrationCode used in Save Import Definition

Success Response (example):

```
{  
  "response": {  
    "message": "Object Migration Triggered Successfully"  
  }  
}
```

Example (curl):

```
curl -k -X POST  
"https://<MMG_SERVICE_HOST>:  
<MMG_GATEWAY_OR_UI_PORT>/<WEB_CONTEXT>/v1/public/migrate/import/imp1_exp41"  
-H "Authorization: Bearer <JWT_TOKEN_VALUE>"  
-H "ofs_workspace_id: <WORKSPACE_ID>"
```

```
-H "ofs_remote_user: <REMOTE_USER>"  
-H "locale: en-US"
```

Note

Use the gateway port when the gateway is enabled; else use the UI port. Authorization must be the Bearer token (OFS_AAI token mechanism). Ensure that ofs_workspace_id and ofs_remote_user are correctly populated per your deployment conventions.

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
IDNTYADMIN	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 viewing 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.
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.

Table 2-2 (Cont.) User Groups

User Group Name	User Group Description
MMGDATAMODELADMIN	The MMG Data Model Admin Group This group contains individual roles that allows the user to perform data model tasks such as view, edit, etc. Note This group has replaced the DMADMIN group.
MMGDATAPOPLNADMIN	The MMG Data Population Admin Group This group contains individual roles that allows the user to perform data population tasks such as view, add, edit, delete, execute etc. Note This group has replaced the DMADMIN group.
MMGHLTHCHKADMIN	The MMG health check user group. Users must be mapped to this group for the Health icon to be enabled on the homepage.
SCHEDULERADMIN	This group contains all the roles and functions related to the Scheduler feature. The SCHEDULERADMINACCESS role contains has all the functions that are related to LINKSCHDLR. The new function LINKSCHDLRCNF, enables you to open the scheduler configuration.

If you create a custom user group (other than the seeded OFS MMG groups) and want users in that group to access the OFS MMG Studio operations, then you must add the group in the following configuration files:

mmgstudio/conf/application.yml

mmgstudio/conf/ofsa-permissions.init.yml

These updates ensure that the custom group is recognized by OFS MMG Studio and is mapped to the required Studio permissions.

Note

Enter group names in UPPERCASE, because AAI/OFSA groups are stored and validated in uppercase.

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
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

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

Group Name	Role Name
MDLAPPR	Model Deployment
MDLAPPR	Workspace Read
MDLAPPR	Model Read
MDLAPPR	Model Access
MDLAPPR	Workspace Access
MDLAPPR	DSAPPROVER
MDLAPPR	MDLRPT (Model Report View)
MDLBATCHUSR	DSBATCH
MDLREV	Workspace Read
MDLREV	Model Review
MDLREV	Model Access
MDLREV	Workspace Access
MDLREV	DSUSER
MDLREV	Model Read
MDLREV	MDLRPT (Model Report View)
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
MDLUSR	MDLRPT (Model Report View)
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
GRAPHUSER	Graph Read Role
GRAPHUSER	Graph Execute Role
GRAPHADMINISTRATOR	Graph Administrator Role
MMGDATAMODELADMIN	MMG_DATAMODELREAD
MMGDATAMODELADMIN	MMG_DATAMODELWRITE
MMGDATAMODELADMIN	MMG_DATAMODELACCESS
MMGDATAMODELADMIN	MMG_DATAMODELADV

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

Group Name	Role Name
MMGDATAPOLNADMIN	MMG_DATAPOLNREAD
MMGDATAPOLNADMIN	MMG_DATAPOLNWRITE
MMGDATAPOLNADMIN	MMG_DATAPOLNEXECUTE
MMGDATAPOLNADMIN	MMG_DATAPOLNACCESS
MMGDATAPOLNADMIN	MMG_DATAPOLNADV
MMGHLTHCHKADMIN	MMGHLTHCHKREAD
OBJMIGADMIN	OMEXADVND
OBJMIGADMIN	OMEXPHTM
OBJMIGADMIN	OMEXREAD
OBJMIGADMIN	OMEXWRITE
OBJMIGADMIN	OMIMADVND
OBJMIGADMIN	OMIMPHTM
OBJMIGADMIN	OMIMREAD
OBJMIGADMIN	OMIMWRITE
SCHEDULERADMIN	BATCH_ADV
SCHEDULERADMIN	SCHEDULERADMINACCESS
SCHEDULERADMIN	BATCH_ADMIN_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	- MDLUSR - MDLREV - MDLAPPR	Summary view
CONDAENVVIEW	CONDAENVREAD	- MDLUSR - MDLREV - MDLAPPR	Read
CONDAENVEXP	CONDAENVREAD	- MDLUSR - MDLREV - MDLAPPR	Export yml file
CONDAENVEXP	CONDAENVWRITE	- MDLREV - MDLAPPR	Export yml file
CONDAENVDEL	CONDAENVWRITE	- MDLREV - MDLAPPR	Delete a registered Conda environment
CONDAENVEDIT	CONDAENVWRITE	- MDLREV - MDLAPPR	Edit a Conda environment
CONDAENVADD	CONDAENVWRITE	- 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:

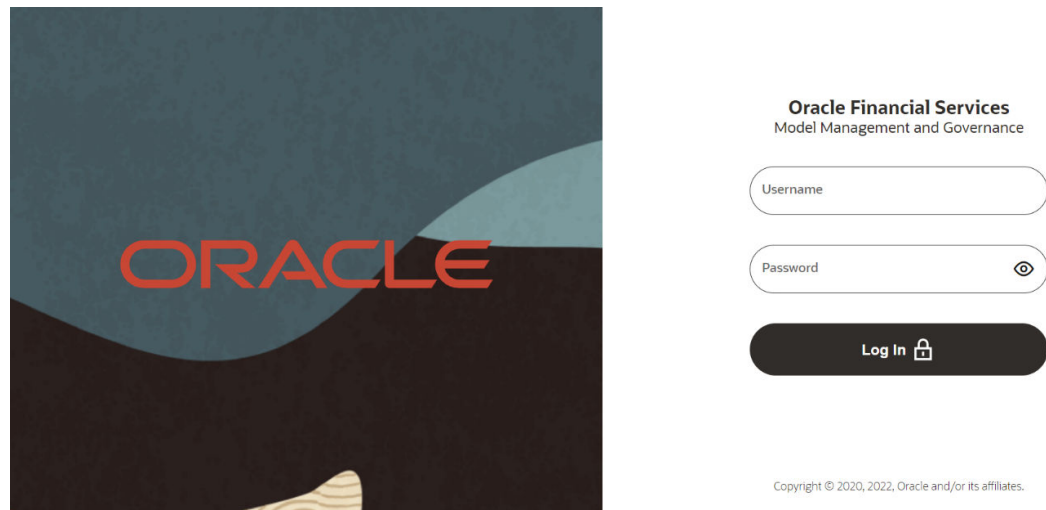
1. Enter the application URL in your browser.

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.

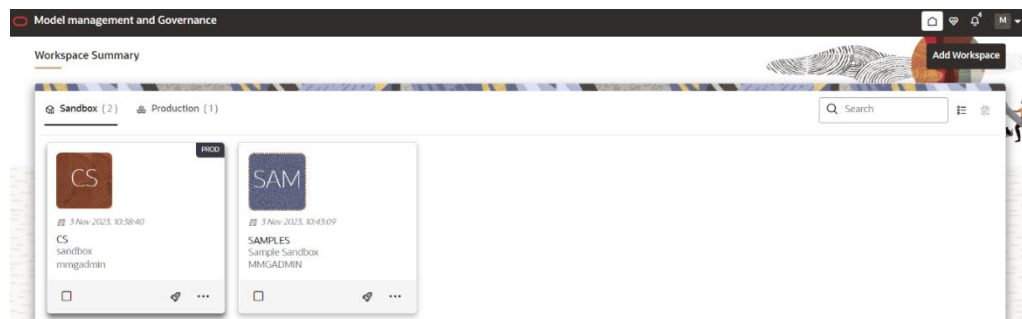
Figure 2-2 OFS Model Management and Governance Application Login Page



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 Infodom 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 nextgenmf_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.

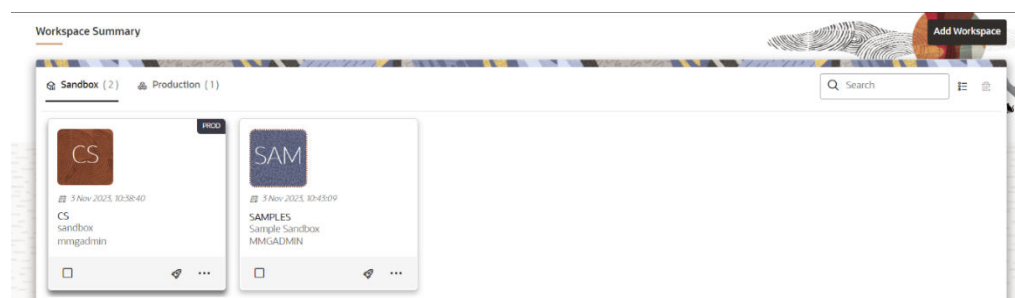
For more details, refer to the Component: STUDIO section of [AAI User Guide](#).

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

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

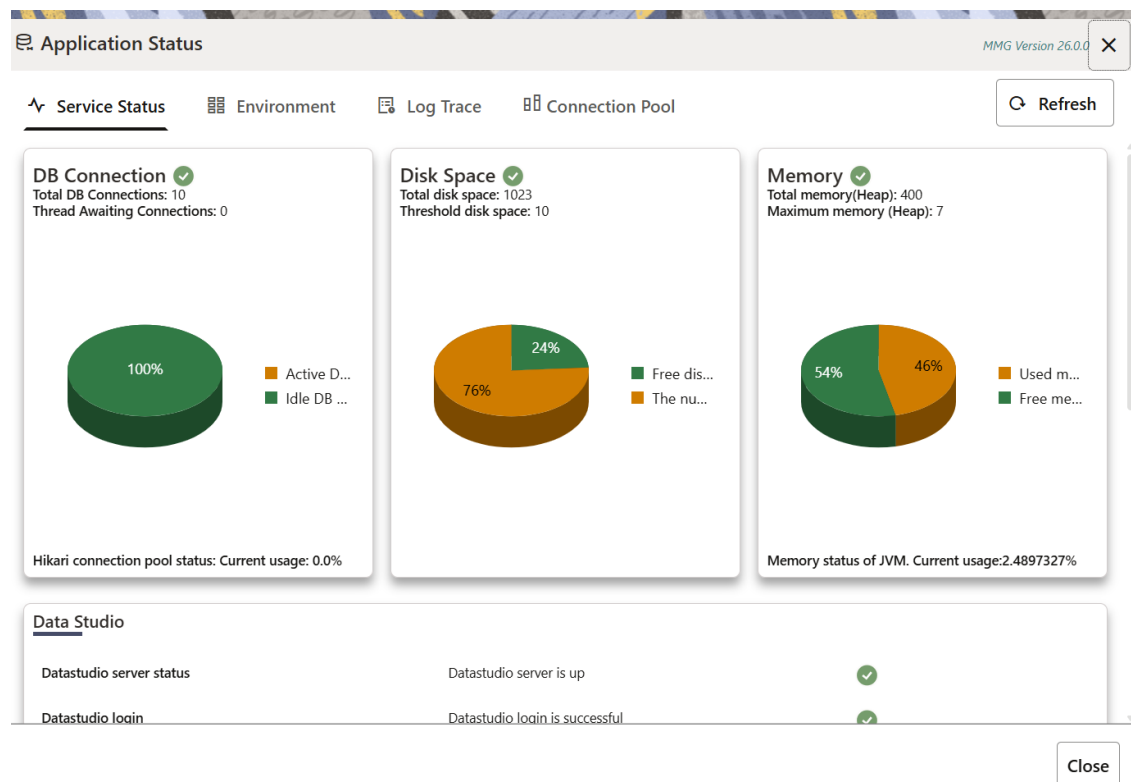
Click **Home** to navigate to **Workspace Summary** page from any other window in the application. You can add and manage Workspaces from this page.

Health

This feature is only accessible if the users are mapped to the health check user group (**MMGHLTHCHKADMIN**). This user group contains the **MMGHLTHCHKREAD** role and **MMGHLTHCHKVIEW** function. If users are not mapped to this group, then the Health icon will not be visible on the homepage.

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

Figure 4-2 Application Status Window



- **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

Notification Settings

DATA POPULATION Update Data Population

Default Setting

Priority: High Category: Alert

User Group List

User Group: Default mapped user groups x

E-mail List

User Name: E-mail: Add

No items to display.

The notification alerts details are updated.

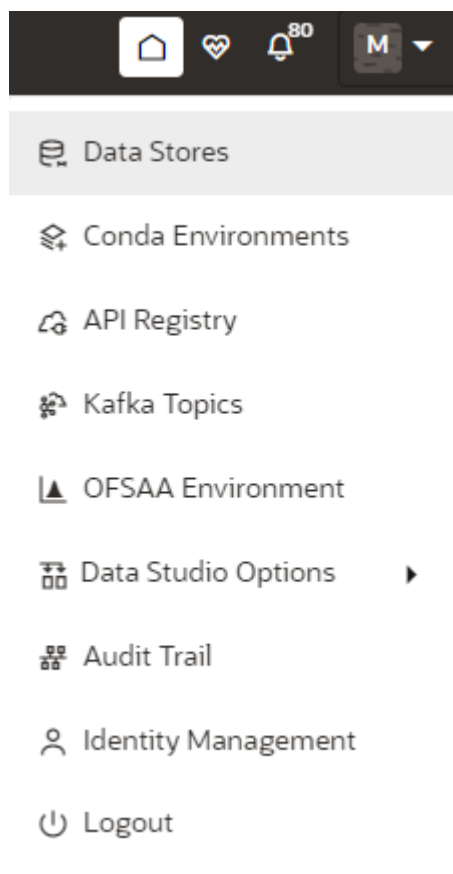
Note

An email notification is sent to the default email id added in AACL_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.
- **OFSA Environment:** Click to navigate to OFSAA Environment page. You can register and 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
 - * 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 nextgenmf_configtable with the values in the MMG installation.

Example:

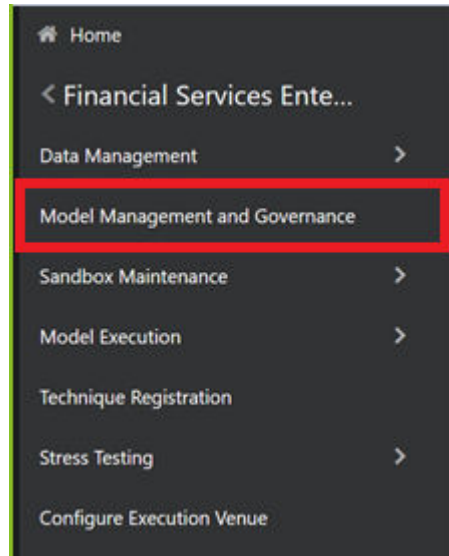
- MMG_UI_URL:http(s)://<FQDN/IP>:<mmg_ui_port>/<mmg_context_name>/home
- MMG_SERVICE_URL :<FQDN/IP>:<mmg_be_port>/<mmg_context_name>

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 preseeded 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

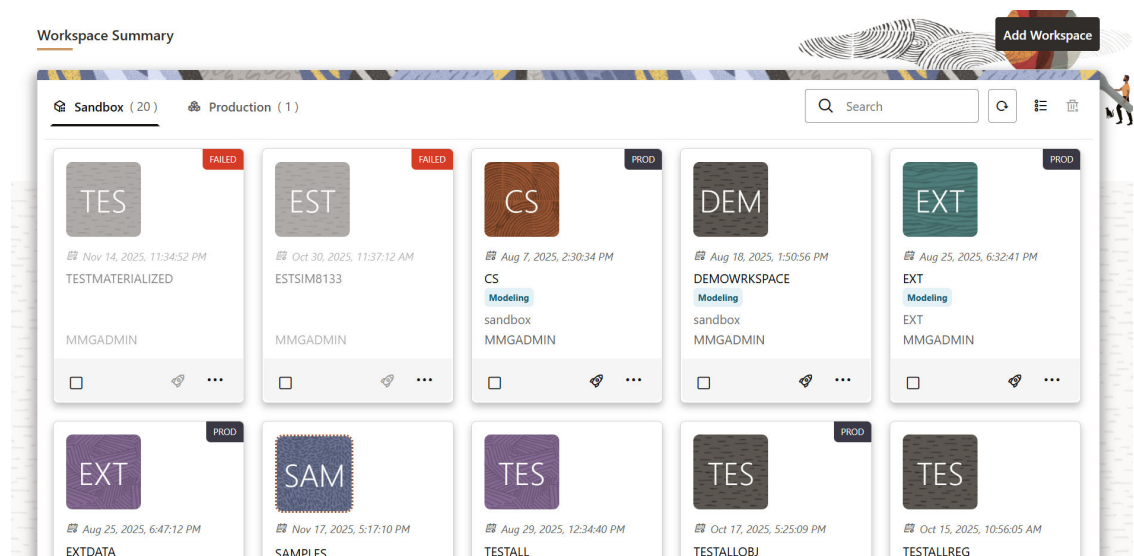
Workspaces

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.

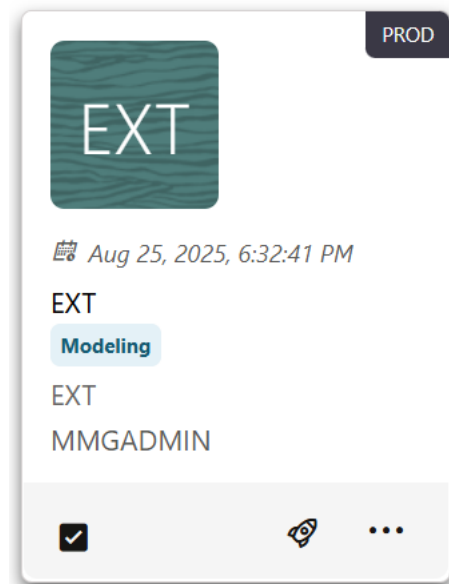
Workspaces can be in the production environment (deployed), or they can exist in a separate instance on their own (on a local system for testing) with a copy of the data that comes from the desired data source (production or external data source).

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.

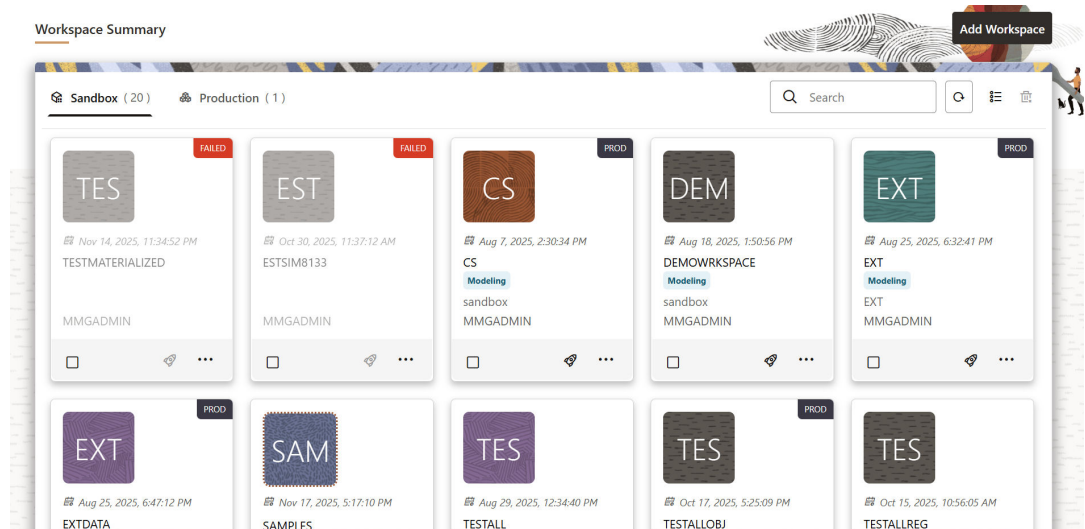
Note

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

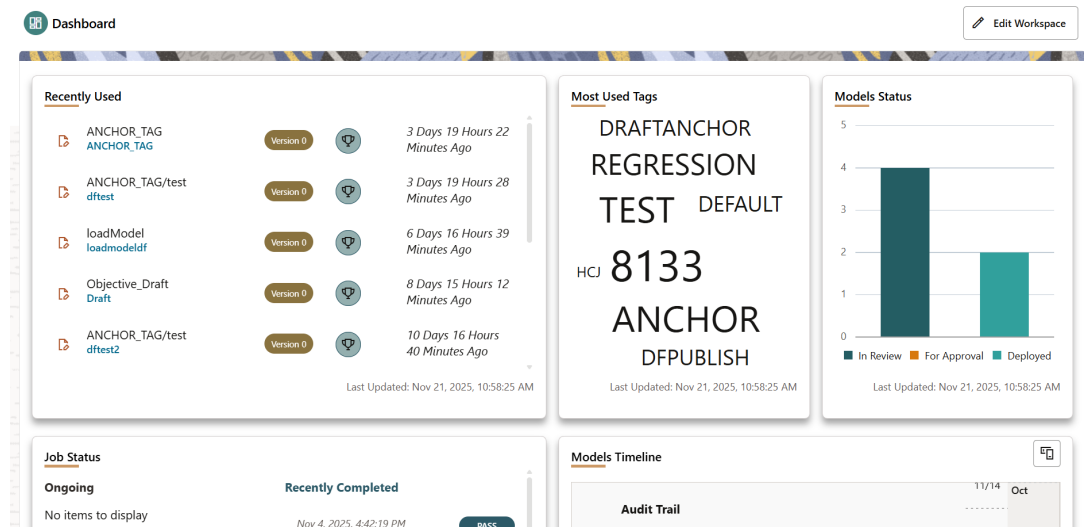
- File Read
- File Delete
- File Write

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

↑ Create Workspace

1 Basic Details

2 Workspace Schema

3 Data Sourcing

4 Metadata Sourcing

5 Validate

6 Summary

Use Template

Workspace Code Required

Purpose Required

User-group Required

User-group Required

Type ☒ Modeling ☐ Simulation Subtype ☒ Sandbox Workspace ☐ Production Workspace

Production Workspace ☐

Import Archive File
Drag & Drop file here

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:

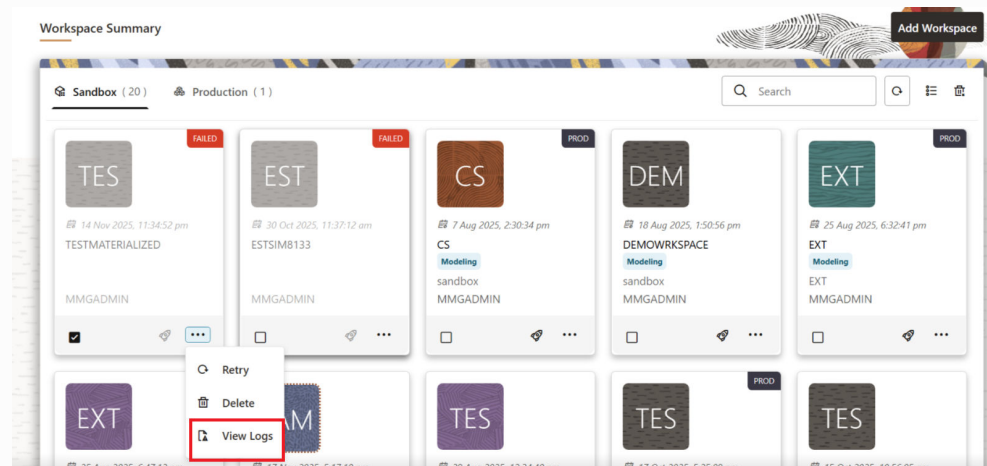
- a. [Basic Details](#)
- b. [Workspace Schema](#)
- c. [Data Sourcing](#)
- d. [Metadata Sourcing](#)
- e. [Validate](#)
- f. [Summary](#)

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

Create Workspace

1 Basic Details

2 Workspace Schema

3 Data Sourcing

4 Metadata Sourcing

5 Validate

6 Summary

Workspace Code Use Template

Purpose

User-group ⓘ

User-group

Type ☒ Modeling ☐ Simulation Subtype ☒ Sandbox Workspace ☐ Production Workspace

Production Workspace ⓘ ☐

Import Archive File

Drag & Drop file here

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.
i Note Use up to 20 characters consisting of alphanumeric and space characters only. Do not use "ALL" as a workspace code.	
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 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.

① 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

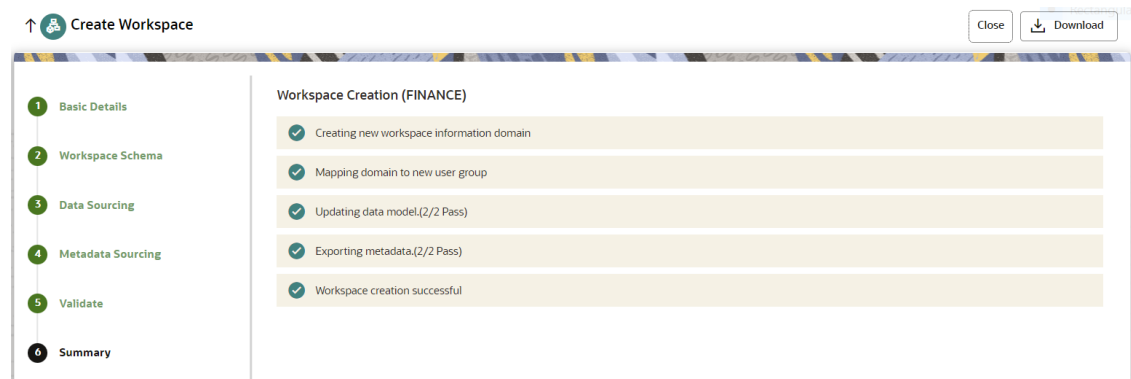
To validate the workspace and deploy it, perform the following steps:

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.

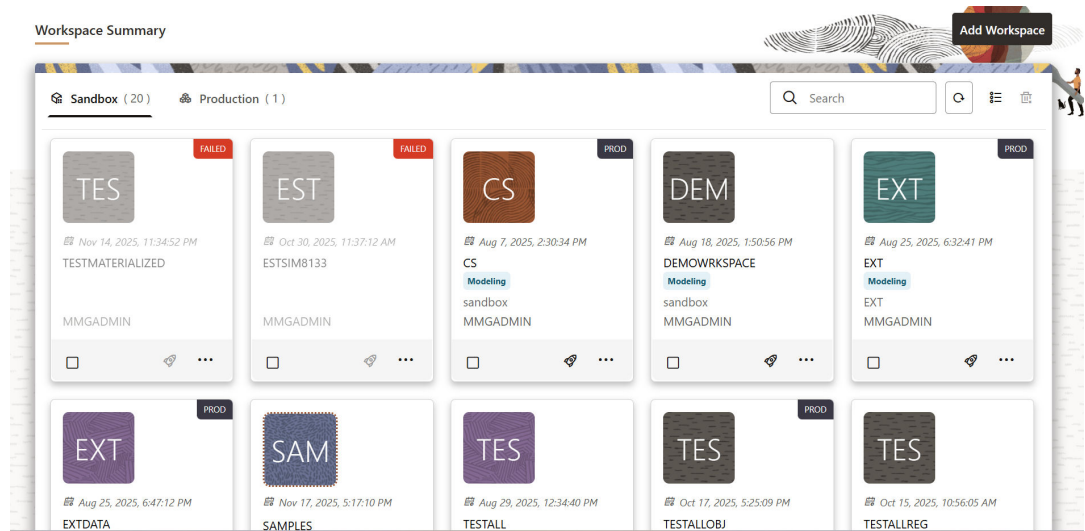
Populate a Workspace

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

Batch Commit Size

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. • 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: • 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	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. Note: You can also pass runtime values in the workspace population for the user-defined parameters in the Data Filter – Global/Table field. Example: Table Filter /Global Filter <pre>[{"id":1,"filter":"CUSTOMER_ID =\$CUSTID and INCOME =\$income and CUSTOMER_NAME =\$customerName","tables":["CUSTOMERS"]}]]</pre> Additional Parameters: <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> 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.
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	The following two options are available: • 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	☒
Parameter	\$BATCHRUNID\$	Value	BATCHRUNID	☒
Parameter	Global Filter	Value		☐
Parameter	Table Filters	Value		☐
Parameter	Additional Parameters	Value		☐

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

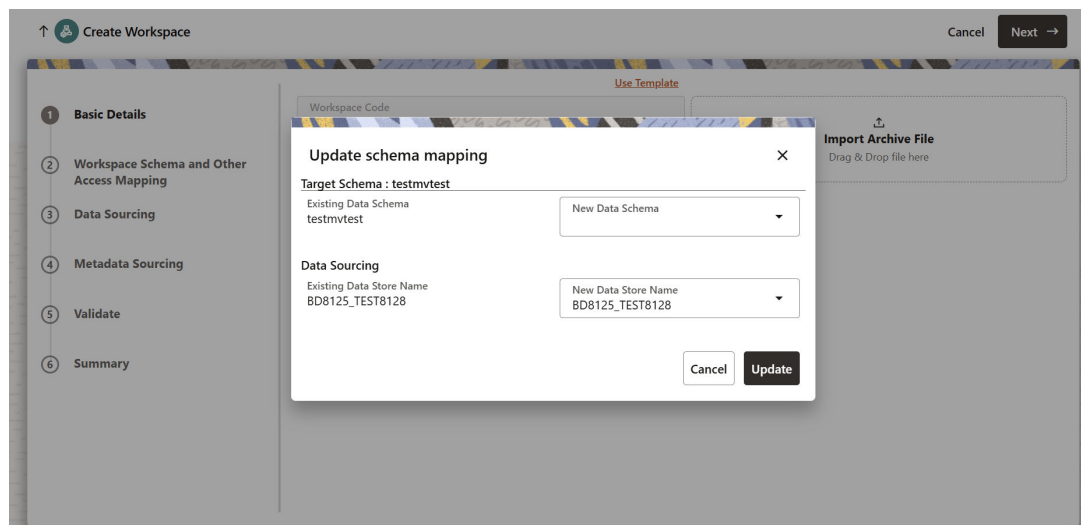
Perform the following steps to retry a workspace that has failed.

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

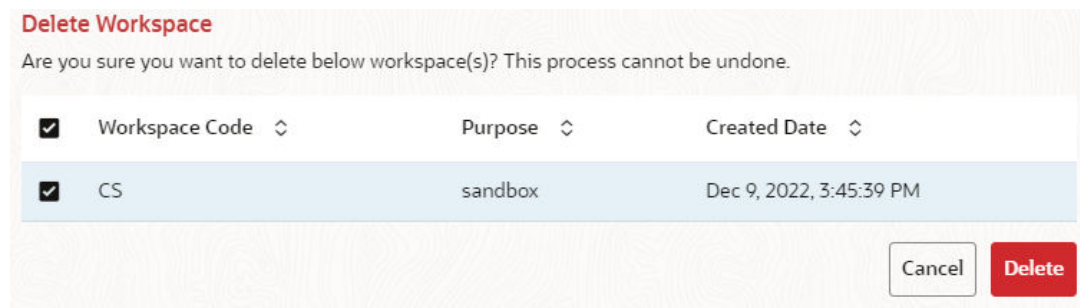
When you delete a Workspace, all the underlying objects such as Dataset, Scheduler service metadata and so on from the associated tables are deleted.

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:

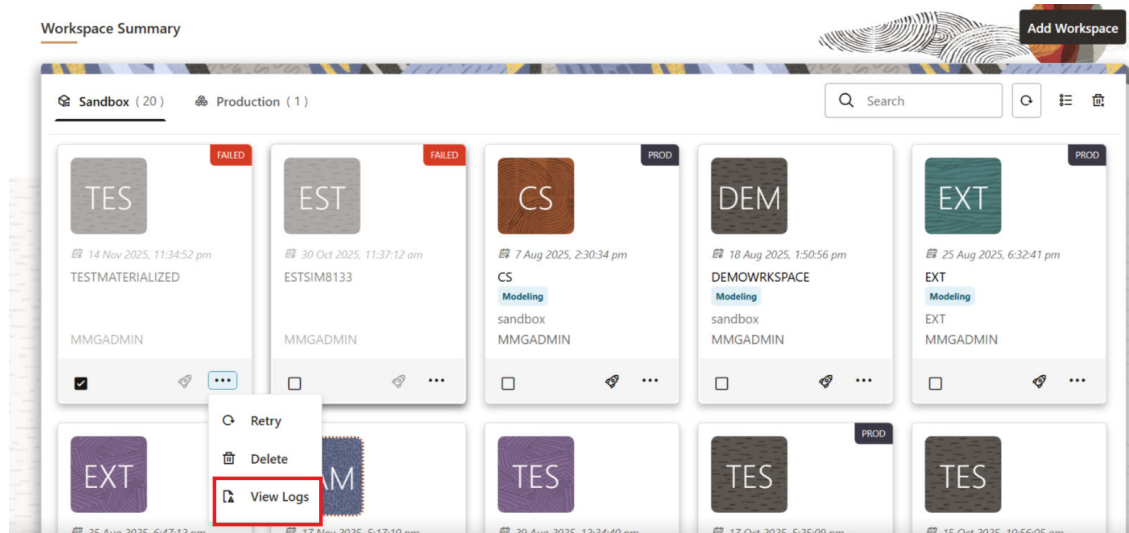
1. Navigate to **Data Source Summary** window.
2. Click **Add Datasource** .

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

Details Screen

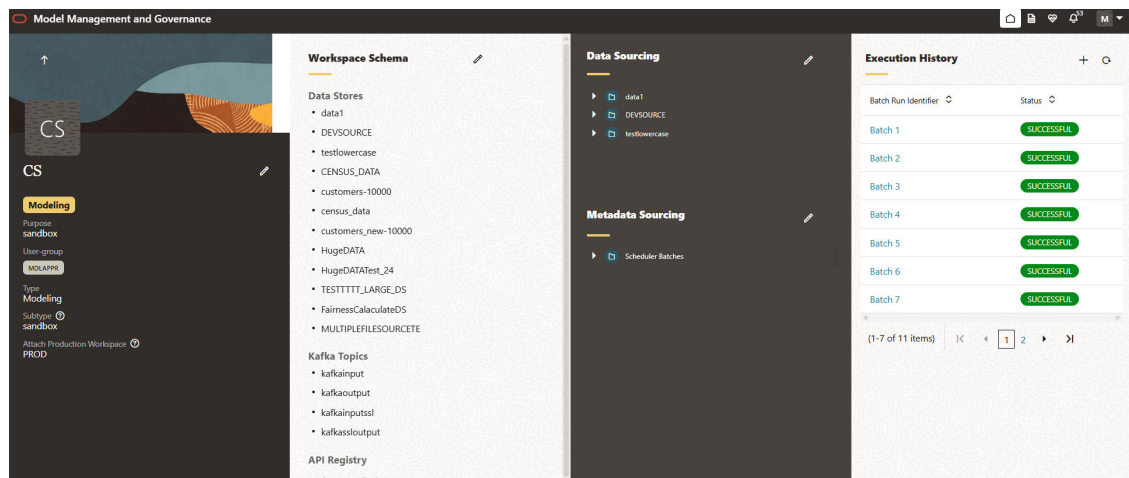
The **Details** screen enables you to edit and view a workspace.

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.

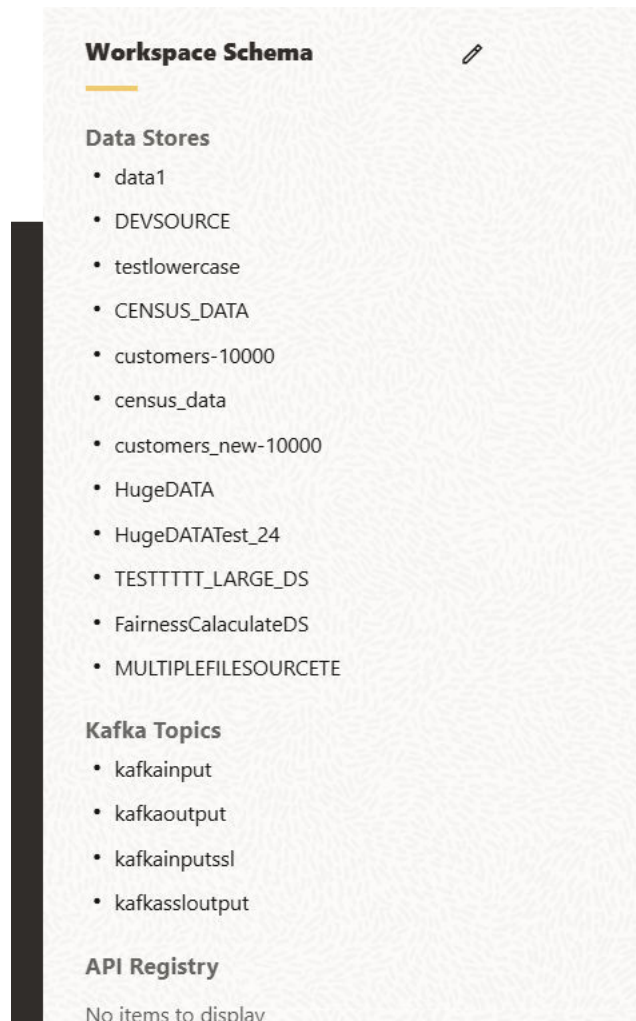
The first column is the basic details screen. It contains the workspace name, purpose, user-group, type, subtype, and the attach production workspace details. You can edit the details in this section if you have edit rights.

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

The screenshot displays the 'Workspace Schema' interface. At the top, there are 'Close' and 'Save' buttons. The main content area is divided into four sections:

- Data Schema:** This section contains a list of data sources, each with a close icon (X):
 - data1 X
 - DEVSOURCE X
 - testlowercase X
 - CENSUS_DATA X
 - customers-10000 X
 - census_data X
 - customers_new-10000 X
 - HugeDATA X
 - HugeDATATest_24 X
 - TESTTTTT_LARGE_DS X
 - FairnessCalaculateDS X
 - MULTIPLEFILESOURCETE X
- Kafka Topics:** This section contains a list of Kafka topics, each with a close icon (X):
 - kafkainput X
 - kafkaoutput X
 - kafkainputssl X
 - kafkassloutput X
- API Registry:** This section is currently empty.
- Conda Environments:** This section is currently empty.

A plus sign (+) icon is located to the right of the Data Schema section, indicating a 'Create Data Store' action.

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

Figure 6-19 Add a Datastore

Add Data Store [X]

Data Store Name Required

Description Required

File Availability
JDBC

Database Type
Oracle

Wallet Alias

Table Owner Required

Additional Properties ?

JDBC Connection String

User Name Password [Eye Icon]

Test Connection Cancel Create

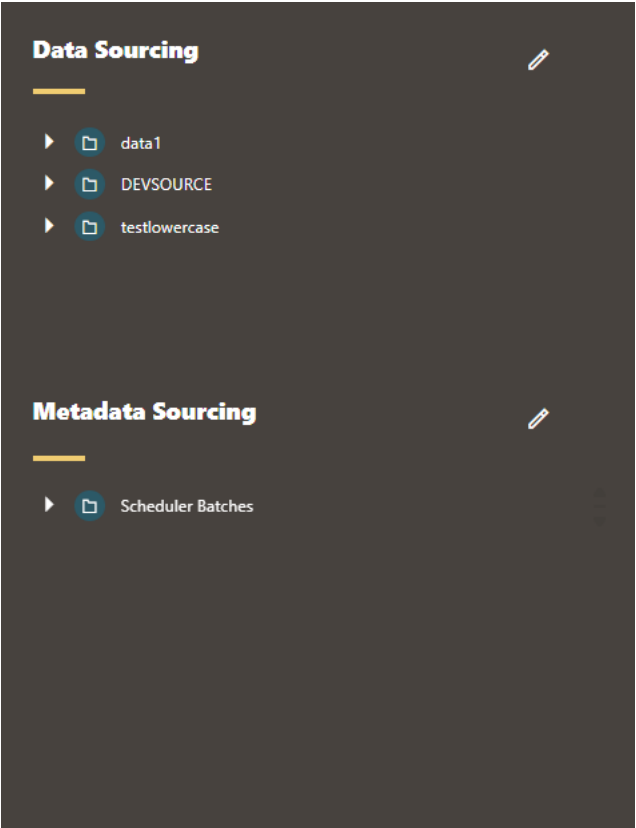
Data Sourcing

Data Sourcing is divided into two sections:

- Data Sourcing
- Metadata Sourcing

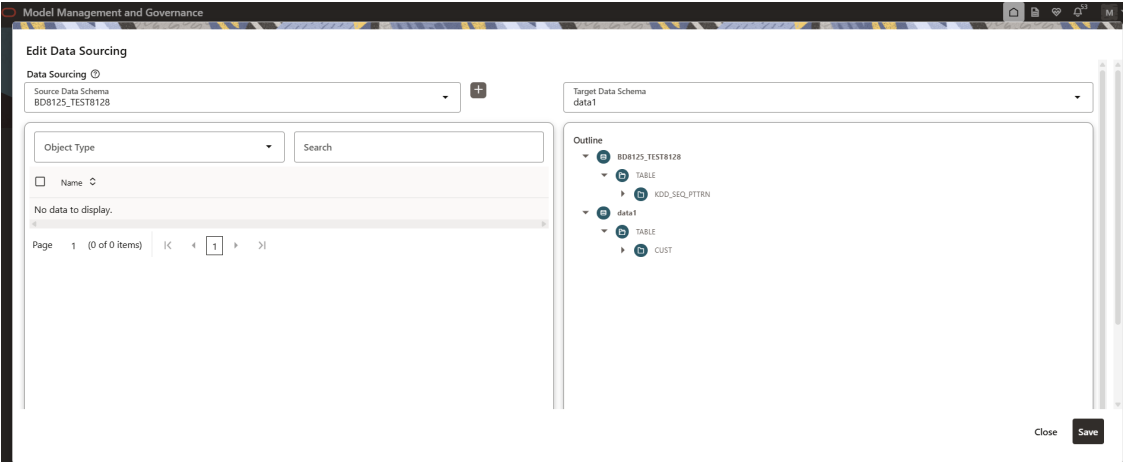
Click the **Add** icon adjacent to **Data Sourcing** to edit it.

Figure 6-20 Data Sourcing



The Edit Data Sourcing pop-up window appears.

Figure 6-21 Edit Data Sourcing

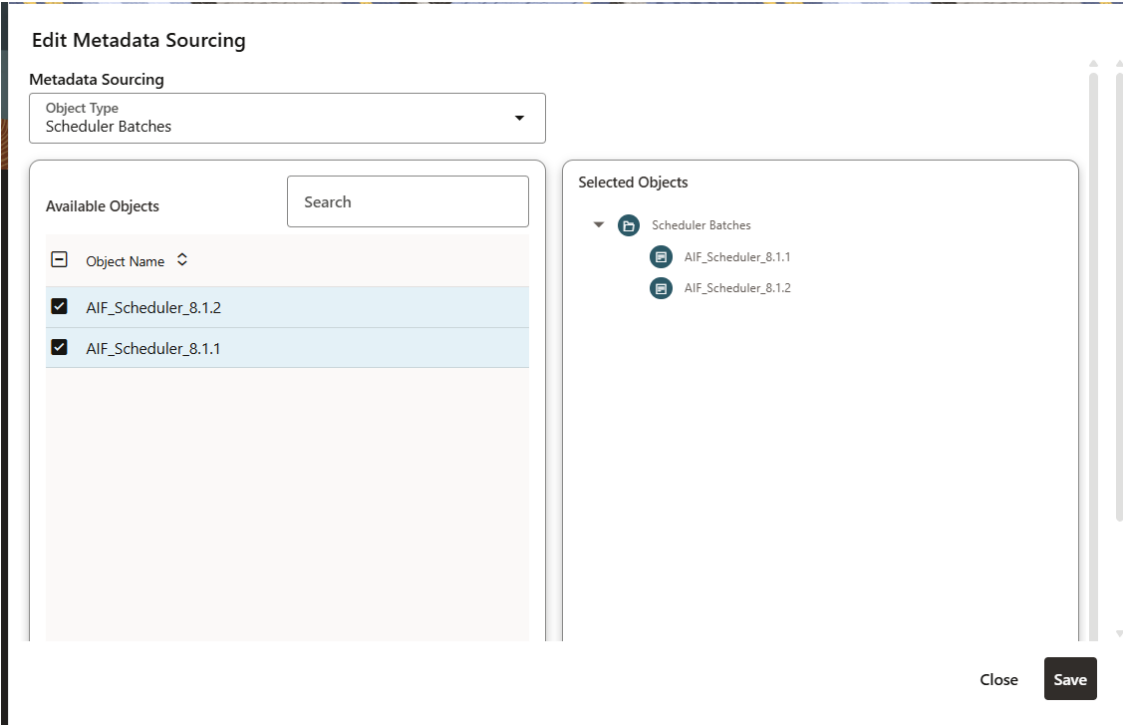


Modify the desired details and then click **Save**.

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

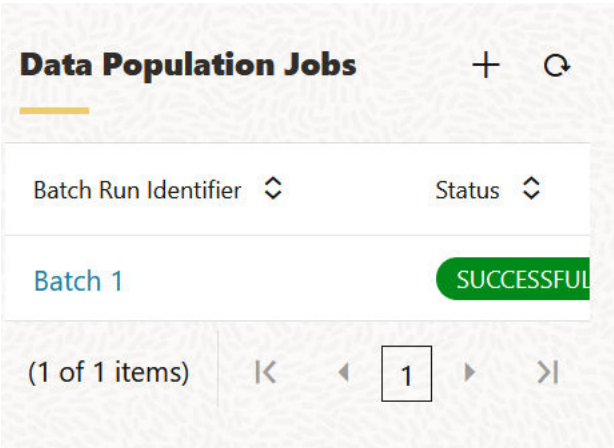


In this window, you can add new objects or edit the existing ones.
Modify the desired details and then click **Save**.

Data Population Jobs

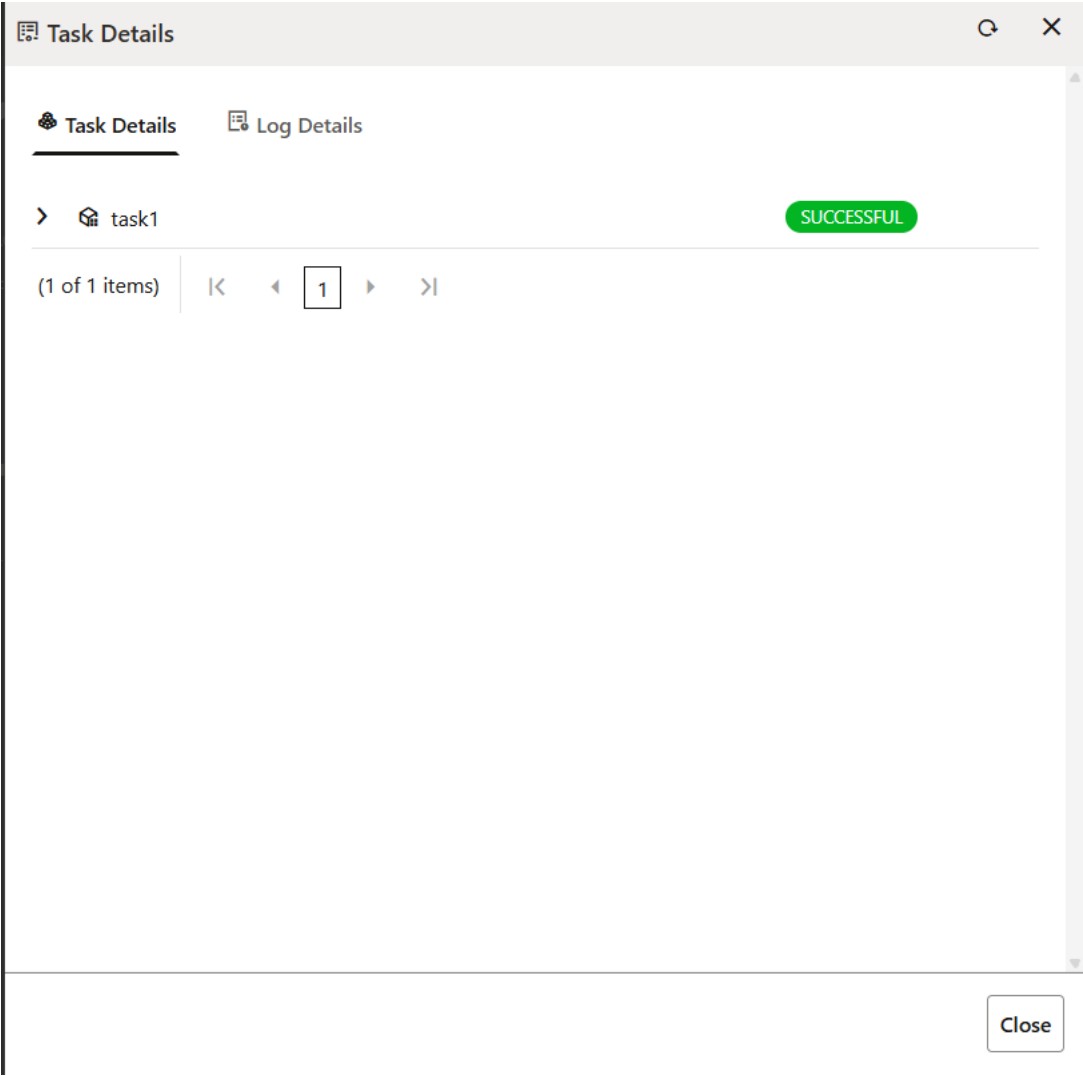
In the Data Population Jobs section, the you can view the output of the population under **Batch Run Identifier** and **Status** columns for the workspaces.

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 [X]

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 [Delete]

Hints ⓘ +

Tables [Required] Source Hint Target Hint [Delete]

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

1 Basic Details

2 Workspace Schema

3 Data Sourcing

4 Metadata Sourcing

5 Validate

Workspace Code: CS Use Template

Purpose: sandbox

User-group ⓘ

User-group: Modeling Approver x Modeling Reviewer x Modeling User x

Type ⓘ Modeling ☒ Simulation ☐

Subtype ⓘ Sandbox Workspace ☒ Production Workspace ☐

Production Workspace ⓘ ☒

Workspace: PROD

Import Archive File
Drag & Drop file here

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.

MMG Schema Sync Utility

The MMG Schema Sync Utility, `sync_source_target_schema_utility.sh`, enables administrators and advanced users to synchronize database schema meta data between a work space's source and target environments.

This tool is essential when modifications, such as new table creation, are performed manually outside OFS MMG, by using tools like SQL Developer, which may cause inconsistencies between the source and target schema definitions and OFS MMG's meta data catalog. By keeping the DMM meta data in sync ensures that subsequent workspace edits and data sourcing operations proceed without errors.

The following is an example of how this works; you had initially sourced **Table1** and **Table2** into your workspace through OFS MMG. Later, you or another team member manually creates a new **Table3** directly in both the source and target schema outside the OFS MMG user interface. If you try to edit your workspace in OFS MMG to bring in **Table3**, the operation will fail because OFS MMG's meta data (managed by DMM) does not recognize that **Table3** already exists in the target. DMM will attempt to recreate it that will in turn cause a conflict.

The MMG Schema Sync Utility solves this by refreshing DMM's meta data, allowing OFS MMG to recognize all the tables that currently present in the target schema, including those that are added manually.

The MMG Schema Sync Utility works in the following way:

1. The utility invokes a backend DMM API.
2. If the run contains no table filter, it scans and re-registers all the tables in the target schema into DMM's catalog.
3. If the run contains a list of tables, it updates only the specified tables.
4. Once the synchronization is performed, these tables are visible and selectable within OFS MMG as if they had originally been sourced there.

Use the MMG Schema Sync Utility

Before you use the MMG Schema Sync Utility, the following prerequisites must be met:

- You must have shell access to the OFS MMG server.
- You require the following information:

- **Workspace name**, for example, WS123
- **Login user name**, for example mmgadmin
- **Locale**, for example, en_US
- **V_SERVICE_ID** - This is the wallet alias that corresponds to the target schema and can be found in the tenant_connection_info table.
- Optional, a space-separated list of tables in the required format for synchronizing specific tables.

Synchronize Tables with the MMG Schema Sync Utility

Use the following commands to synchronize the database schema meta data between a workspace's source and target environments in any of the following scenarios:

- After manually creating or altering tables in the target schema outside OFS MMG.
- Whenever you observe errors related to missing or duplicate objects while editing the workspace or data sourcing in OFS MMG.
- As part of a scheduled reconciliation or maintenance process in environments with frequent manual DBA operations.

1. To synchronize specific tables, run the following code:

```
./sync_source_target_schema_utility.sh WORKSPACE_NAME LOGIN_USER LOCALE  
V_SERVICE_ID -t "<name of tables space separated>"
```

Or

To synchronize all tables, run the following code:

```
./sync_source_target_schema_utility.sh MYWORKSPACE mmgadmin en_US  
MY_SERVICE_ID
```

Note

The correct value for V_SERVICE_ID, the wallet alias, must be determined from your environment's tenant_connection_info table. This might not always be your schema or data store name.

2. After you accept the request, the utility returns a *status: received* message.
3. Check the log file that can be found in mmg-service.logs for further details such as ensuring that the synchronization has completed and to diagnose errors.
4. After running the utility, you will be able to add the newly synchronized tables in OFS MMG through the UI as part of data sourcing or workspace editing.

Limitations and Best Practices

- The utility does not validate your parameters, workspace, service ID, etc.. Incorrect entries will not cause visible errors in the command line, but it will result in sync failure, hence it is recommended to view the log file in mmg-service.logs.
- For best results, double-check all the inputs and consult your system administrator for the correct V_SERVICE_ID.
- Large schema or multiple tables may take time to synchronize.

Troubleshooting

- If your synchronization status displays the *status: received* message but your changes don't appear in OFS MMG, you must check the log file in `mmg-service.logs` for details.
- Verify that the provided `V_SERVICE_ID` matches your intended target schema connection.
- When in doubt, consult the technical team or system administrator for reviewing the log file or further troubleshooting actions.

7

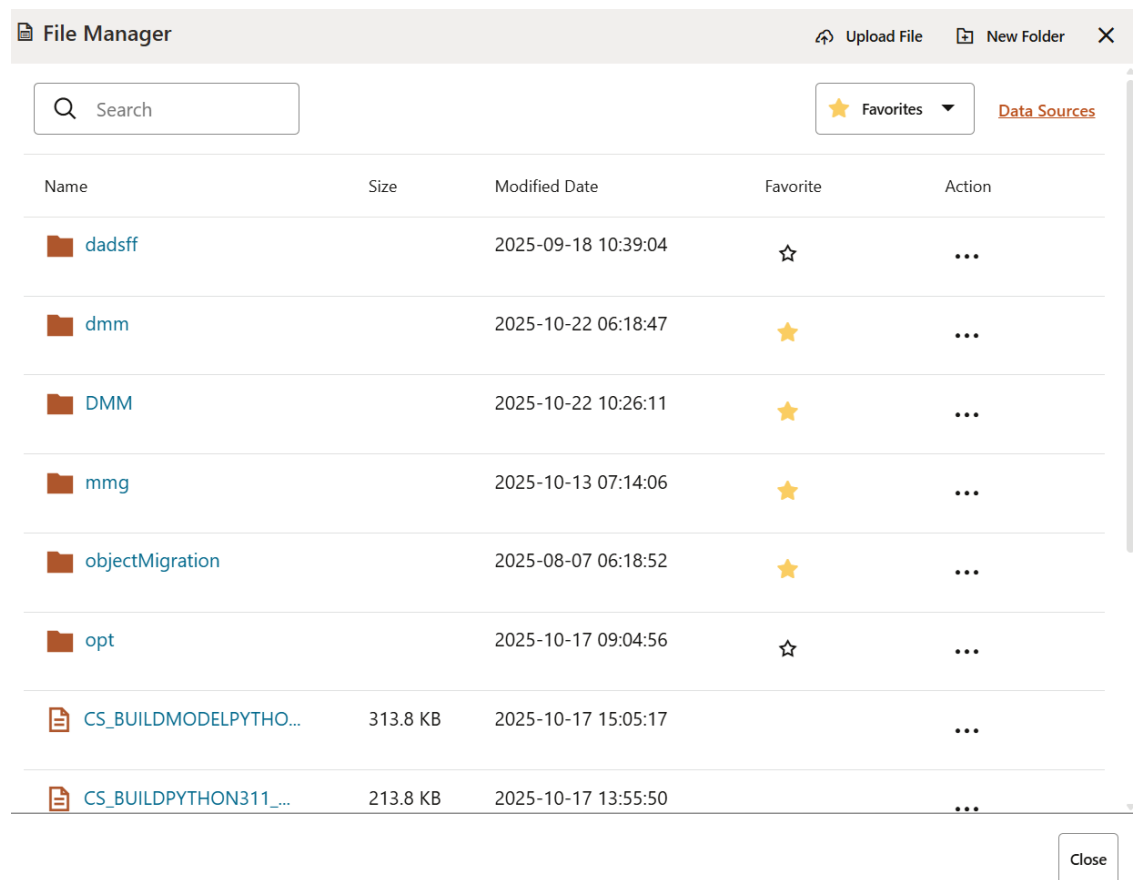
File Manager

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. At the top, there is a header bar with the title "File Manager" and buttons for "Upload File", "New Folder", and a close button. Below the header is a search bar and a "Favorites" dropdown menu. The main area contains a table with the following columns: Name, Size, Modified Date, Favorite, and Action. The table lists several folders and two files.

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		...

A "Close" button is located at the bottom right of the interface.

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.

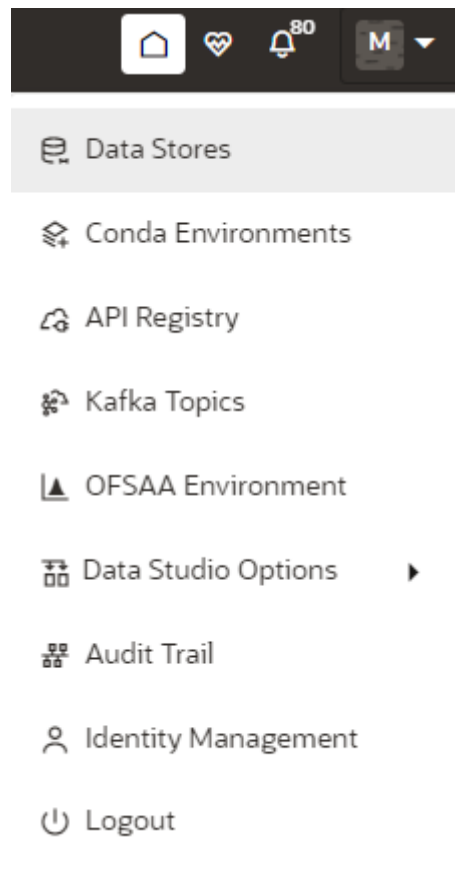
Additionally, workspace-level access controls are also applicable to files/folder. To access files/folders in python see the Access Module for File/Folder Operations section in [Notebook](#).

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.

- **OFSA 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

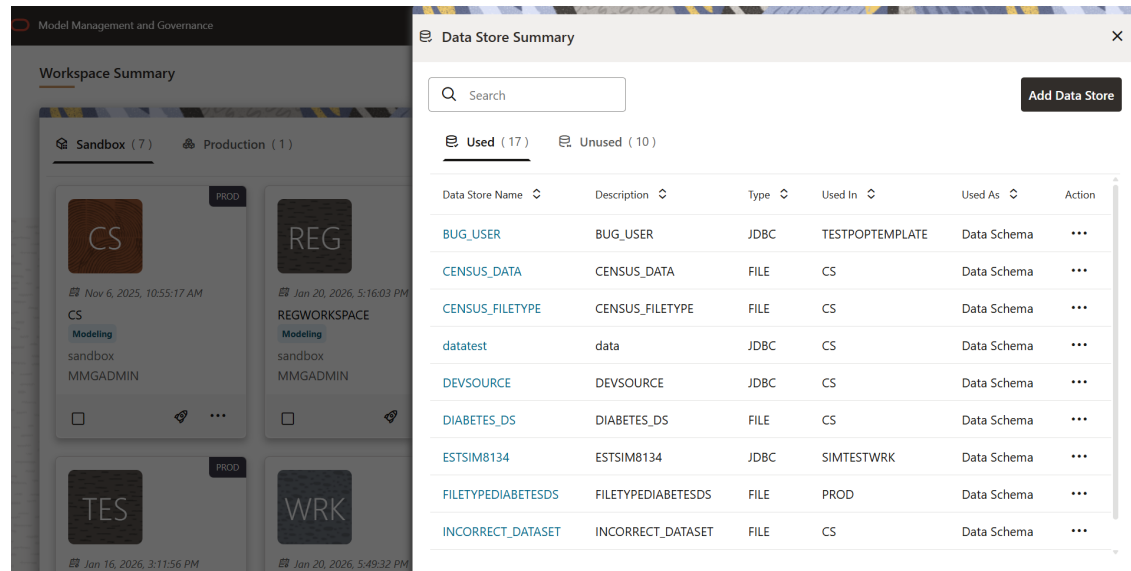
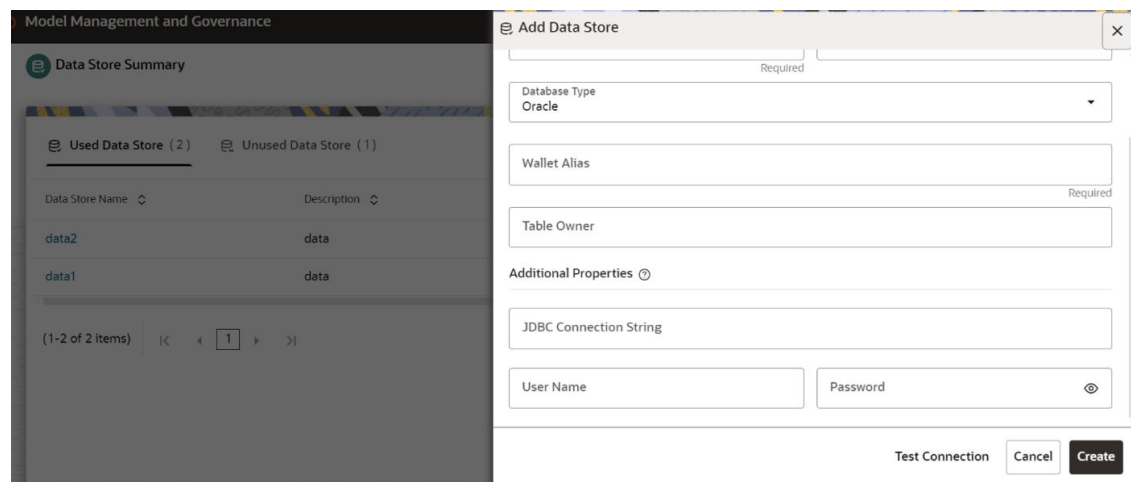
Clicking on the option will navigate you to the specific page. For more details, see [Data Studio Options](#) section.

- **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 drawer displays 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 view the **Data Source Summary** drawer. This window can be accessed and opened from any UI

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

Figure 8-2 The Data Store Summary Drawer**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

Data Store Name

Description

Type
JDBC

Database Type
Oracle

Wallet Alias

Test With Wallet Alias

Table Owner

Additional Properties ?

JDBC Connection String

User Name

Password

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	Enter the type of the database connection. The supported types are JDBC and File. JDBC : If selected, the Database type options Oracle and Hive are displayed. File: If selected, the following options are displayed. 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##/* 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##/* 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. File Type : Select the Type as text or csv format. 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. 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. 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. Derive Schema: It is an optional feature that populates the data grid by analyzing schema details from any data file stored in the above

Table 8-1 (Cont.) Add Data Source

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

Table 8-1 (Cont.) Add Data Source

Field	Description
Database Type	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: • 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. If you select Database Type as Hive (see Figure 11), then following additional fields are displayed to enter details: 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: • Schema population for Hive as target is not supported. • This is applicable only for Sandbox Workspace. For more information on setting the environment, see the Oracle Financial Services Model Management and Governance Installation Guide.

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

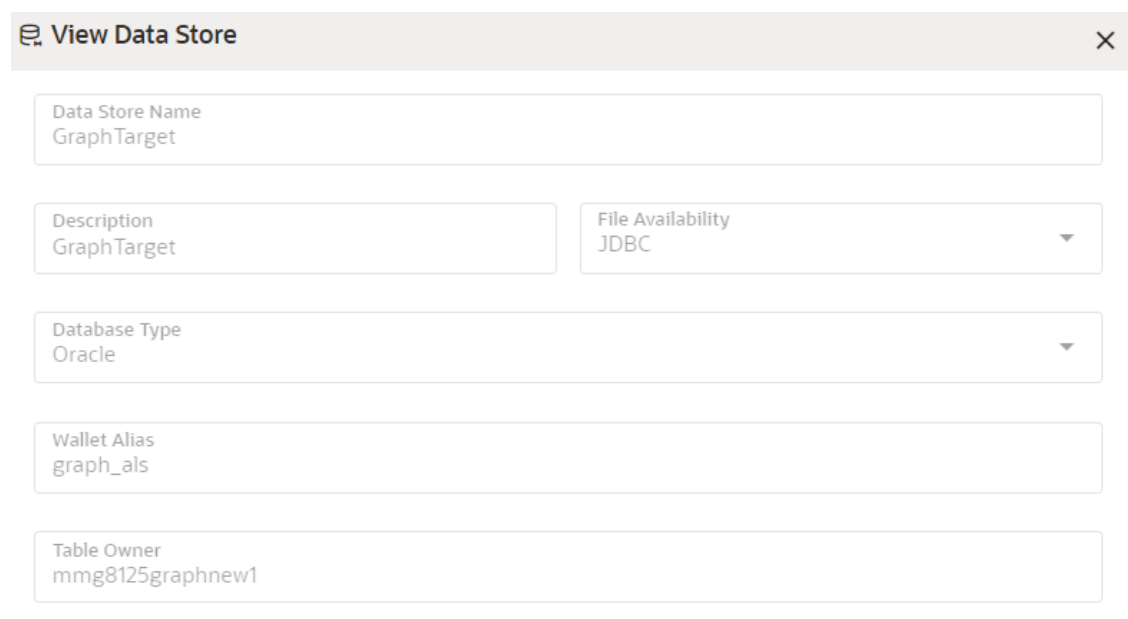
The correct JDBC connection string which the user should provide is:
jdbc:oracle:thin:@sevicehost:port/servicename

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:

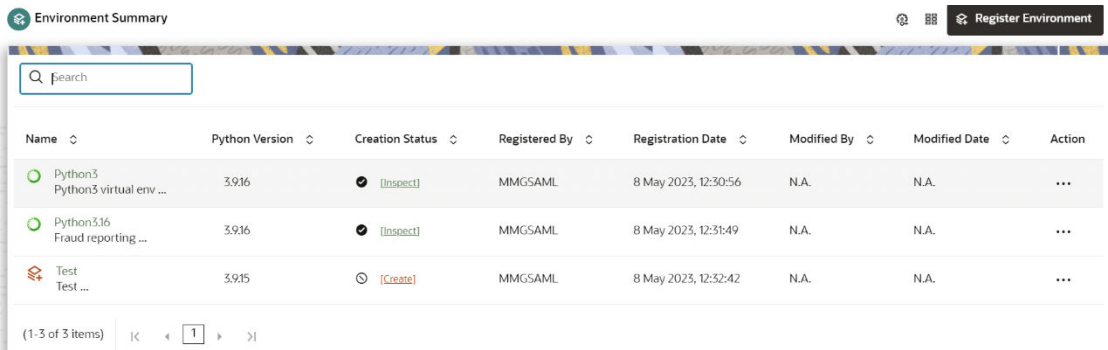
- http: http://www-proxy.idc.oracle.com:80
- https: http://www-proxy.idc.oracle.com:80

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



The screenshot shows the 'Environment Summary' page with a search bar and a 'Register Environment' button. Below is a table with columns: Name, Python Version, Creation Status, Registered By, Registration Date, Modified By, Modified Date, and Action. There are three rows of data.

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, 12:32:42	N.A.	N.A.	...

(1-3 of 3 items) |< < 1 > >|

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 following are the available status that appear in this column: • Inspect: The environment is registered and created. You can perform functions such as Edit/View/Inspect/Clone/Export/Remove for the Inspect status environments. Click this link to view the environment details. For more information, see the Inspect a Conda Environment. Note that you cannot view the log information from this option. If you select the View Logs button from the Action icon, the Environment Logs window appears, informing you that the logs are not available for successfully created environments. • Create: The environment is registered but not created. You can perform functions such as Create/Edit/View/Deregister for the Create status environments • Import: The selected YAML file for the Conda environment has been uploaded and is ready to import. Click the Import link to create this Conda environment in the system. • Failed: This status appears for failed operations. Click this link to open the What's Next? window. The available options in this window are: – Retry: Click this button to open the Create Environment window. Click the Create button to retry the import or create operation by using the same parameters. – View Logs: Click this button to open the Environment Logs window and view the log information. This window displays the complete, detailed logs for the failed operation, enabling you to troubleshoot or share information with the support team.
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.

Table 8-2 (Cont.) Fields and icons on the Sandbox Summary page

Field or Icon	Description
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 Create/ Import/Edit/ View/Inspect/Clone/Export/Remove/Deregister 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

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

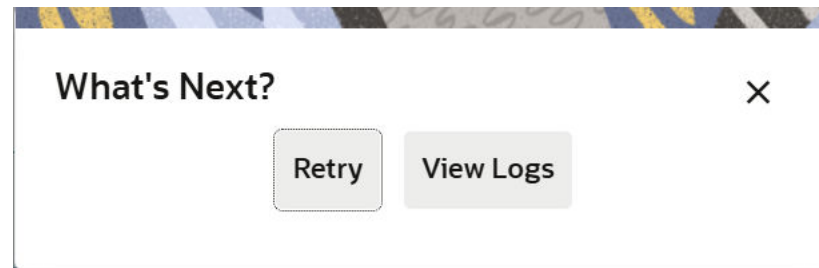
The **Location** of the Conda environment is displayed. This field is greyed out and cannot be modified.

6. From the **How do you want to register the environment?** section, select either of the following options:
 - **Create from Version:** This option is selected by default. Select a Python version number from the **Version** drop-down list.
If this option is selected and the **Register** button is clicked, then the creation operation is performed asynchronously in the background. This enables you to interact with other UI features in the application as the screen is not locked. Real-time status updates, operation histories, and context-specific actions are now provided. An operation status or progress area is available, where each in-progress action is listed with states such as *Queued*, *In Progress*, *Completed*, or *Failed*. Once this operation is completed, the

status is updated on the **Environment Summary** page. The following are the status that are displayed in the **Creation Status** column adjacent to an environment:

- **Inspect:** Click this link to view the environment details. For more information, see the [Inspect a Conda Environment](#). Note that you cannot view the log information from this option. If you select the **View Logs** button from the **Action** icon, the **Environment Logs** window appears, informing you that the logs are not available for successfully created environments.
- **Failed:** This status appears for failed operations. Click this link to open the **What's Next?** window.

Figure 8-8 What's Next? Window



The available options in this window are:

- * **Retry:** Click this button to open the **Create Environment** window. Click the **Create** button to retry the operation.
- * **View Logs:** Click this button to open the **Environment Logs** window and view the log information.

Multiple environment creation tasks can be started in parallel and each task is tracked individually in the status or progress panel. This enables you to receive further details about the operation.

Note

The following are the known issues and limitations of this feature:

- * The log information is only available for failures. For successful operations, the **Inspect** option allows you to view the summary details.
- * When multiple creation operations run in parallel, there may be a delay or omission in the live status updates for a few entries until the UI is manually refreshed. This is a known issue.

Figure 8-9 Create from Version Option

How do you want to register the environment?

☒ Create from Version ☐ Import from YAML

Version
3.12.11 (Recommend... ▼)

- **Import from YAML:** Select this option if you want to import the metadata from a YAML file. Click **Import Archive File** or drag and drop the desired .yaml/.yml file from your local system into the Import Archive File section. You must ensure that the Python version is included in the .yaml/.yml file.

Figure 8-10 Import form YAML Option

How do you want to register the environment?

☐ Create from Version
 ☒ Import from YAML

Import File

Users to verify that the environment name matches the one in the imported .yaml/.yml file. Ensure the Python version is explicitly defined in the YAML to avoid compatibility issues. If a version constraint (e.g., >=3.9,<3.12) is specified, the actual installed version will be shown after the environment create completes.



Import Archive File
Drag & Drop file here

After attaching the .yaml/.yml file in the registration dialog, the .yaml/.yml file content is displayed in a read-only, scrollable preview area labeled as **File Details** along with the file name and size. This preview allows you to fully inspect the uploaded environment specification before submission. If the attached file is removed or replaced, the preview disappears or updates automatically to reflect the new file. You cannot modify the content of the file in this preview area. Any desired modifications must be done in the file and then the modified file must be reattached.

Note

The following limitations are present in this version of the application for this feature:

- Currently, this feature is only available during the registration process. This preview feature will be available in the **View** or **Edit** windows in future releases.
- In the **Edit** screen, the preview of the attached .yaml/.yml file is currently not displayed. Also note that if updates are made in any of the fields in the **Edit** window, then the changes can be saved without reuploading a new .yaml/.yml file. The file preview and validation are known issues that will be fixed in future releases.

You must review the .yaml/.yml file details in the **File Details** section to confirm that the required metadata, such as environment name and Python version, are present before registering the environment. Post-registering, the file content will not be displayed in the **View** and **Edit** windows. Hence, ensure that you keep your local copies of the uploaded files for audit.

OFS MMG now processes Conda environment imports asynchronously in the background. When you begin an import operation, the **Environment Summary** page automatically displays the current status of the import. This allows you to continue using the application while your Conda environment is being imported.

The **Import** status on the **Environment Summary** keeps you informed so that you do not have to wait or keep the page open.

Until the environment is imported, the actions that you can perform in the **Action** icon on the **Environment Summary** page, such as **Edit**, **View**, and **Deregister** will be temporarily disabled.

Note

If you do not see the **Creation Status** on the **Environment Summary** page update automatically, you can refresh this page to view the latest status.

7. 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 the **Create** link in the **Creation Status** column of a desired environment to create the Conda environment, or, you can also click the **Action** icon adjacent to a desired environment and then click **Create**. Note that this option is only available if the status in the **Creation Status** column is *Create*.

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.

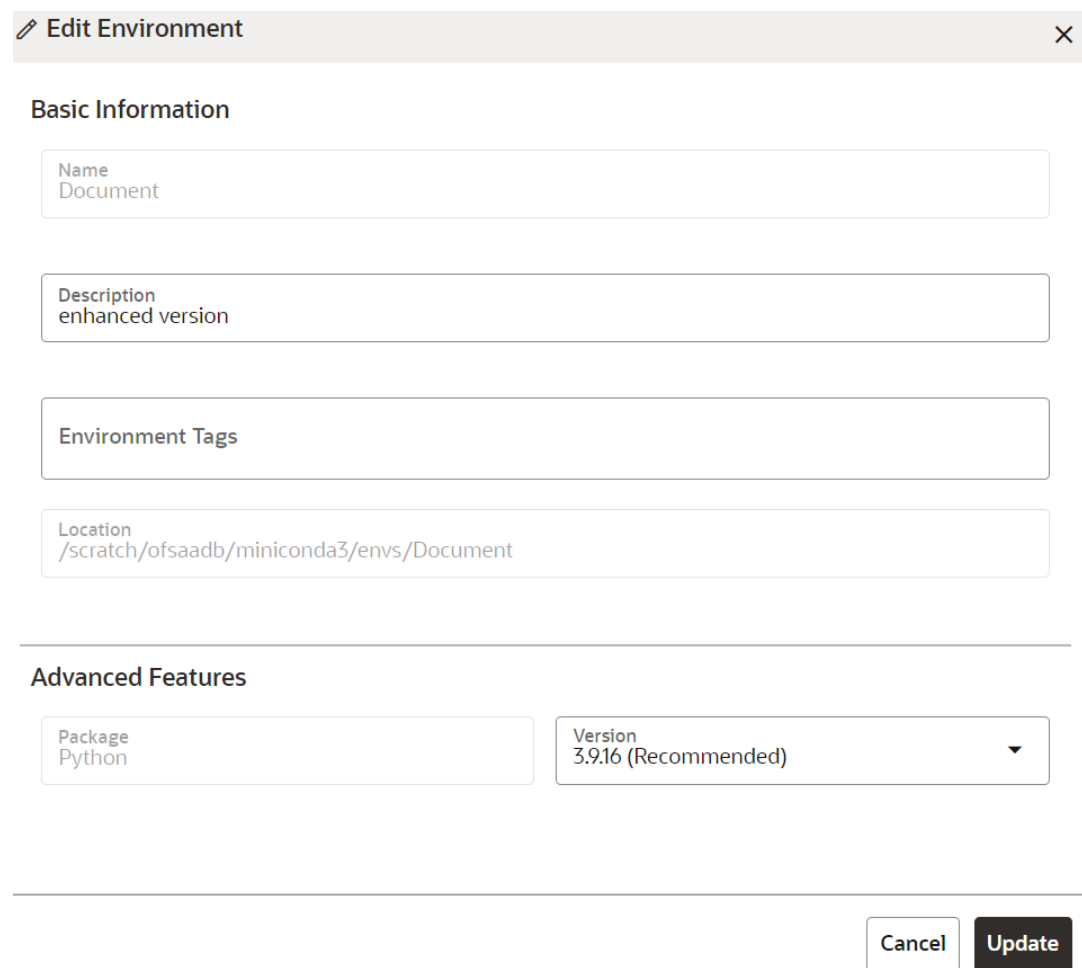
The **Edit Environment** page is displayed.

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 8-11 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 8-12 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 8-13 Conda environment details

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 8-14 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 

Cancel

 Clone

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:

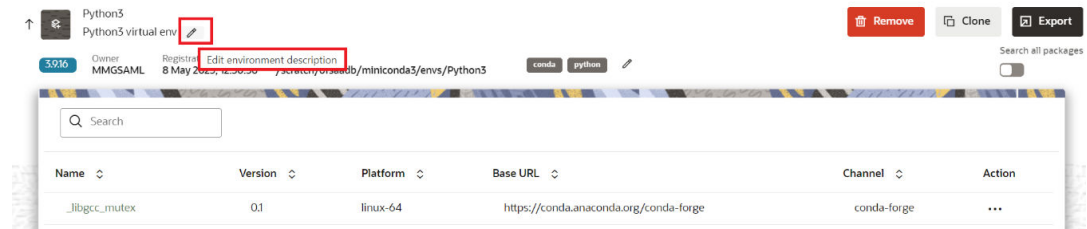
- In the Conda details page, click **Export**.
The Conda environment is exported in **yml** format.

Edit a Conda Environment Description

To edit a Conda environment description:

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

Figure 8-15 Conda details page



2. 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:

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

Figure 8-16 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 8-17 Package Properties



Key	Value
build_string	<i>conda_forge</i>
Key	Value
dist_name	<i>_libgcc_mutex-0.1-conda_forge</i>
Key	Value
base_url	<i>https://conda.anaconda.org/conda-forge</i>
Key	Value
channel	<i>conda-forge</i>
Key	Value
name	<i>_libgcc_mutex</i>
Key	Value
version	<i>0.1</i>
Key	Value
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 8-18 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 

Cancel

 Clone

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:

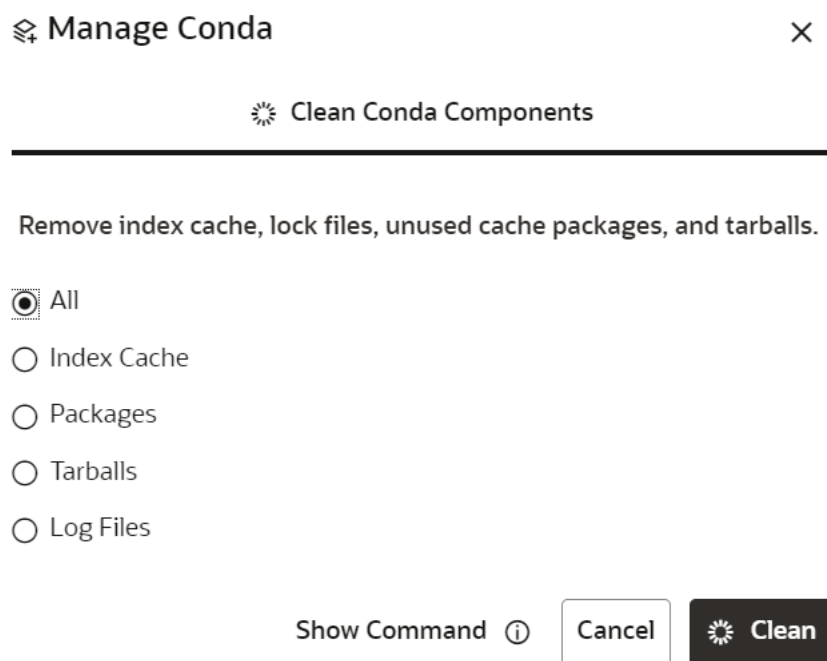
1. Click the **Conda Environments** icon to navigate to the **Environment Summary** page from any other window in the application.

This page displays the Conda environments in a tabular format.

2. Click the **Conda Settings** icon.

The **Manage Conda** screen is displayed.

Figure 8-19 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 8-20 Conda Properties

Conda Properties		X
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

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](https://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 a simple [YAML syntax](#).

Example:

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

Alternatively, you can open a text editor such as Notepad on Windows, or 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 edit it 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 `.condarc` 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 section 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 the user-specific configuration files should be stored or defined by following the XDG Base Directory Specification (XDG BDS).
- Default to `$HOME/.config` must be used. `CONDA_ROOT` is the path for your base conda installation.
- `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`).

Channels

You can create or configure the `.condarc` file (the global configuration file for Conda) by using the following example:

```
channels:
- conda-forge
- defaults

channel_alias: https://oracle.repo

default_channels:
- https://oracle.repo

ssl_verify: False

proxy_servers:
http: http://www-proxy.us.oracle.com:80
https: http://www-proxy.us.oracle.com:80
```

Channel Locations

When you list the channel locations in the `.condarc` file, it overrides the Conda defaults that in turn cause Conda to only search the listed channels in a given order.

To automatically include all default channels, use `defaults`. Non-URL channels are interpreted as an Anaconda.org user or as per the organization names. Perform the steps documented in the [Set a channel alias \(channel_alias\)](#) to modify the `channel_alias`. The default is `defaults`.

Example

```
channels:
- <anaconda_dot_org_username>
- http://some.custom/channel
- file:///some/local/directory
- defaults
```

Add a `.condarc` file in the root directory of the environment (or when you are using `conda config`, use the `--env` option) to select channels for a single environment. For example, you have installed Miniconda with Python 3 in your home directory and the environment is named `newenvironment`, the path may be:

```
~/miniconda3/envs/newenvironment/.condarc
```

Default channels (default_channels)

The defaults channel points to several channels in the repo.anaconda.com repository, but a new list of default channels are set if the `default_channels` is defined. This is especially useful for airgapped and enterprise installations:

Setting both the [channel_alias](#) and `default_channels` by an administrator, ensures that all users only pull packages from an on-premises repository, for example:

`default_channels:`

- `http://some.custom/channel`
- `file:///some/local/directory`

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 is shown in this. 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 as `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. The following 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

The API Registry feature enables you to register APIs above the workspace level. Configured APIs can be mapped to multiple workspaces and the mapped APIs will be available for use in model pipelines. The **API Registry** drawer can be accessed by clicking the drop-down arrow that is adjacent to the user name on the upper-right corner of the screen, and then clicking **API Registry** from the drop-down list. This drawer can be accessed from any page of the application UI.

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

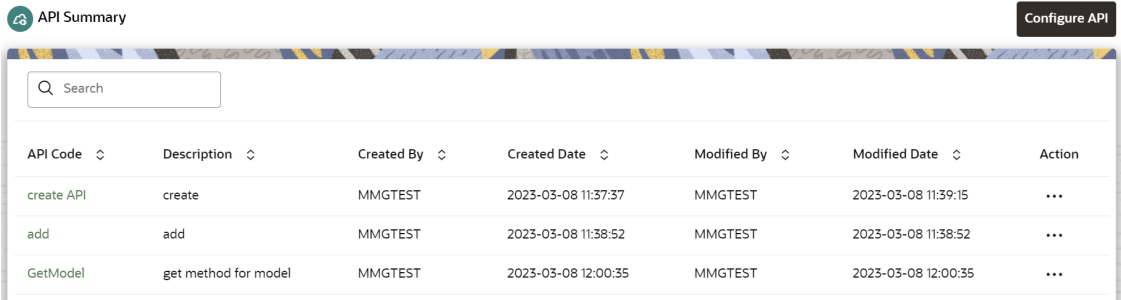
Since variables are fetched during run time, it will be available in notebook or execute screen only after the first execution. This is the default behavior of all programmatic dynamic forms in Datastudio.

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 8-21 API Summary page



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

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

Figure 8-22 Configure API Window

Configure API ×

Code

API Type
Default

Description

Request Type
REST

Method Type
POST

Url

Required

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

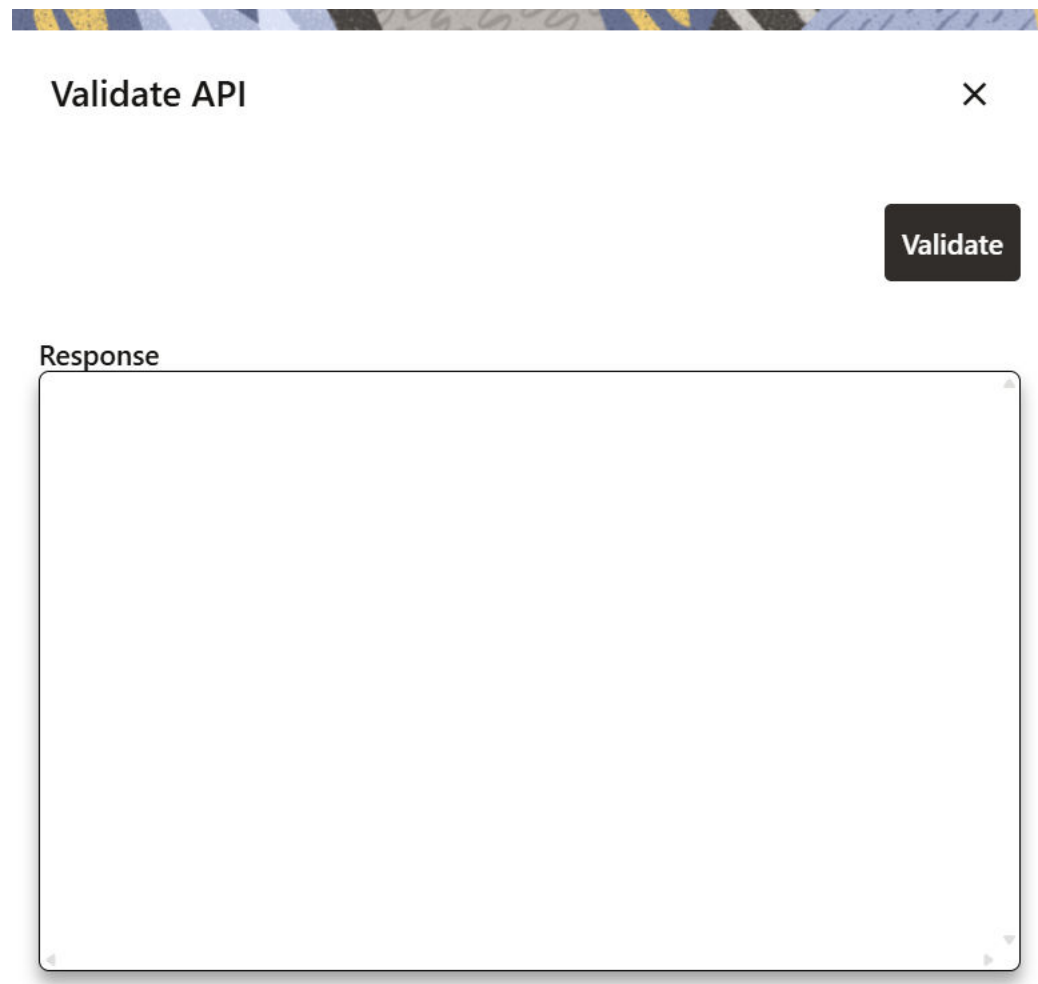
Cancel

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 8-23 View API Details

View API Details [X]

Code create API	Request Type REST
Description create	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

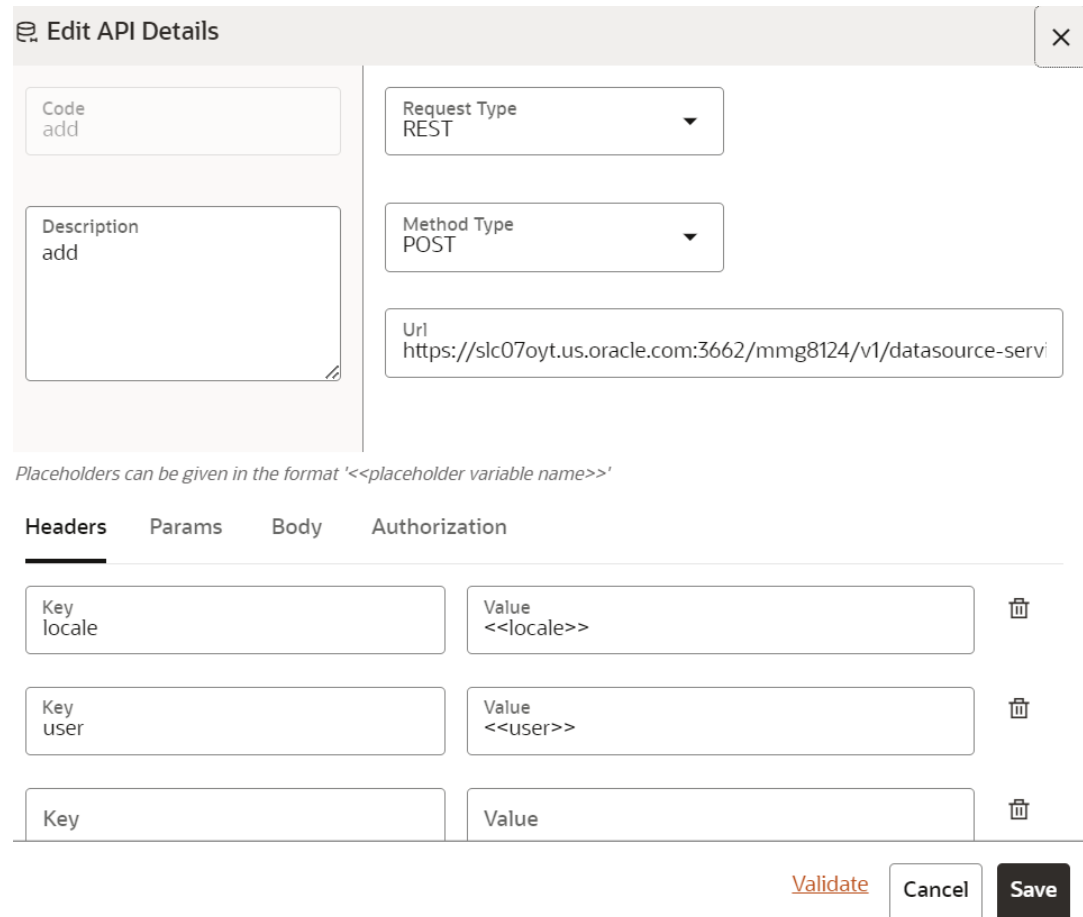
To edit an API configuration:

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 8-24 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.

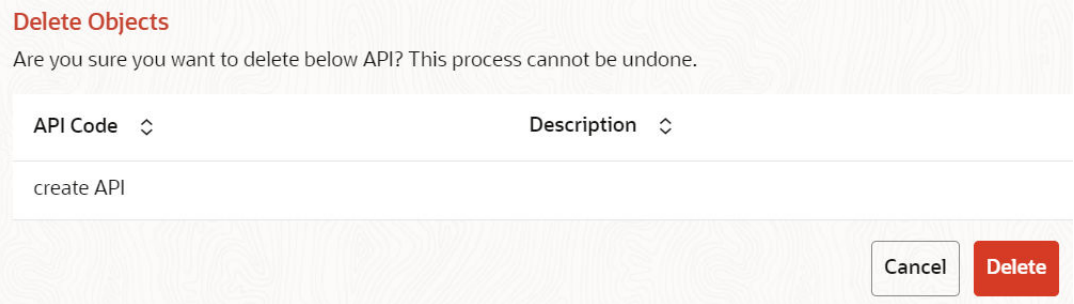
3. Click **Save**.

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 8-25 Delete Objects screen



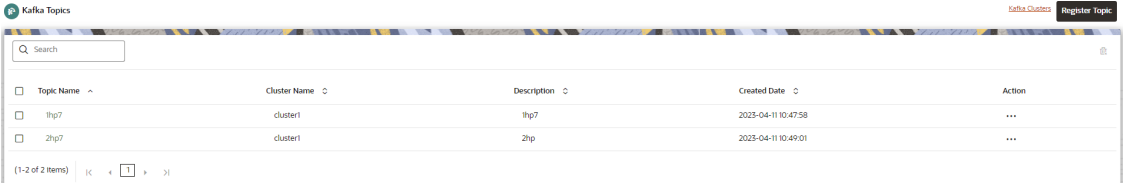
3. Click **Delete**.
- The API Registry is deleted.

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 8-26 API Summary page



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:

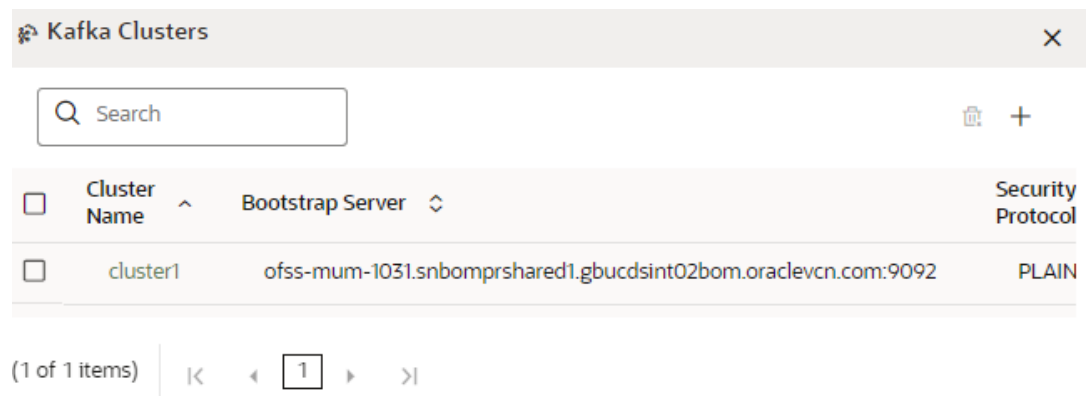
1. Click **Kafka Topics** to navigate to **Kafka Topics** page from any other window in the application.

This page displays the Kafka Topics in a tabular format.

2. Click **Kafka Clusters** option.

The **Kafka Clusters** page is displayed.

Figure 8-27 Kafka Clusters page

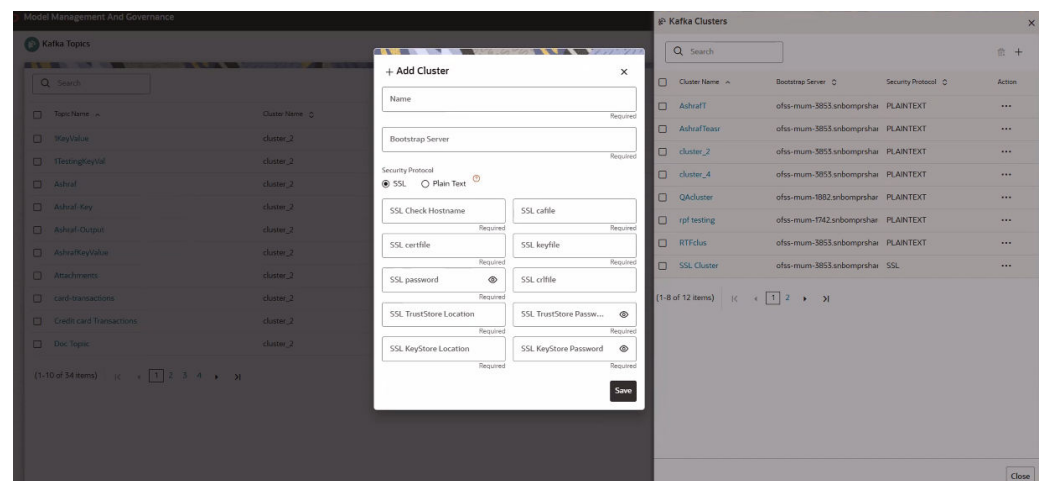


3. Click **Add**.

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 8-28 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.

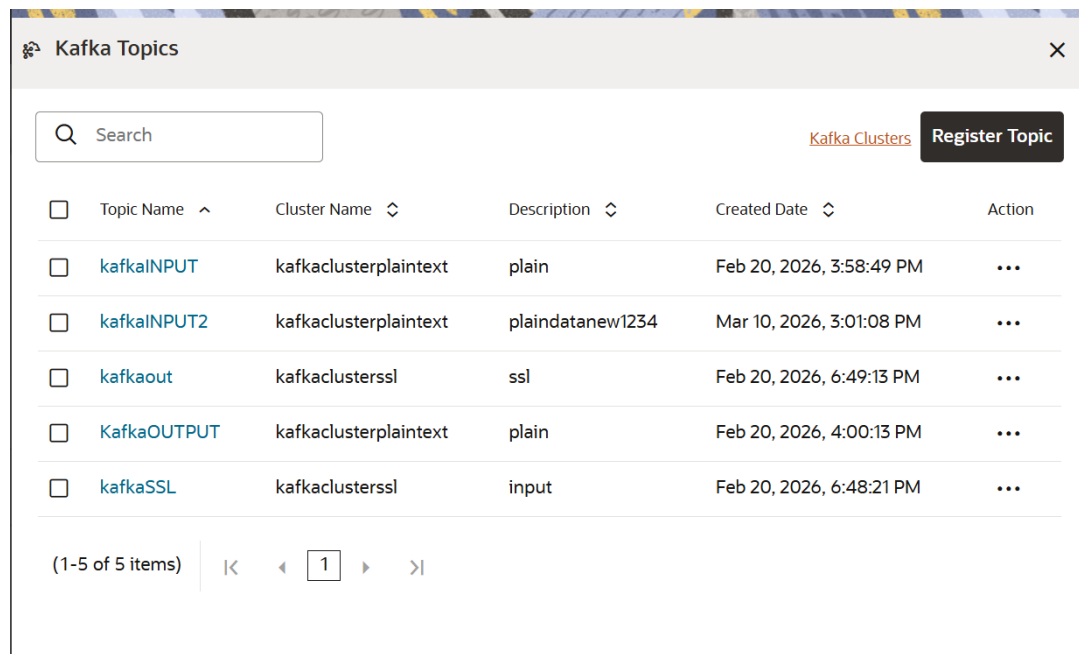
4. Click **Delete**.

The Kafka Cluster is deleted.

Register a Kafka Topic

To register a Kafka topic perform the following steps:

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



The screenshot shows a window titled "Kafka Topics" with a search bar and a "Register Topic" button. Below is a table listing five topics.

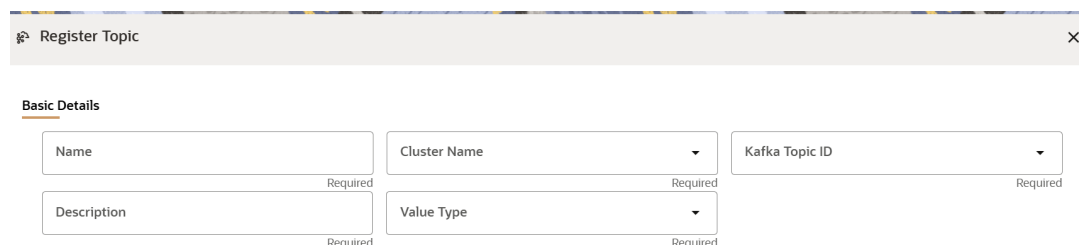
☐	Topic Name ^	Cluster Name ^	Description ^	Created Date ^	Action
☐	kafkaINPUT	kafkaclusterplaintext	plain	Feb 20, 2026, 3:58:49 PM	...
☐	kafkaINPUT2	kafkaclusterplaintext	plaindatanew1234	Mar 10, 2026, 3:01:08 PM	...
☐	kafkaout	kafkaclusterssl	ssl	Feb 20, 2026, 6:49:13 PM	...
☐	KafkaOUTPUT	kafkaclusterplaintext	plain	Feb 20, 2026, 4:00:13 PM	...
☐	kafkaSSL	kafkaclusterssl	input	Feb 20, 2026, 6:48:21 PM	...

At the bottom, it shows "(1-5 of 5 items)" and pagination controls with "1" selected.

2. Click **Register Topic**.

The **Register Topic** window appears.

Figure 8-29 Register Topic Window



The screenshot shows the "Register Topic" window with the "Basic Details" section. It contains five required fields:

- Name**: Text input field, Required.
- Cluster Name**: Drop-down menu, Required.
- Kafka Topic ID**: Drop-down menu, Required.
- Description**: Text input field, Required.
- Value Type**: Drop-down menu, Required.

3. Populate the form:

- **Name:** Enter a name in this field for the Kafka topic. This is a mandatory field.
- **Cluster Name:** Select the name of the cluster from the drop-down list. When the name of the cluster is selected, the Kafka Topic IDs related to the selected cluster are populated in the **Kafka Topic ID** field. This is a mandatory field.
- **Kafka Topic ID:** Select a topic ID from the drop-down list. This is a mandatory field.
- **Description:** Enter a description for the Kafka topic.
- **Value Type:** Select a value from the drop-down list.
- **Python Runtime:** This option enables you to select a Conda environment for a Kafka topic that can then be used in the Pipelines feature.
- **Record Schema:** 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.

4. 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 8-30 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 8-31 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 8-32 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 8-33 The Audit Trail Page

 Audit Trail

Search	From 	To 	Date Search	 Refresh	
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 ...

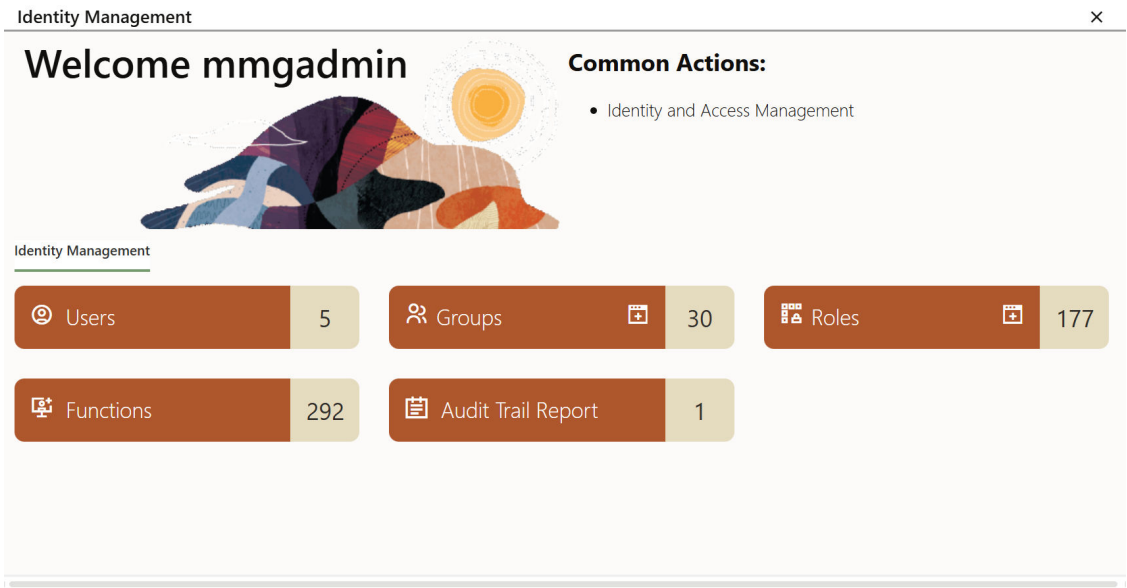
Select the **Audit Timeline View** icon, adjacent to the **Refresh** icon, to view the listed sequence of actions that have been performed. Additionally, the Audit Table view (graphical representation) can also be viewed from the main **Audit Trail** page. Models can be searched by

- **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.

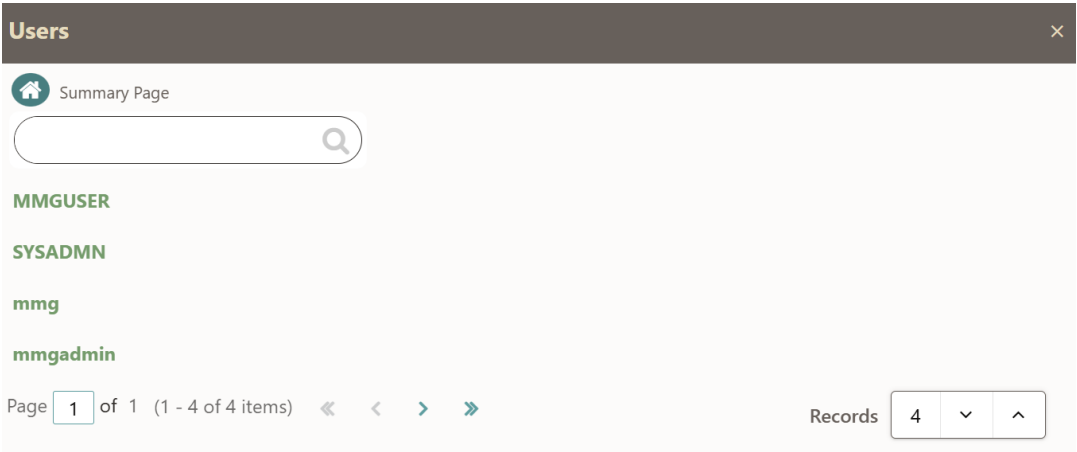
Figure 8-34 The Identity Management Page



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 8-35 Identity Management Users



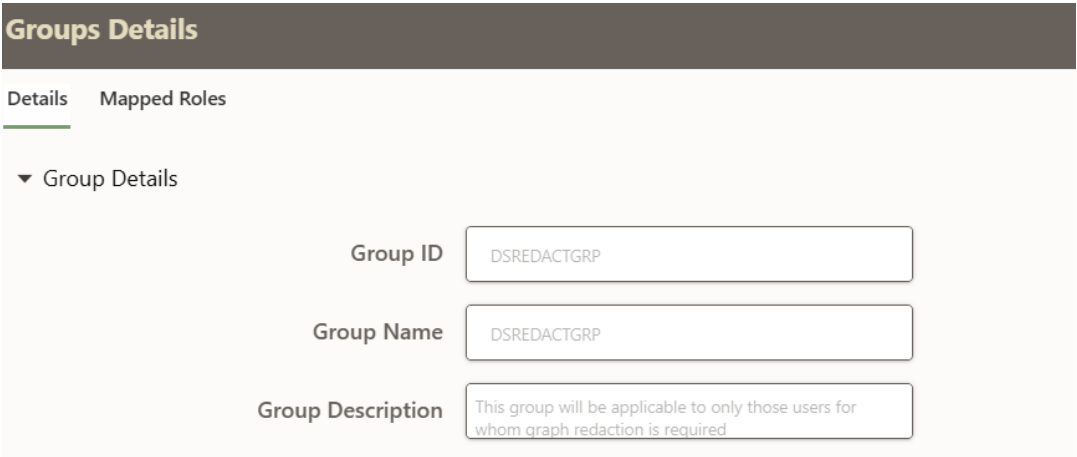
- **Groups:** The application admin users contain groups. This page displays the groups that are assigned to a user.

Figure 8-36 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 8-37 Group Details - Details

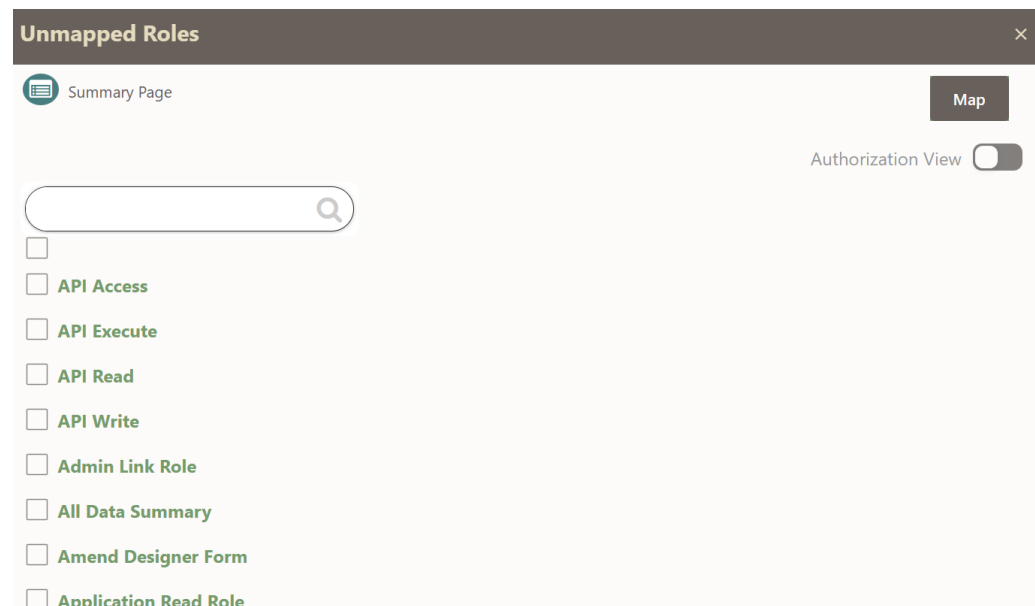


Additionally, click the **Mapped Roles** tab to view details about roles that are mapped to specific groups.

Figure 8-38 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 8-39 Unmapped Roles

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

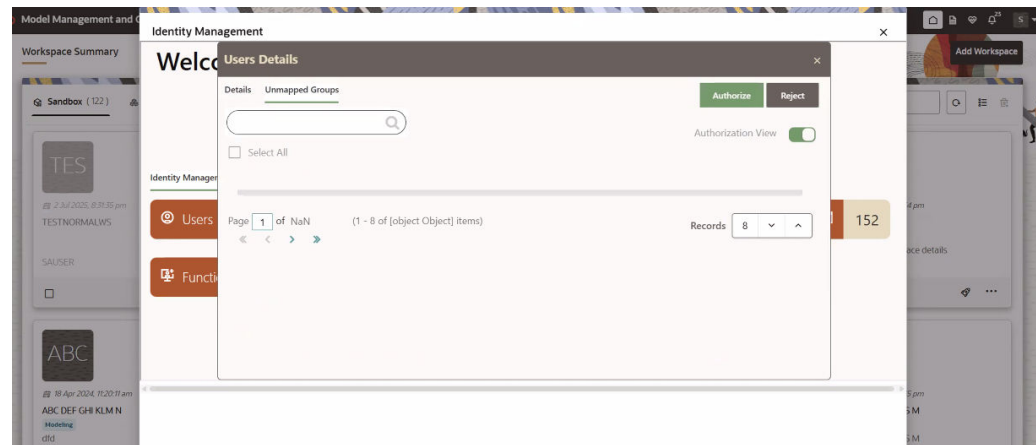
Note

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 8-40 Authorization View screen



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

Figure 8-41 Roles



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

Figure 8-42 Role Details

The screenshot shows a web interface 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" with the value "APIACCESS", "Role Name" with the value "API Access", and "Role Description" with the value "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.

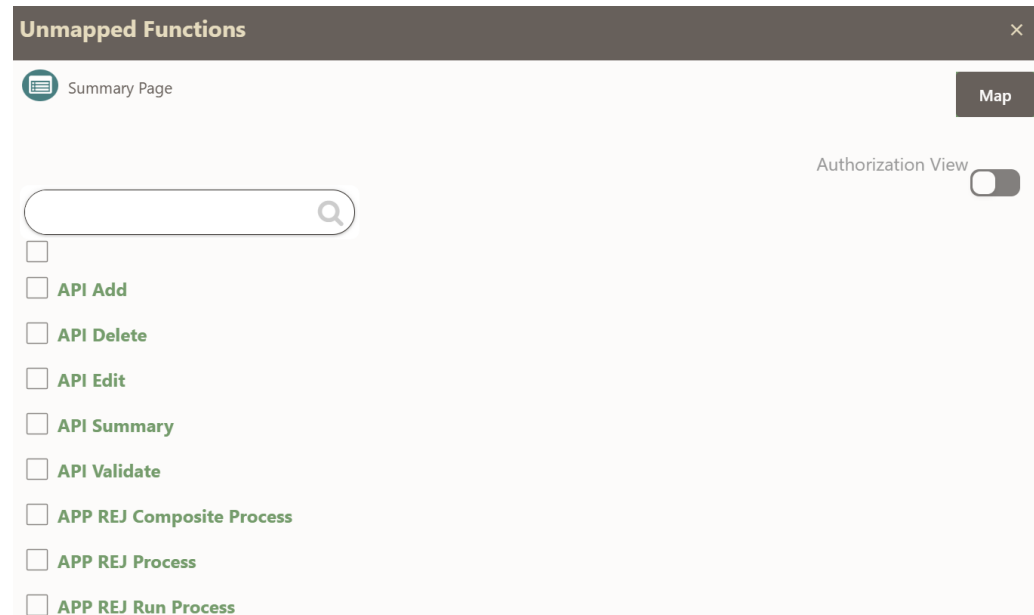
Figure 8-43 Mapped Functions

The screenshot shows the "Mapped Functions" tab selected in the "Roles Details" interface. The "Details" tab is also visible. In the top right corner, there are two buttons: "New Mapping" (green) and "Unmap" (dark grey). Below the tabs, there is a section titled "Summary Page" with a search bar and a magnifying glass icon. Below the search bar, there are two checkboxes: the first is unchecked, and the second is checked and labeled "API Summary". Below the checkboxes, there is a pagination bar showing "Page 1 of 1 (1 - 1 of 1 items)" with navigation arrows. To the right of the pagination bar, there is a toggle switch labeled "Authorization View" which is currently turned off. Below the toggle switch, there is a "Records" section with a dropdown menu showing "1" and up/down arrows.

All the roles have specific functions mapped to it.

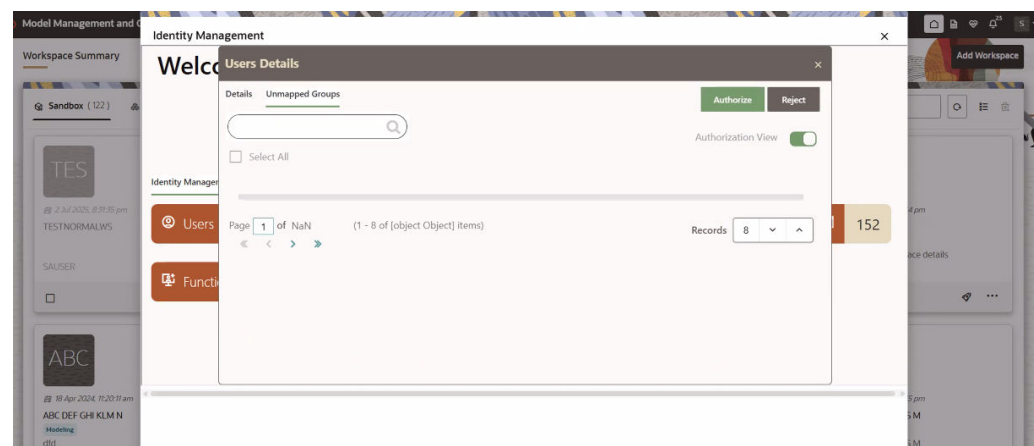
- **New Mapping:** Click **New Mapping** to map a role to a function via the **Unmapped Functions** page. On this page, select the desired functions and click **Map**. You can also search for a specific function by entering the name of the function in the **Search** field and then clicking the search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

Figure 8-44 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 8-45 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 8-46 Functions

Functions [X]

Summary Page

[Search Bar]

- API Add
- API Delete
- API Edit
- API Summary
- API Summary
- API Validate
- APP REJ Composite Process
- APP REJ Process

Page 1 of 32 (1 - 8 of 249 items) [Navigation Icons]

Records 8 [Dropdown]

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

Figure 8-47 Audit Trail Report

Audit Trail Report [X]

Search and Filter [Reset] [Search]

User Name [All] [Search Icon]

Action [All] [Dropdown Arrow]

From Date [Calendar Icon]

To Date [Calendar Icon]

Action Detail [Search Code...]

- **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.

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.

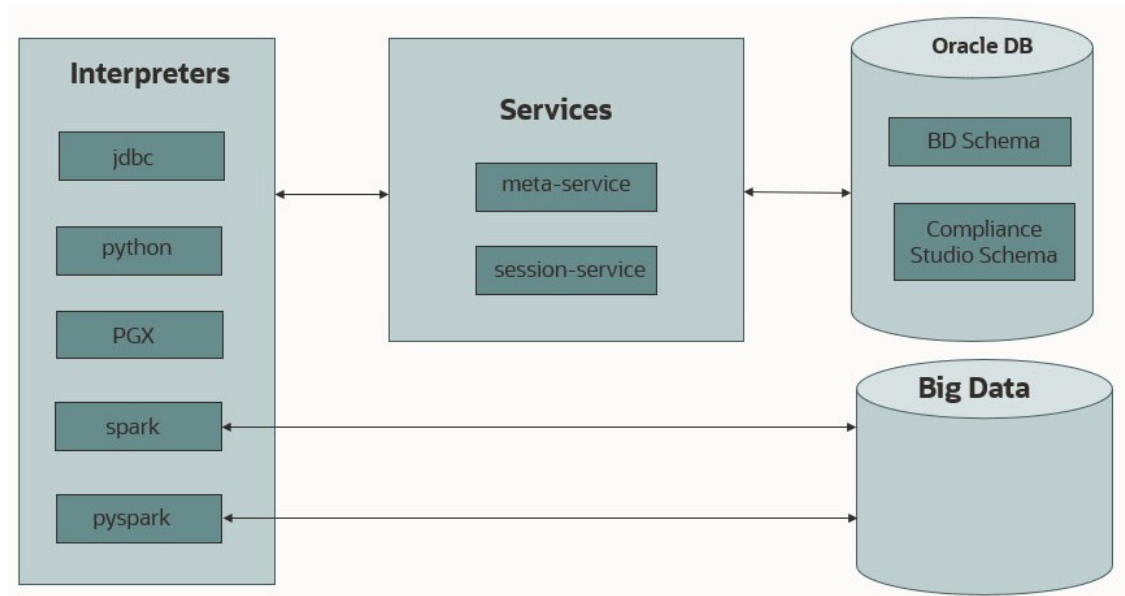
In OFS MMG, Interpreters are used in Notebooks to execute code in different languages. Each Interpreter has a set of adjusted and applied properties across all notebooks. For example, using the python-interpreter makes it possible to change between versions, whereas the jdbc-interpreter offers to customize 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.

Note

By default, the interpreter configuration is refreshed from the filesystem `interpreter.json` the next time you restart the application. If you update `interpreter.json` on the filesystem, the changes take effect after the restart. Changes made only in the UI are not overwritten when you restart (overwrite is driven only by the filesystem JSON).

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

Figure 9-1 Examples of Interpreters



① Note

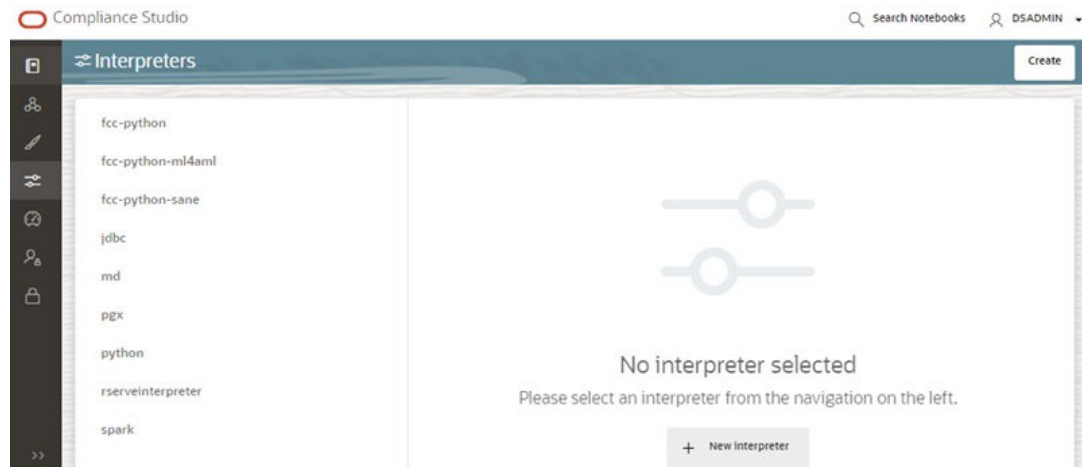
From OFS MMG Release 26.0.0 onwards, the shell interpreter capability (python.shell) and related configurations are fully removed from the system. Executing shell commands via the Python shell is no longer supported. There is no option to enable this functionality in this release or future releases.

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:

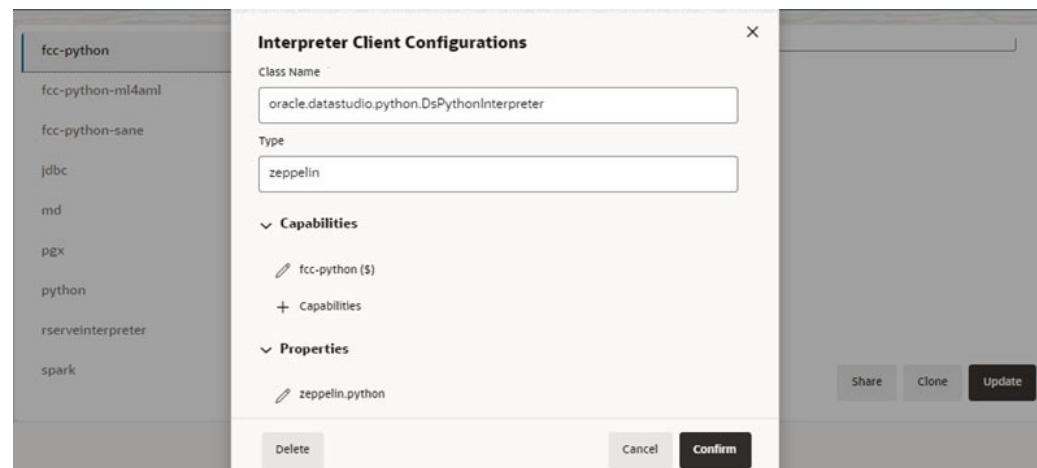
1. On the **Workspace Summary** page, select Launch workspace to display the **CS Production workspace** window.
2. Click the **User Profile** drop-down list and select **Data Studio Options** widget. The following options are available:
 - Interpreters
 - Tasks
 - Permissions
 - Credentials
 - Templates
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

The screenshot shows a 'Properties' dialog box with a close button (X) in the top right corner. The dialog contains several input fields and a 'Type' label. The fields are labeled as follows:

- Key**: A text input field containing 'zeppelin.python'.
- Environment Name**: An empty text input field.
- Property Name**: A text input field containing 'zeppelin.python'.
- Default Value**: A text input field containing '/scratch/fccstudio/CS_81210_81240_UP/OFS_COMPLIAI'.
- Description**: A text input field containing 'Python directory. It is set to python by default.(assume p'.
- Type**: A label positioned above three buttons: 'Delete', 'Cancel', and 'Confirm'.

The 'Delete' button is light gray, while 'Cancel' and 'Confirm' are dark gray.

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

Interpreter Client Configurations

Class Name

Type

✓ **Capabilities**

✓ **Properties**

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.

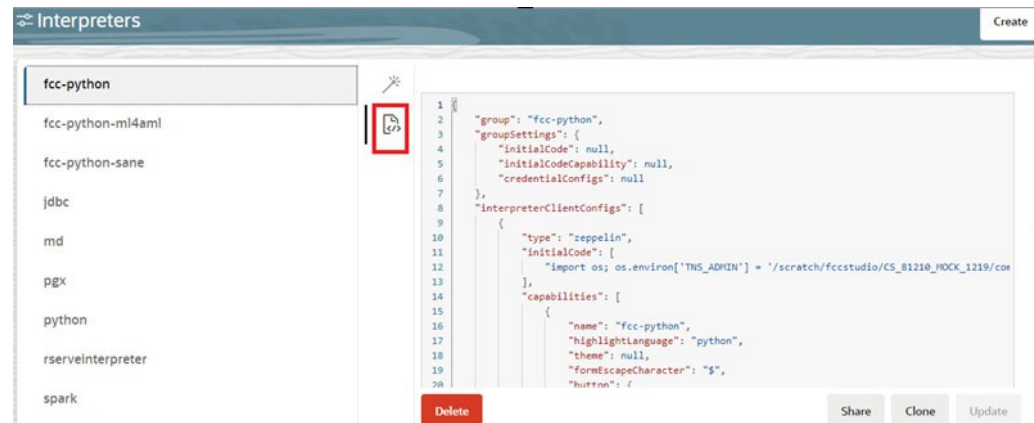
Credentials

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	The python interpreter is used to write Python code in a notebook to analyze data from different sources, machine learning, artificial intelligence, etc. The python interpreter uses a python conda environment. OFS MMG comes with predefined conda environments as follows: • default_<CS version> • ml4aml_<CS version> • sane_<CS version> Before executing any python notebooks, you need to attach the conda environment using drop-down list.

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

Interpreters	Description
jdbc Interpreter	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. 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.
jdbc 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.
md Interpreter	The md interpreter is used to configure the markdown parser type. This Interpreter displays text based on Markdown, which is a lightweight markup language. The connection does not apply to this Interpreter.
pgql Interpreter (part of PGX interpreter)	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-python (part of PGX interpreter)	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.
pgx-algorithm Interpreter (part of PGX interpreter)	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.

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 an 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/fccstudio/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

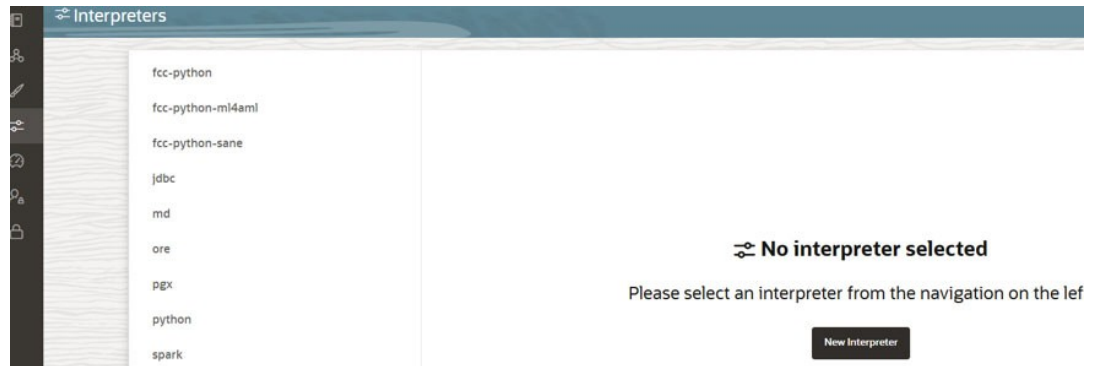
This section is applicable only when `zeppelin.interpreter.output.limit` is unavailable in the Python Interpreter.

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

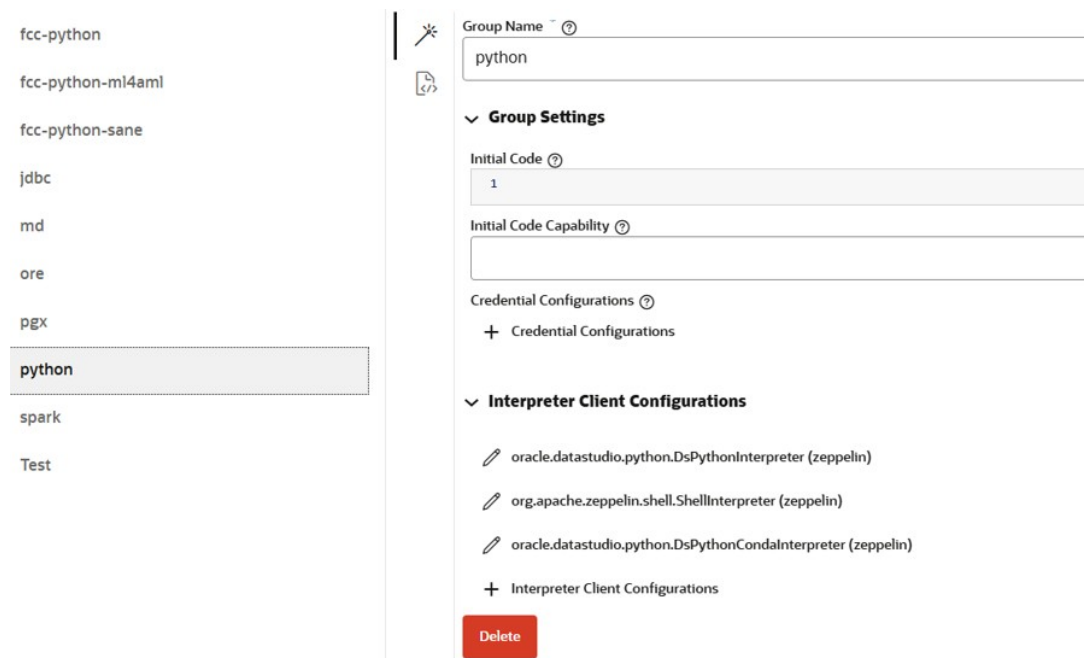
OFS MMG UI

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.

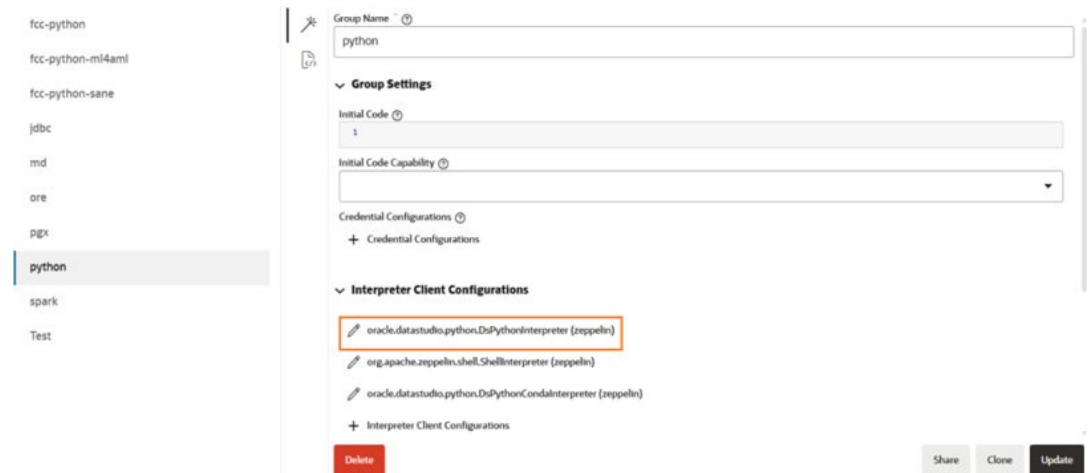
Figure 9-7 Interpreters

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

Figure 9-8 Python Interpreter

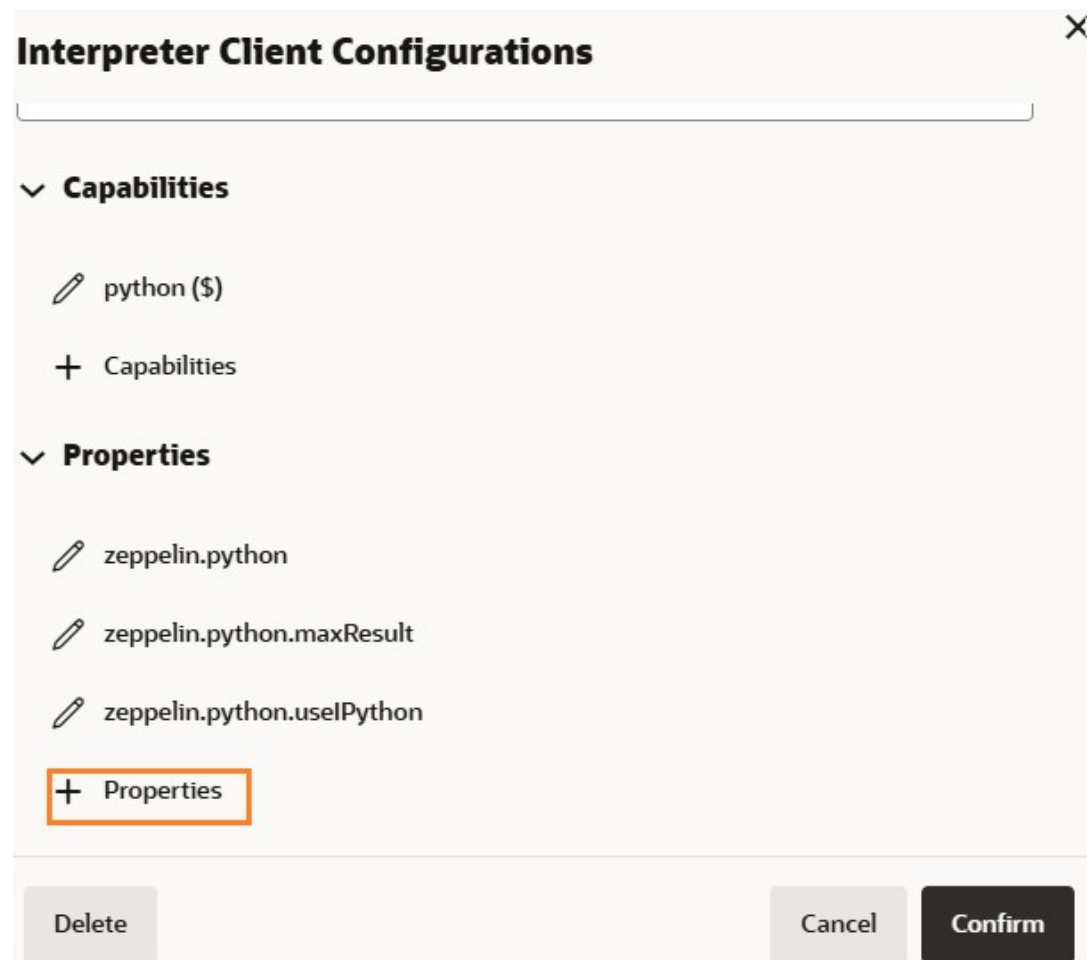
4. On the Right-Hand Side, navigate to **Interpreter Client Configurations**.

Figure 9-9 Interpreter Client Configurations



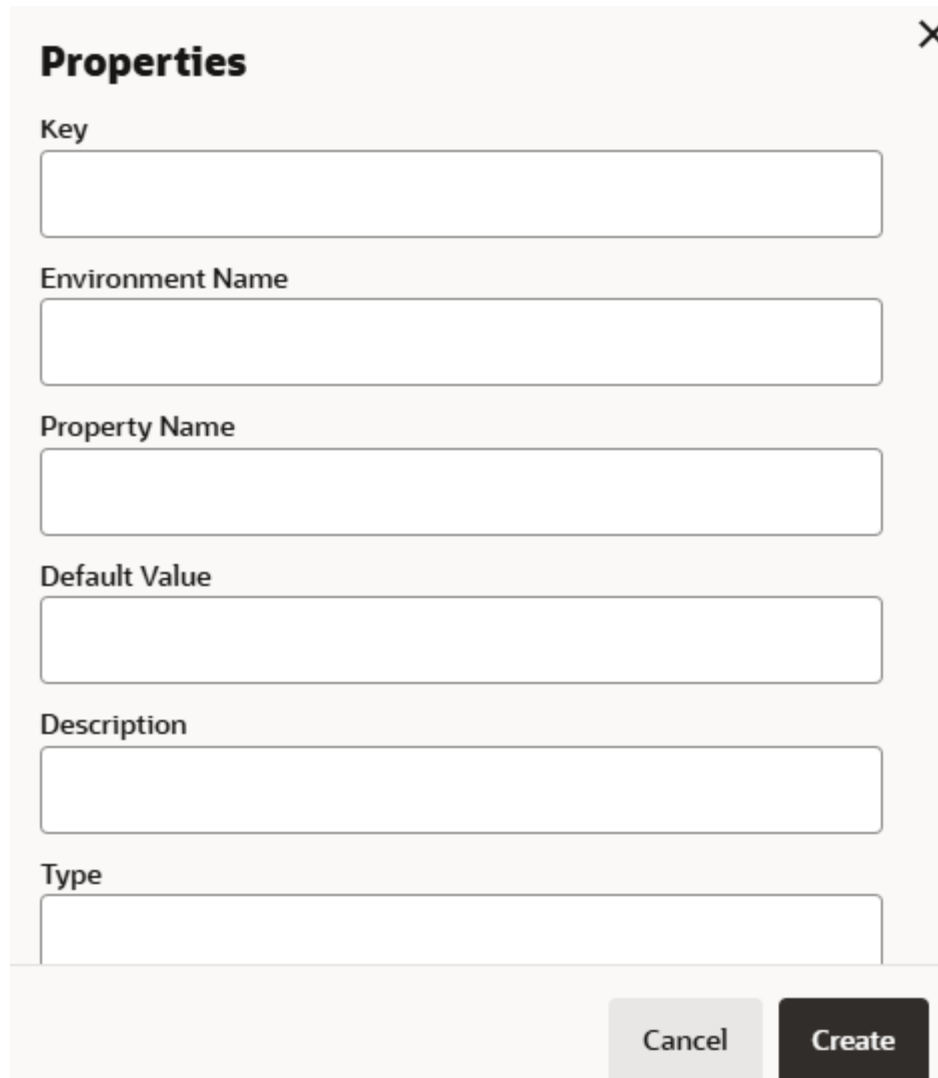
5. Click **oracle.datastudio.python.DsPythonInterpreter (zeppelin)**. The **Interpreter Client Configurations** pane is displayed.

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



Properties [X]

Key

Environment Name

Property Name

Default Value

Description

Type

Cancel Create

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

Interpreter Client Configurations

- zeppelin.python
- zeppelin.python.maxResult
- zeppelin.python.usePython
- zeppelin.interpreter.output.limit**

+ Properties

Initial Code ?

```
1 import sys
2 sys.stdout.isatty = lambda : False
3 sys.stdout.encoding = None
```

Initial Code Capability ?

Delete Cancel Confirm

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

Figure 9-13 Update Python Interpreter

Interpreters

- fcc-python
- fcc-python-m4aml
- fcc-python-sane
- jdbc
- md
- ore
- pgx
- python**
- spark
- Test

Group Name ? python

Group Settings

Initial Code ? 1

Initial Code Capability ?

Credential Configurations ?

+ Credential Configurations

Interpreter Client Configurations

- oracle.datastudio.python.DsPythonInterpreter (zeppelin)
- org.apache.zeppelin.shell.ShellInterpreter (zeppelin)
- oracle.datastudio.python.DsPythonCondainterpreter (zeppelin)

+ Interpreter Client Configurations

Delete Share Close **Update**

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

12. 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:

1. Navigate to the `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/ builtin/interpreters` directory.
2. Open the **python.json** file and navigate to `className`: `"oracle.datastudio.python.DsPythonInterpreter"`.
3. 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"
}
```

4. Save and close the **python.json** file.
5. 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

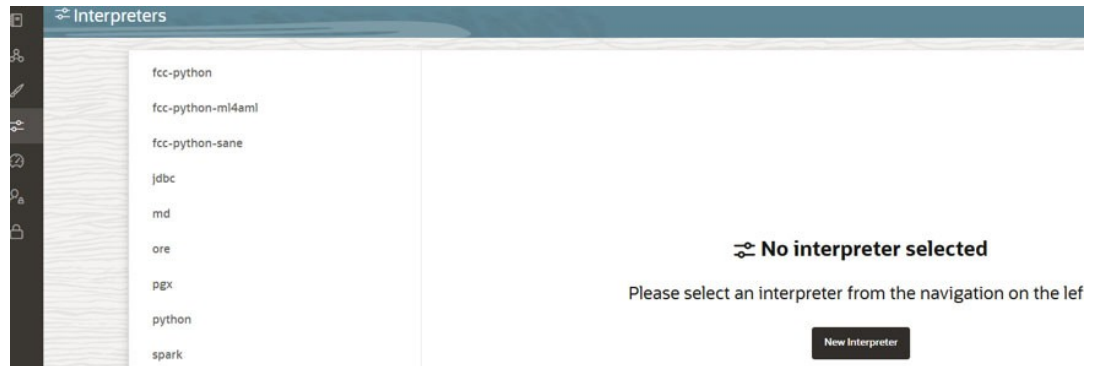
① Note

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:

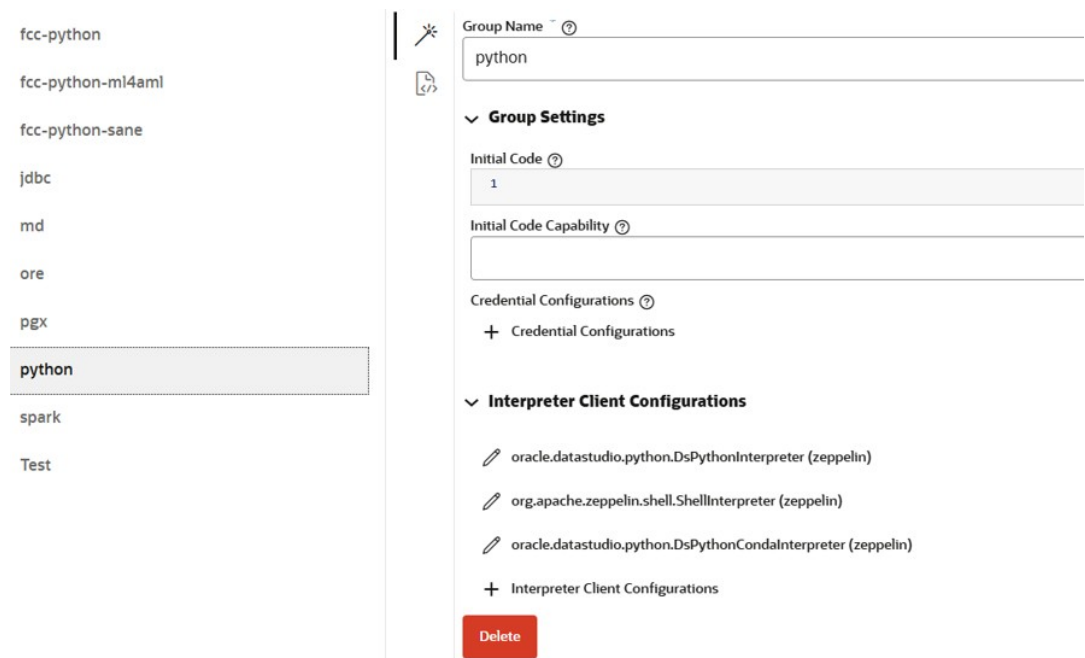
1. Log in to OFS MMG UI.
2. Click **User Profile**, select **Data Studio Options** and then click **Interpreters**. The **Interpreters** page is displayed.

Figure 9-14 Interpreters



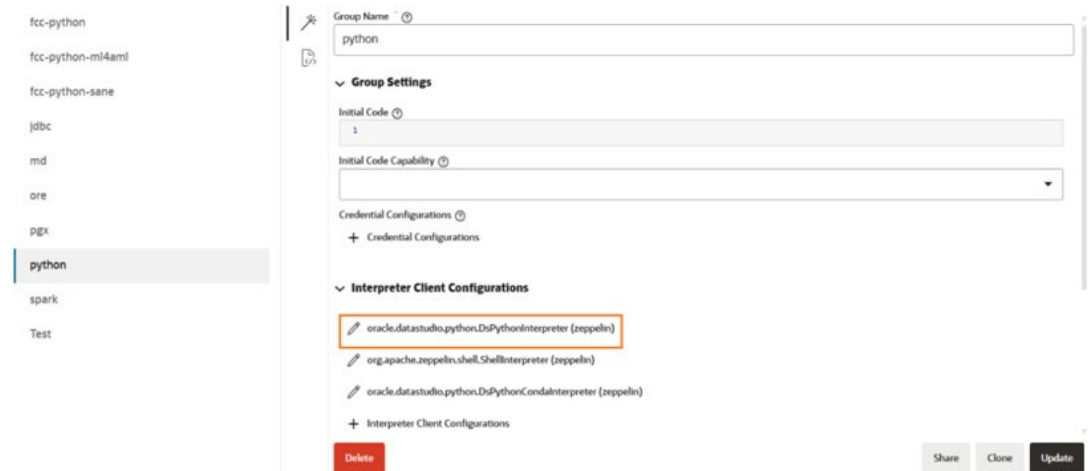
3. On the Left-Hand Side, click **python** interpreter. By default, **Wizard** pane is displayed.

Figure 9-15 Python Interpreter



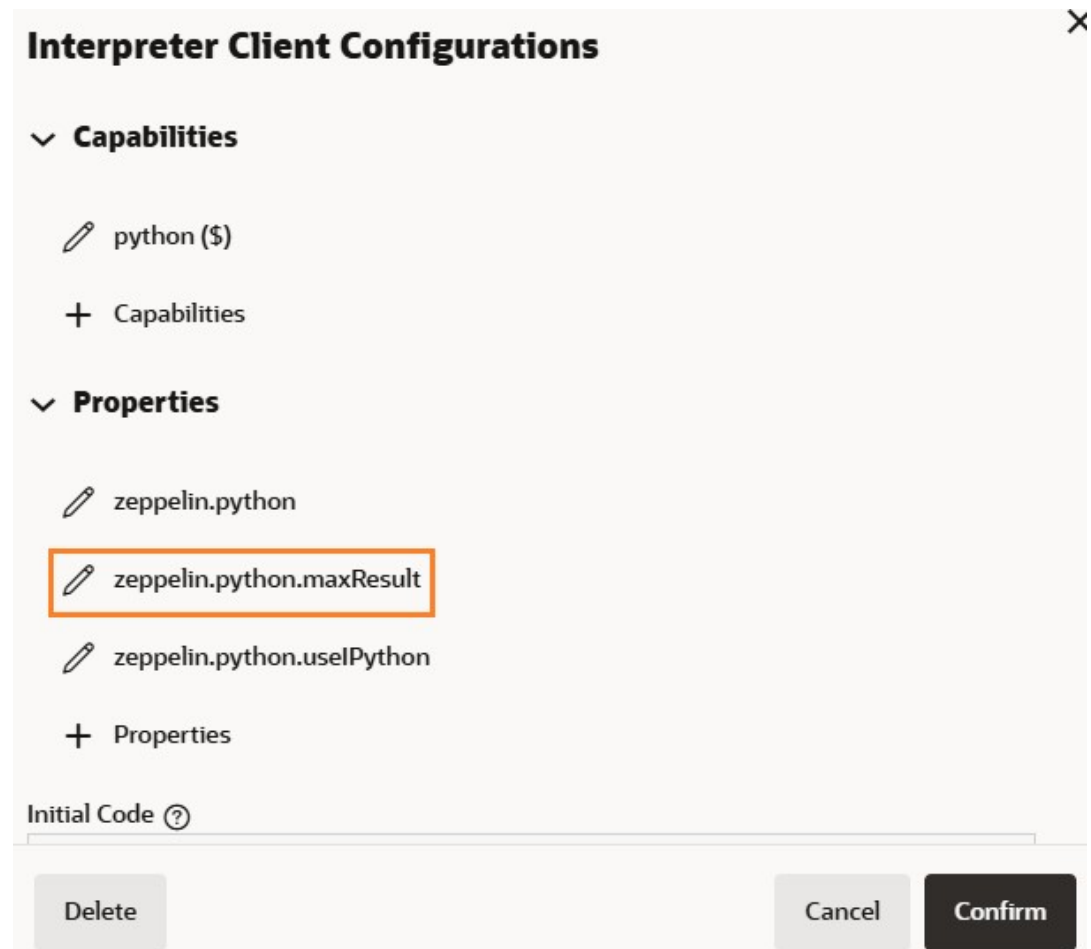
4. On the Right-Hand Side, navigate to **Interpreter Client Configurations**.

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 [X]

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.

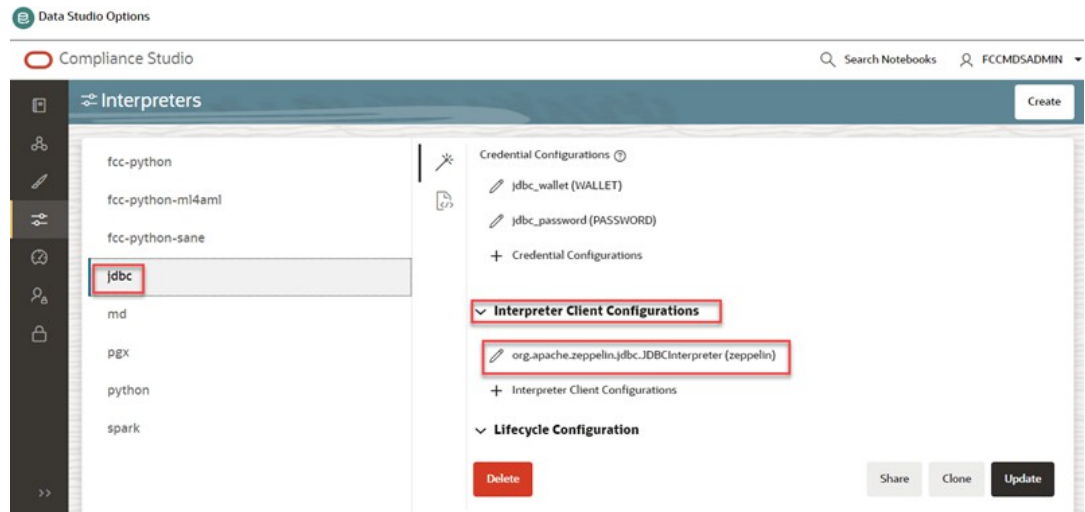
Recommendation

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:

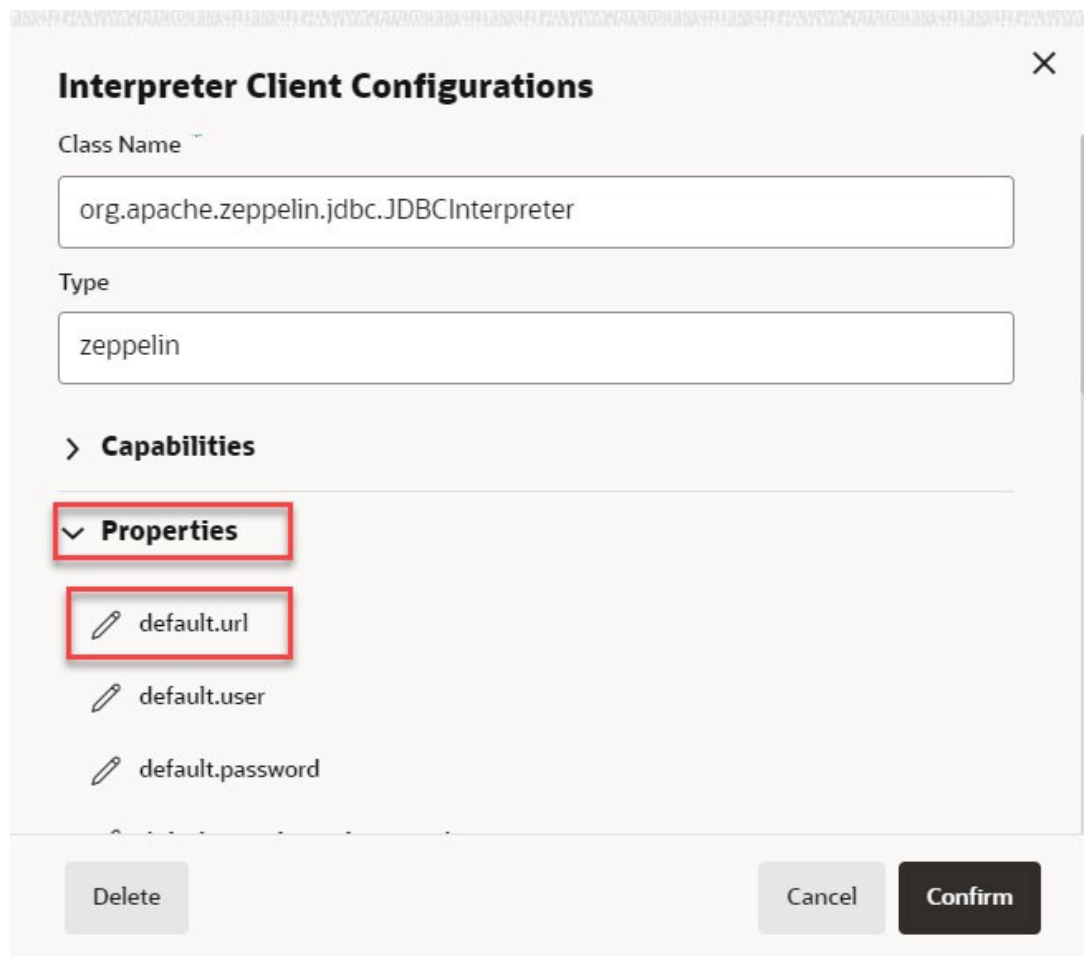
1. Navigate to <MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ bin directory.
2. Open startup.sh file, navigate to line 29 and update jdbc value as **7011**.
For example: `./"$DIR"/datastudio --port 7008 --markdown 7009 --spark 7014 --python 7012 - -jdbc 7011 --shell -1 --pgx 7022 --external`
3. 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



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

Figure 9-21 Properties

Properties [X]

Key
default.url

Environment Name

Property Name
default.url

Default Value
jdbc:oracle:thin:##DB_ALIAS_NAME##

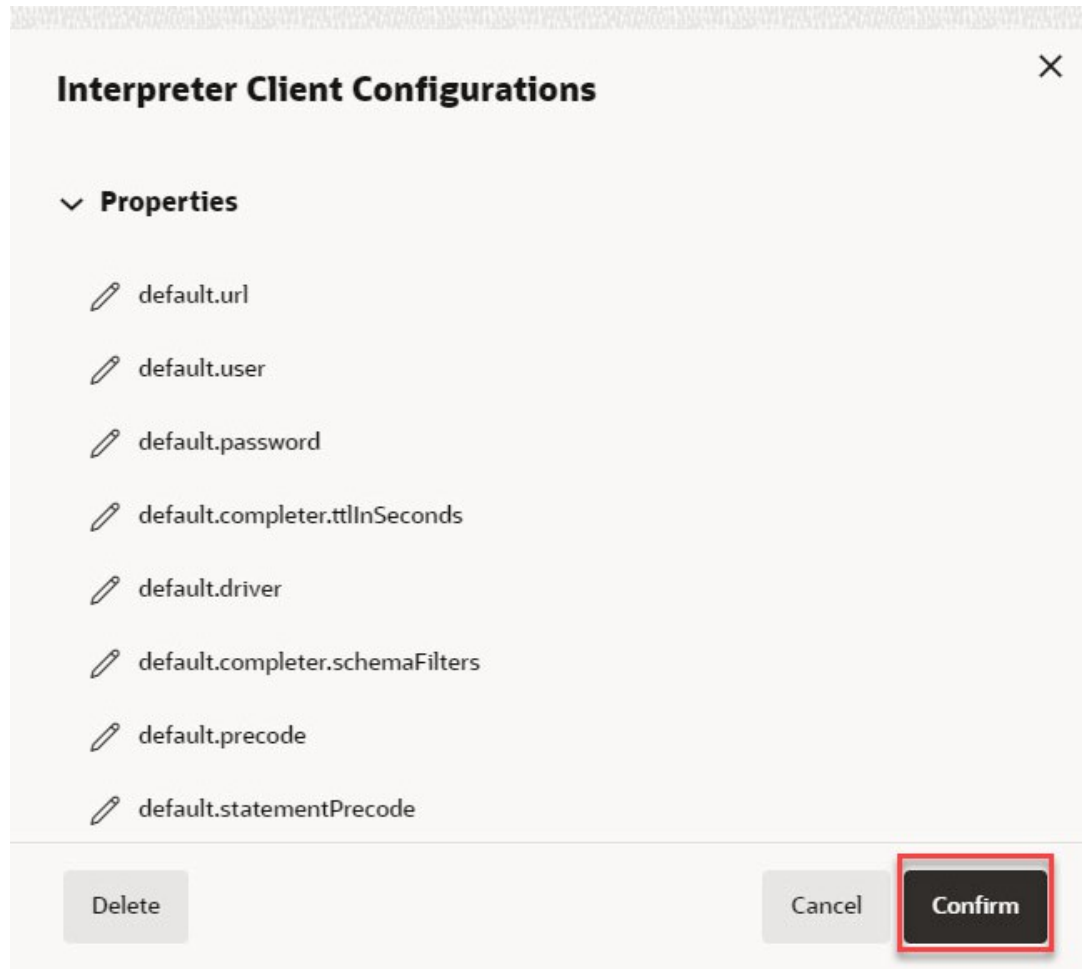
Description
The URL for JDBC.

Type

Delete Cancel Confirm

13. 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##
14. 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
graphviz.formatter.class	Enter the class which implements the formatting of the visualization output. For example, <code>oracle.datastudio.graphviz.formatter.DataStudioFormatter</code>
graphviz.driver.class	Enter the class which implements the PGQL driver. For example: <code>oracle.pgx.graphviz.driver.PgxDriver</code>
base_url	Enter the base URL of the PGX. For example, <code>http://<HOSTNAME>:7007</code>
zeppelin.interpreter.output.limit	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 <MMG_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:

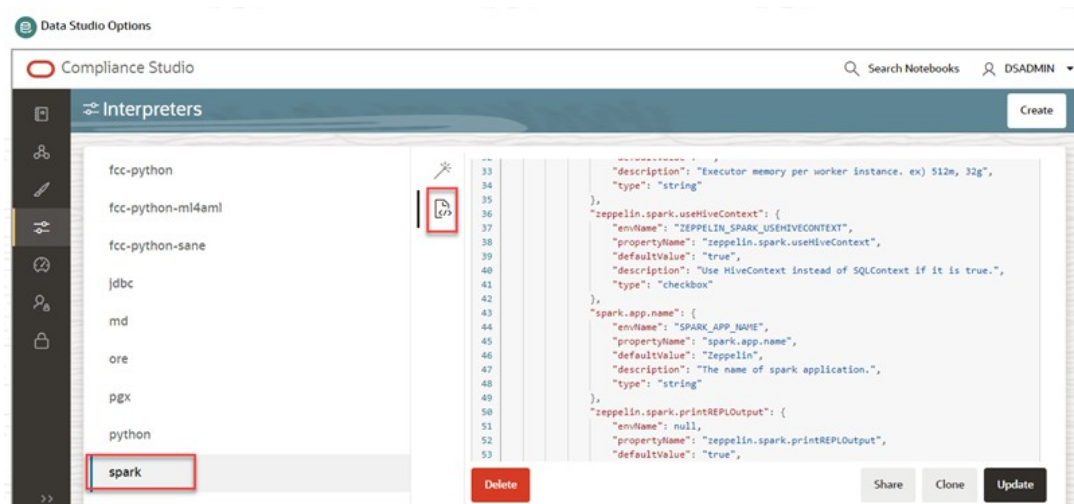
1. Navigate to `<MMG_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/ server/ builtin/interpreters/spark.json` directory.
2. Update the value in `spark.pyspark.python` property of the `spark.json` file.

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
zeppelin.pyspark.python	Enter the Python binary executable for PySpark in both drivers and workers. The default value is python. For example, python
zeppelin.pyspark.useIPython	Set to 'True' to use IPython, else set to 'False'.
zeppelin.interpreter.output.limit	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) [fccstudio@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- 0 https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
 - b. `yum install epel-release-latest-7.noarch.rpm`

- c. `curl- 0 https://cdn.rstudio.com/r/centos-7/pkg/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.

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

Create a Credential

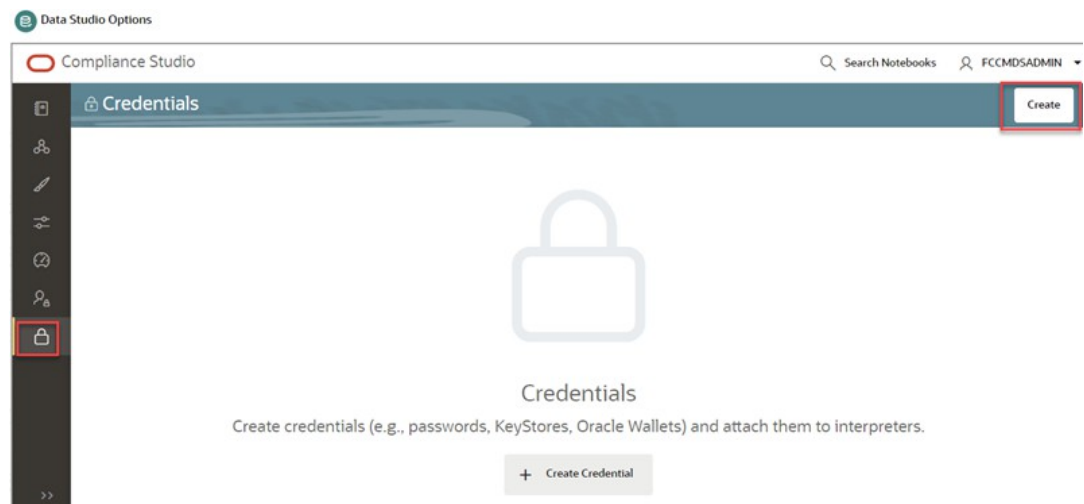
New credentials are created when database details are changed or updated. For example, change in Transparent Network Substrate (TNS) due to hostname change or compulsory periodic update of schema passwords.

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:

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

Figure 9-27 Credentials Page



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

Figure 9-28 New Credential for Password

New credential X

Name

Type
Password ▼

Password

Accessible via APIs in Paragraphs

☐

You can use it in python by writing: ds =
PyDataStudioContext() ds.get_credential("CredentialAlias")
CredentialAlias is the alias you give to the credential in the
notebook.

Cancel Create

- 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.

- 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

New credential [X]

Name

Type
Oracle Wallet ▼

File

Drag and Drop
Select a file or drop one here.

Cancel 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	Upload the wallet zip file that includes the following files: • tnsnames.ora • ewallet.p12 • cwallet.sso These files are available in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/wallet directory. Note: • 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.

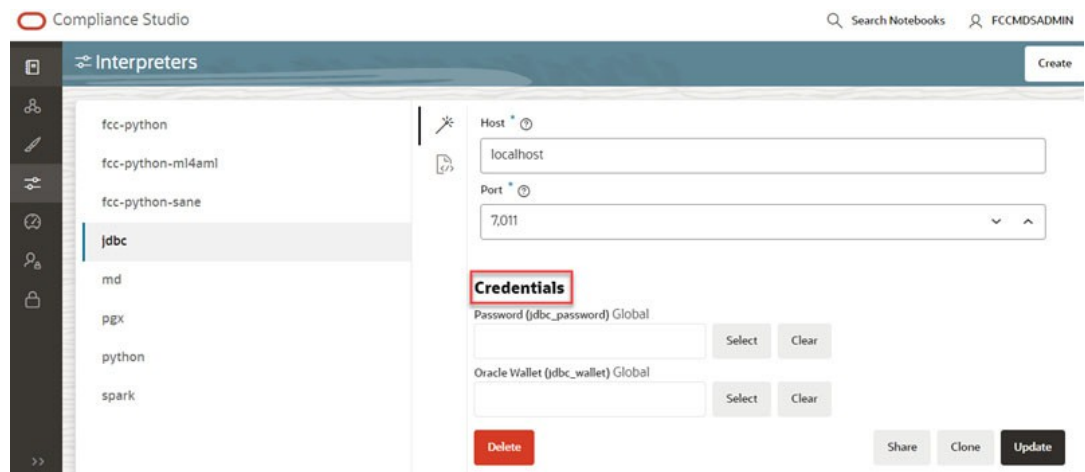
Note

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:

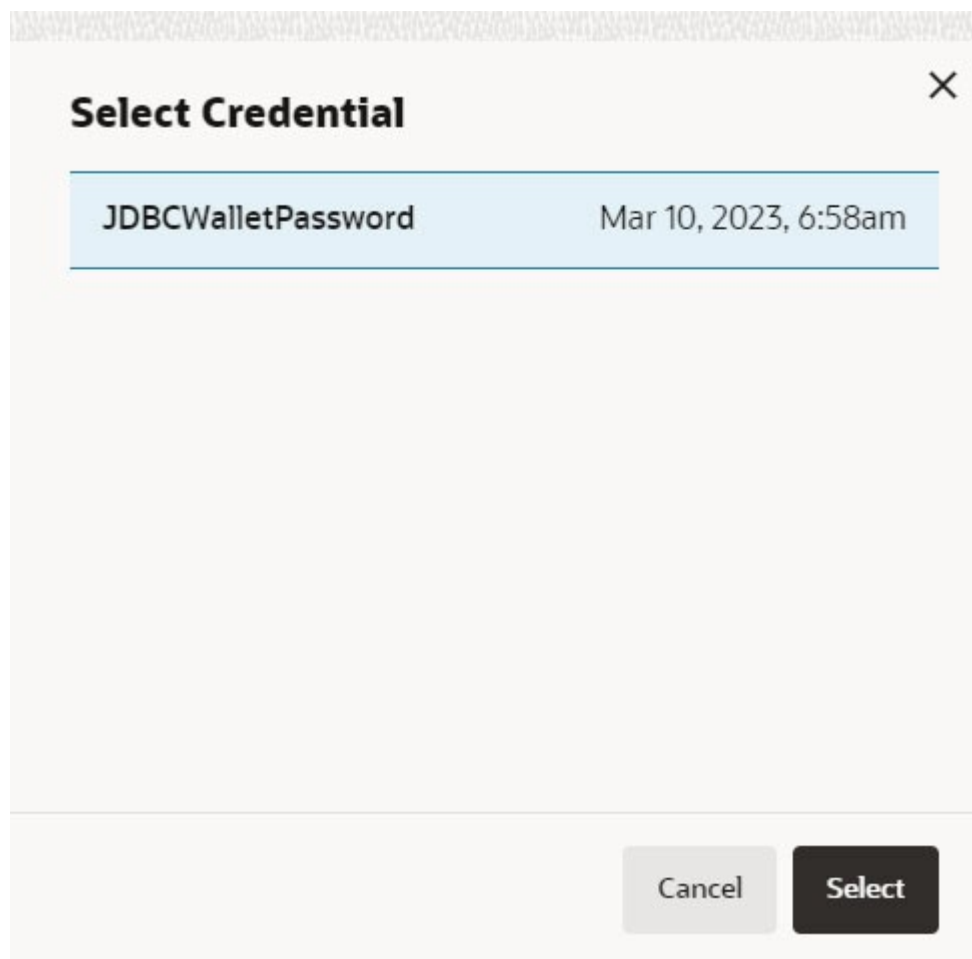
1. On the Interpreters page, select the required interpreters. For example, jdbc.
2. Navigate to the Credentials section.

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.

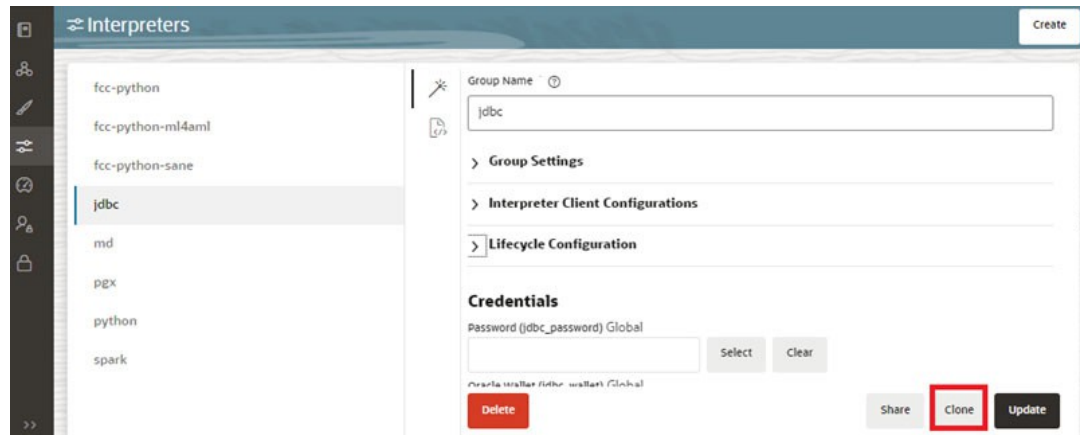
Multiple groups for an interpreter are created to connect different versions of interpreters (Python version: 3, Python version: 2) and connect a different set of users and database schema. For example, Compliance Studio schema, BD schema, etc.

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 you to efficiently manage data stores within a workspace. It provides key features for importing both data models and data from the 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 you can view the structure of the database object.

The previous data management group **DMADMIN** has been removed and now individual groups; **MMGDATAPOPLNADMIN** for data population and **MMGDATAMODELADMIN** for the data model has been added. Each of these groups has individual roles and functions for edit, view, link, delete etc.

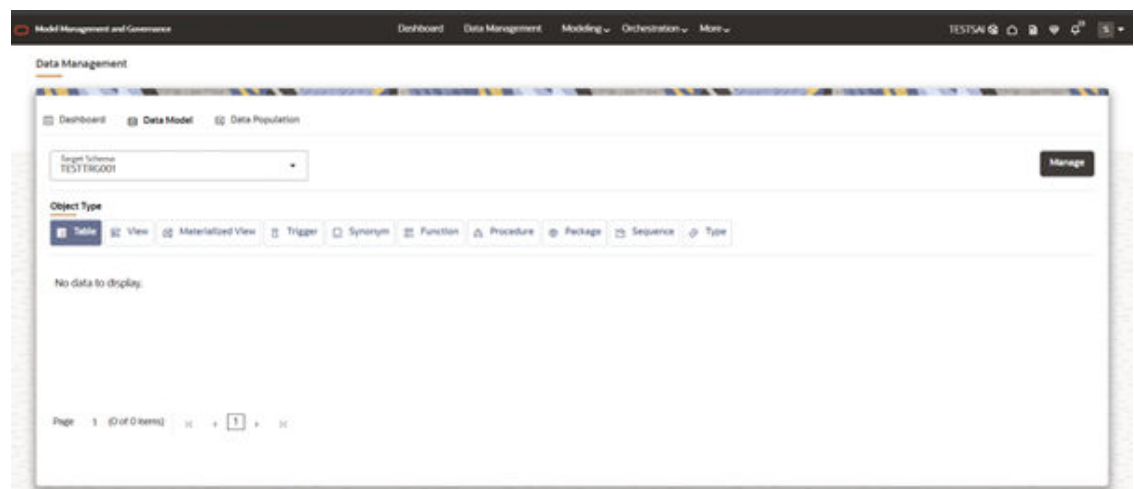
Note

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).

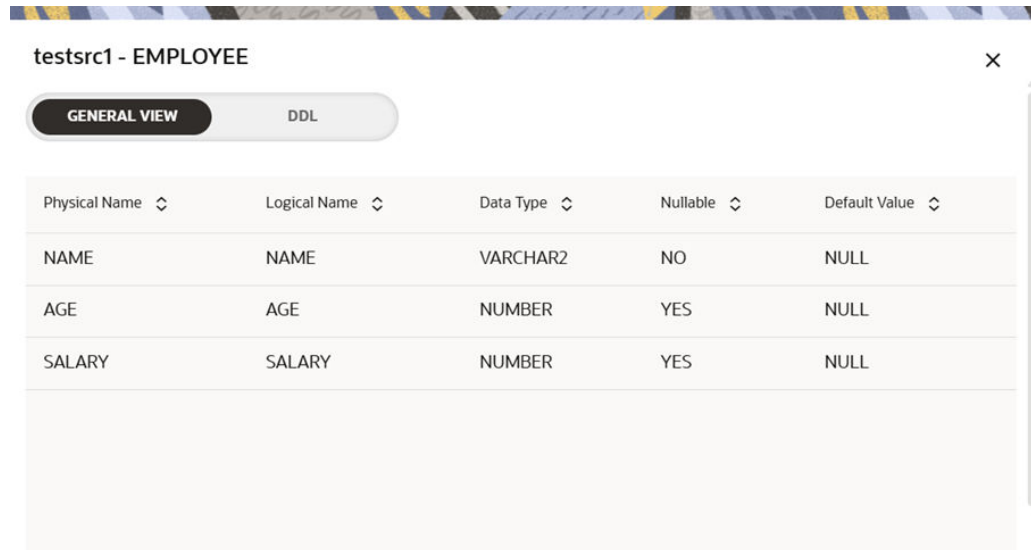
Note

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**

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

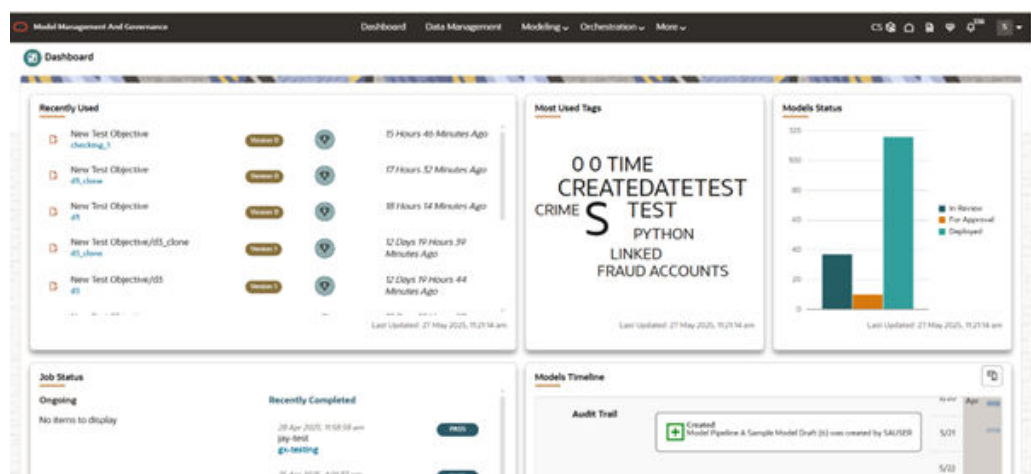
— **Figure 10-3 Generic View**



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

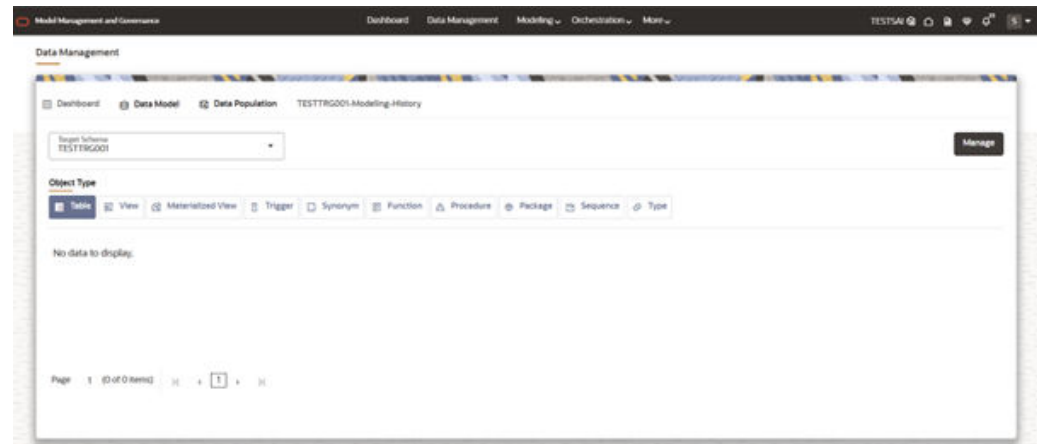
- **Tree View and Add Item buttons**
 - **Tree View:**
 1. Clicking this option opens a new tab within the Data Management UI, titled [TargetSchemaName] - Modeling.
 2. This view provides a hierarchical structure to assist users in creating and organizing new data models.
 - **Add Item:**
 1. This button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



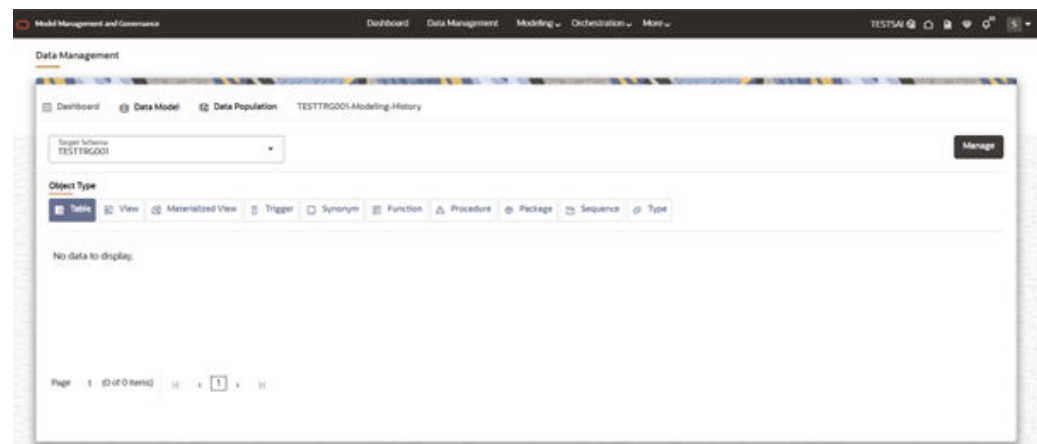
2. Click **Data Model** sub-tab.

Figure 10-5 Data Model sub-tab



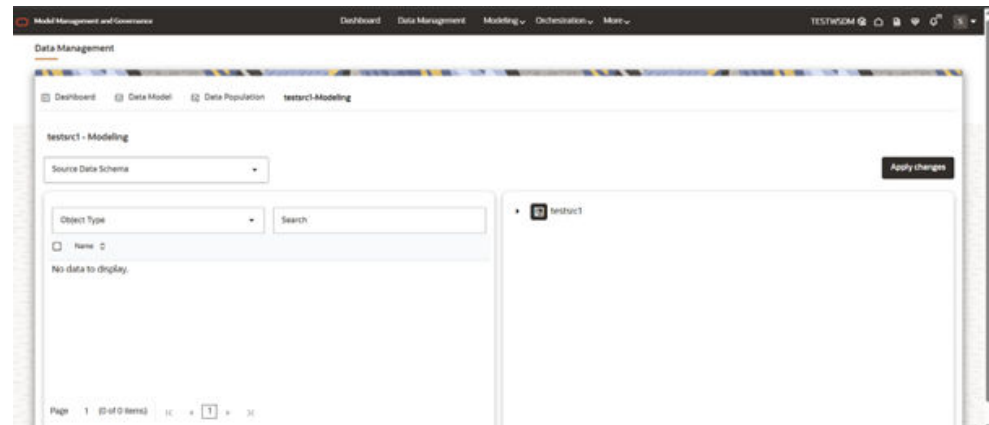
3. This tab contains the **Table Schema** drop-down list, **Object Types** and the list of schemas.
4. Click **Manage** to open a new tab within the MMG- Data Management UI.

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

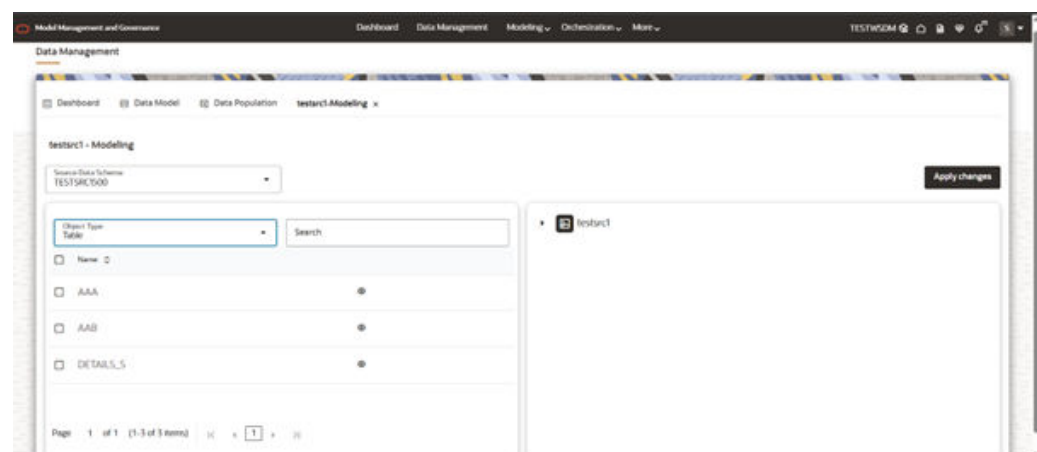


6. Now the user can choose source schema from where user wants to bring in the database objects.
7. After selecting the Source Schema, the user can choose the desired database object type from the left panel. The table in the left panel will then display all available database objects that match the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

Note

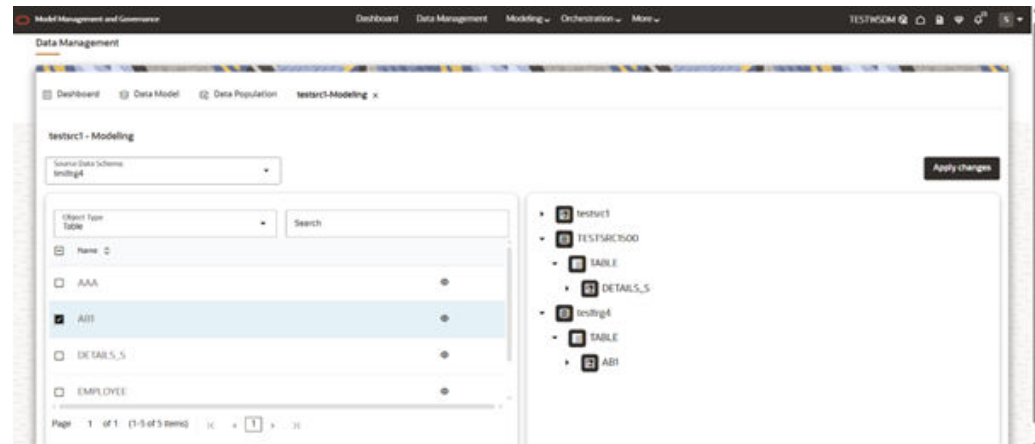
Each database object listed in the table includes an eye icon. Clicking this icon opens a pop-up that provides both the **DDL View** and the **Generic View** of the selected object.

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



9. 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

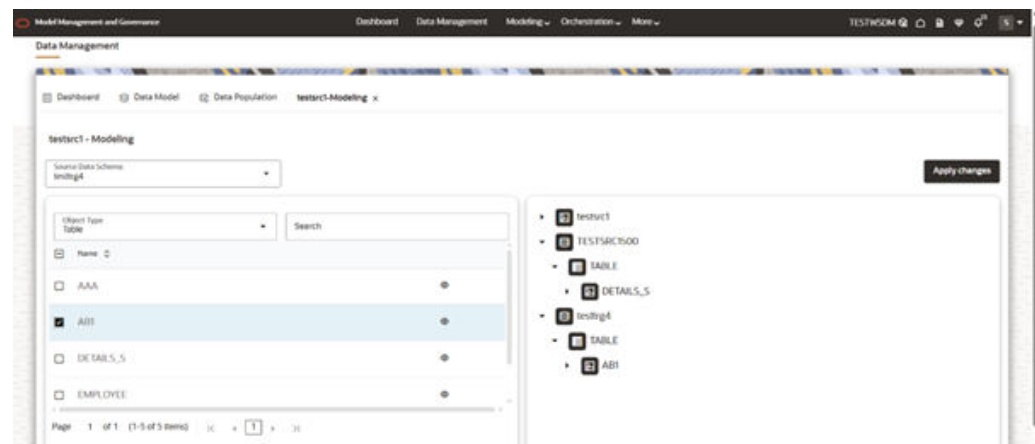
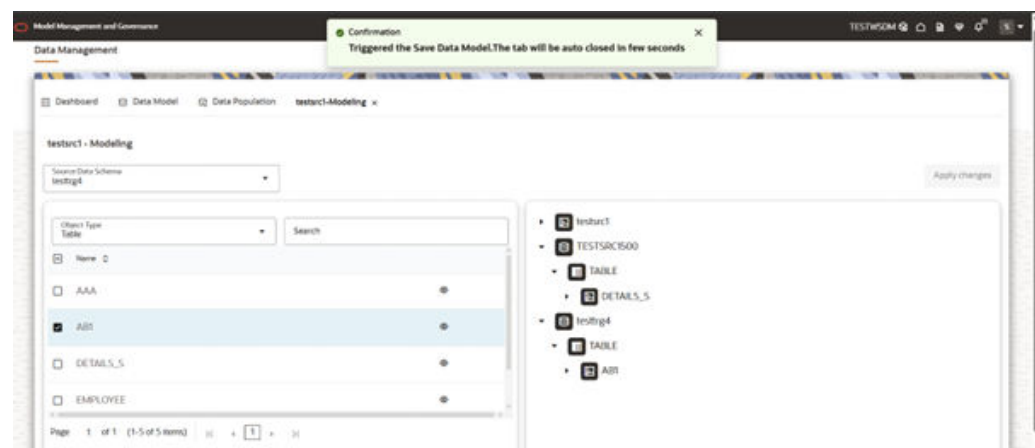


Figure 10-11 Confirmation message



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

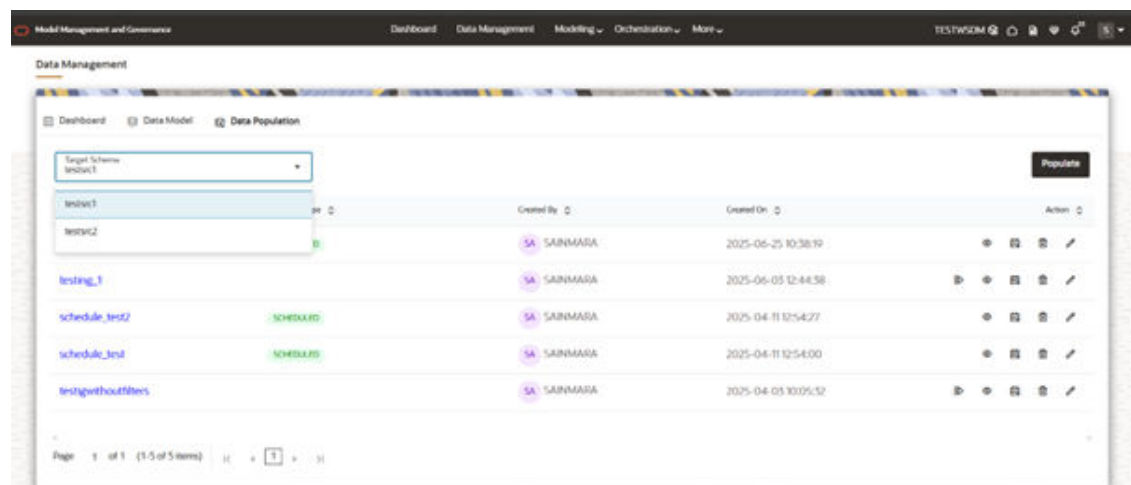
For a selected Target Schema, if any previous Data Modeling is in progress, the user is not allowed to do another Data Modeling.

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.

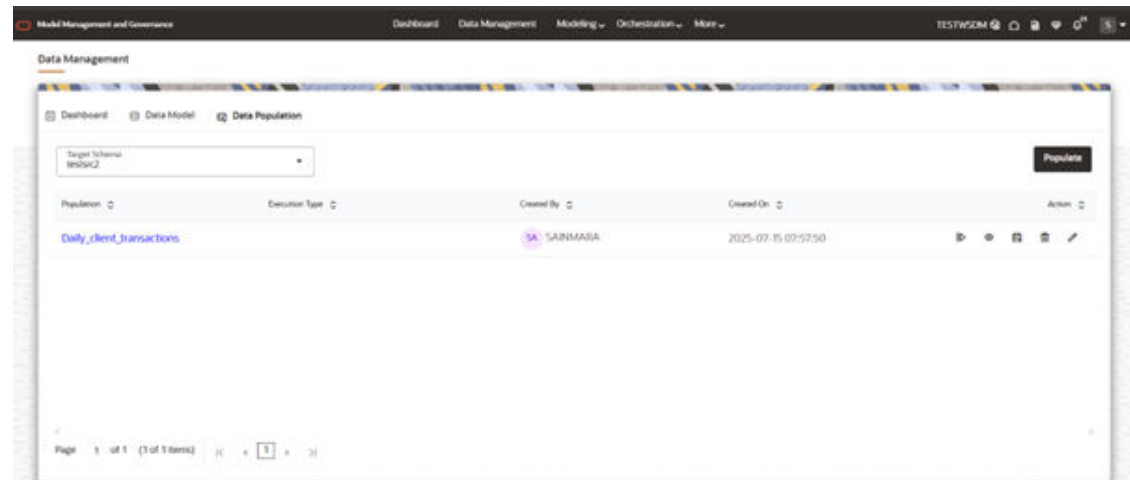
Populate button

- 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).

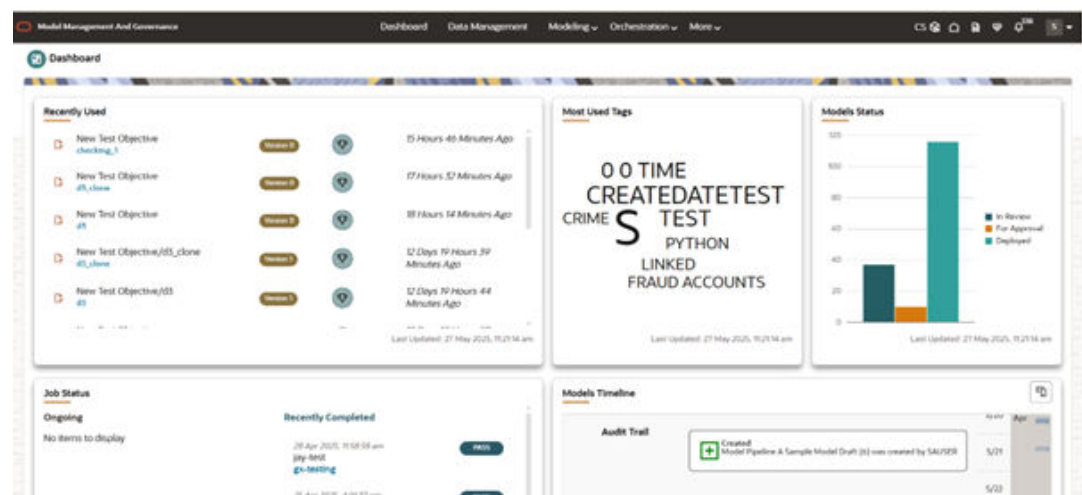
Figure 10-13 Data Population List



Steps of Data Population

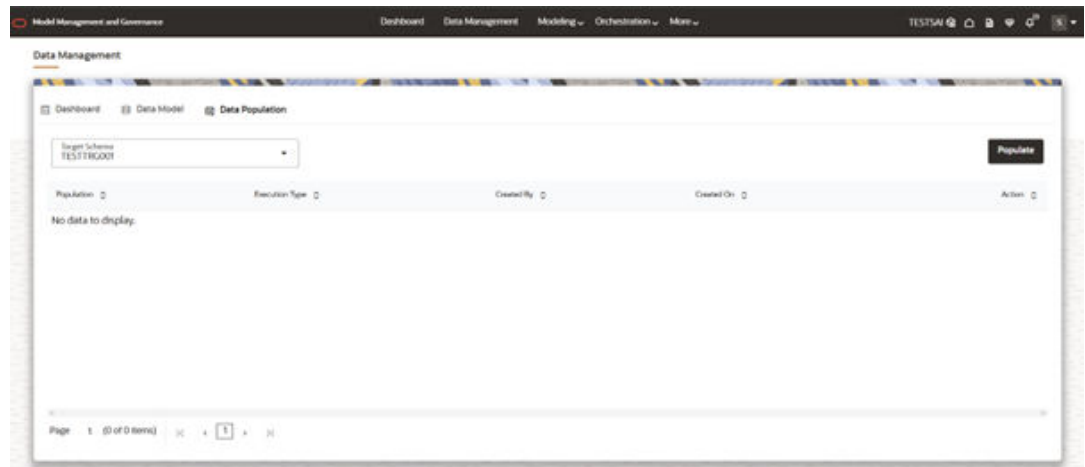
1. Launch any workspace where you wanted to do Data Population and go to Data Management tab from main menu.

Figure 10-14 Data Population Dashboard



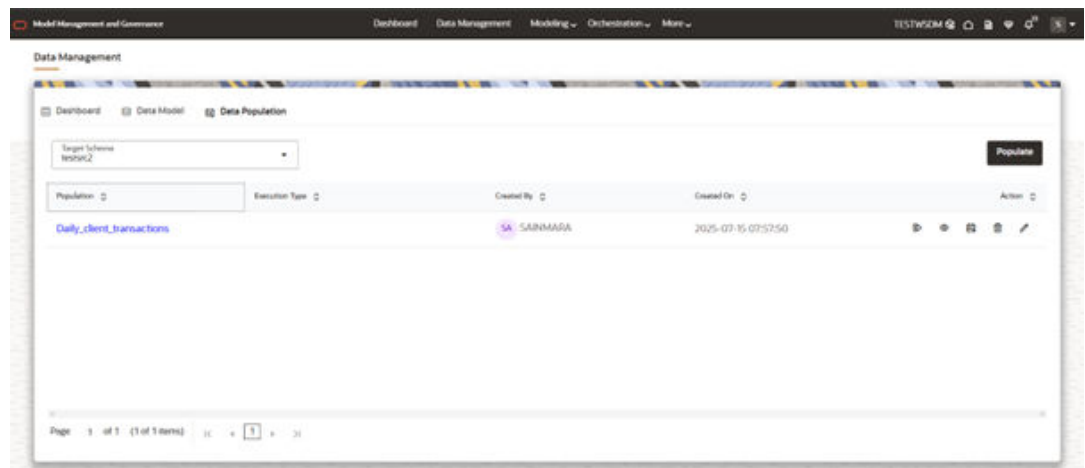
2. Click on Data Population sub-tab.

Figure 10-15 Data Population Sub-tab



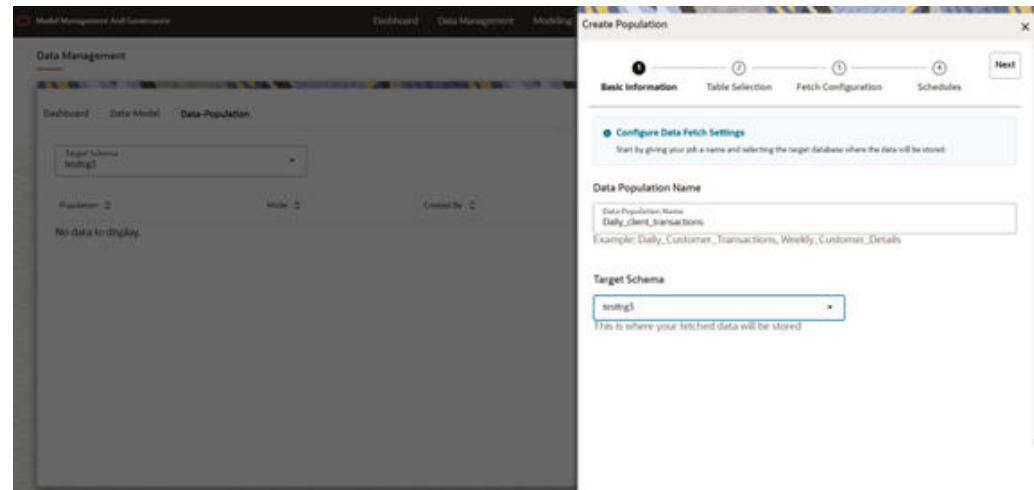
3. Now the user can see the list of Data Population which are created. When we launch for first time, it will be empty.
4. Click on Populate button, which will take you through the step-by-step process to create a new Data Population.

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



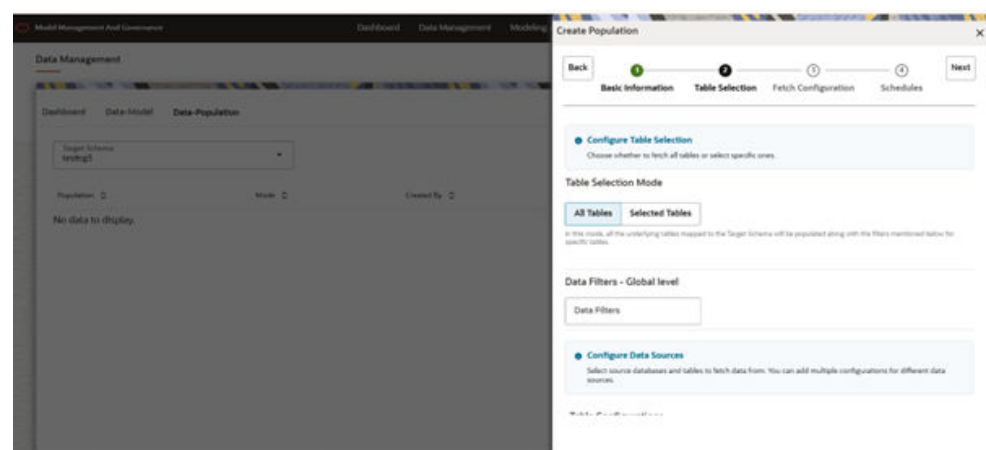
7. Table Selection

- 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.

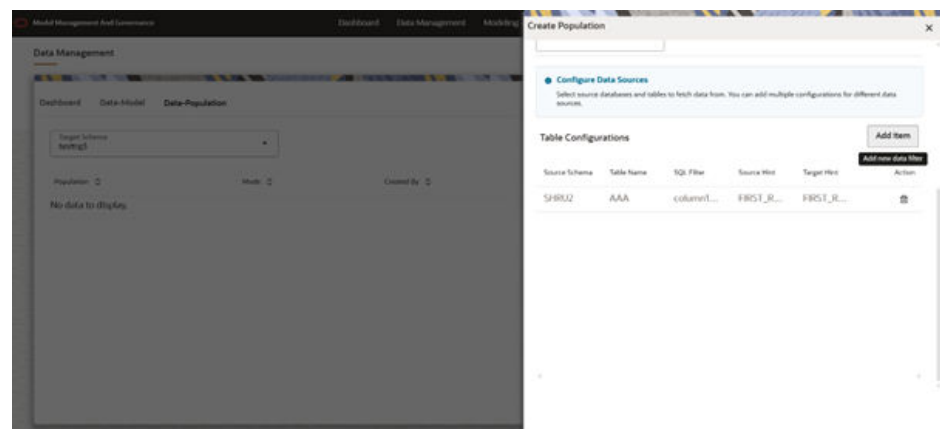
Figure 10-18 Table selection and global filter



- **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.

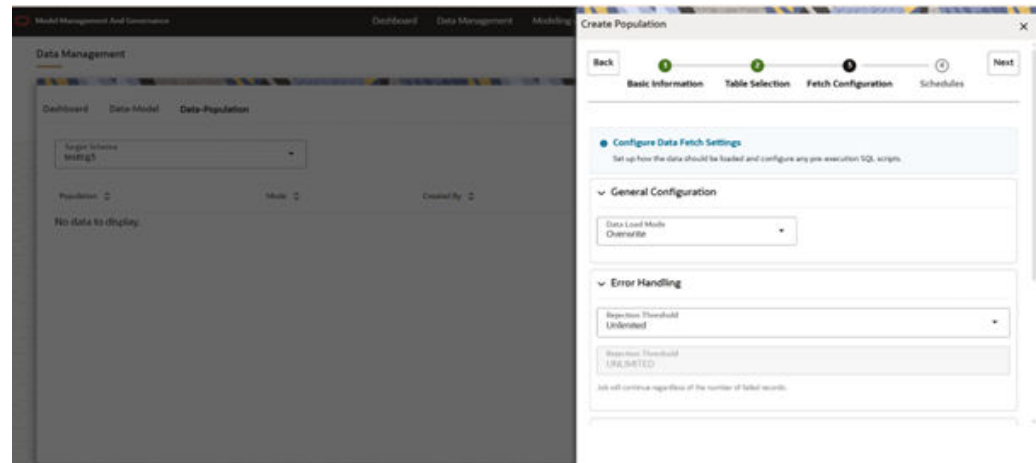
Figure 10-19 Table Level Filters



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.

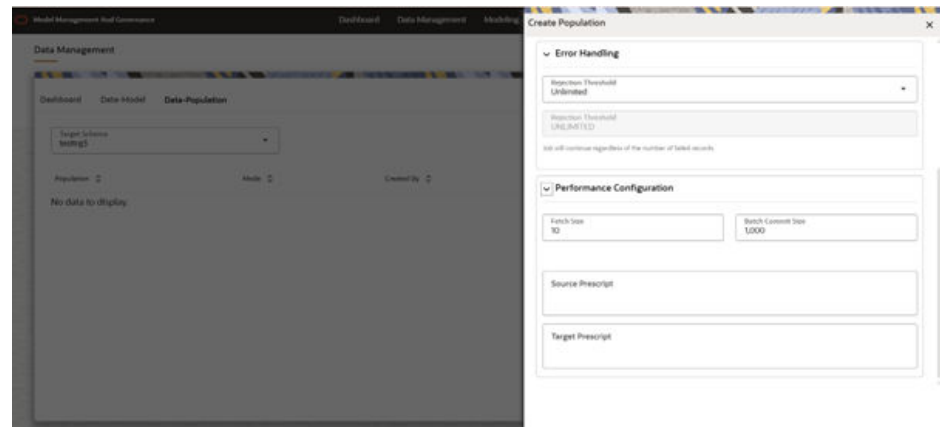
b. Error Handling

- 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.

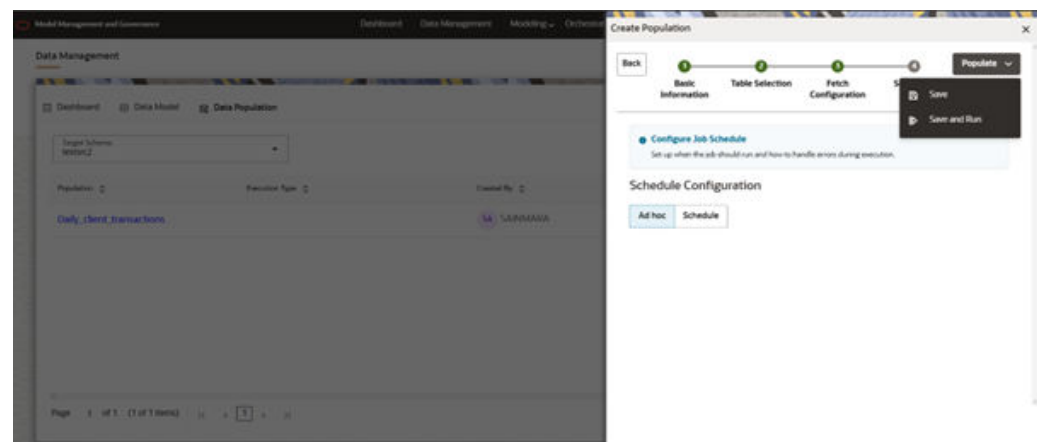
Figure 10-21 Performance Configuration



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

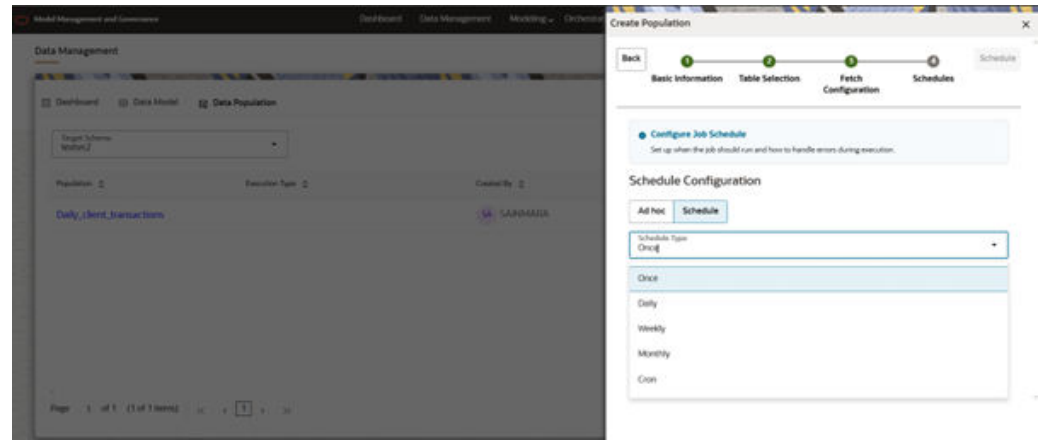


Figure 10-24 Schedule Daily options

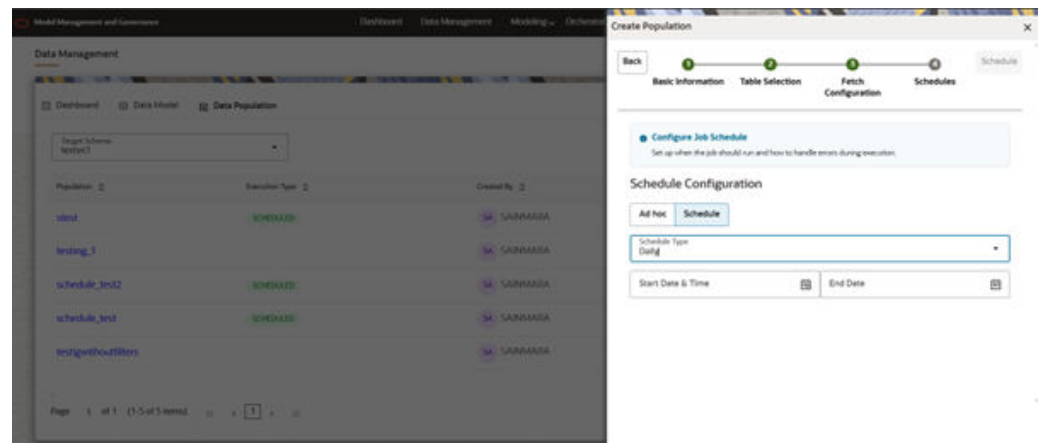


Figure 10-25 Schedule Weekly options

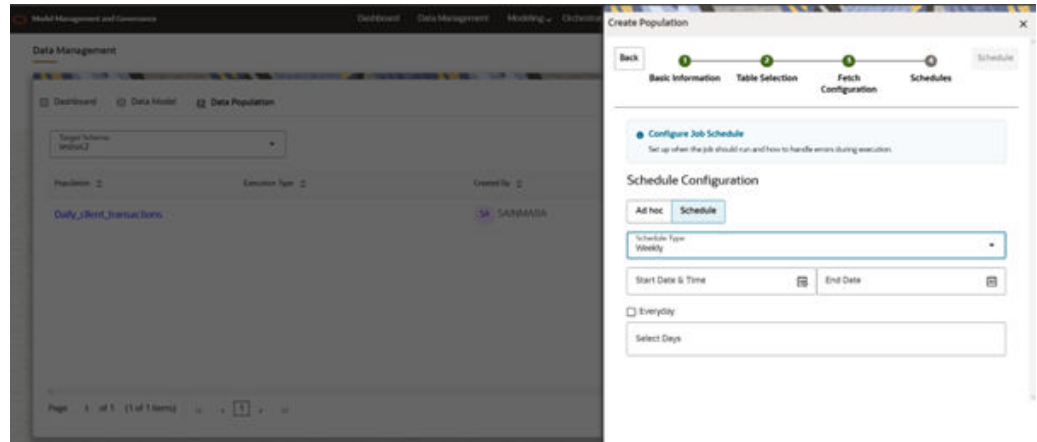


Figure 10-26 Schedule Monthly options

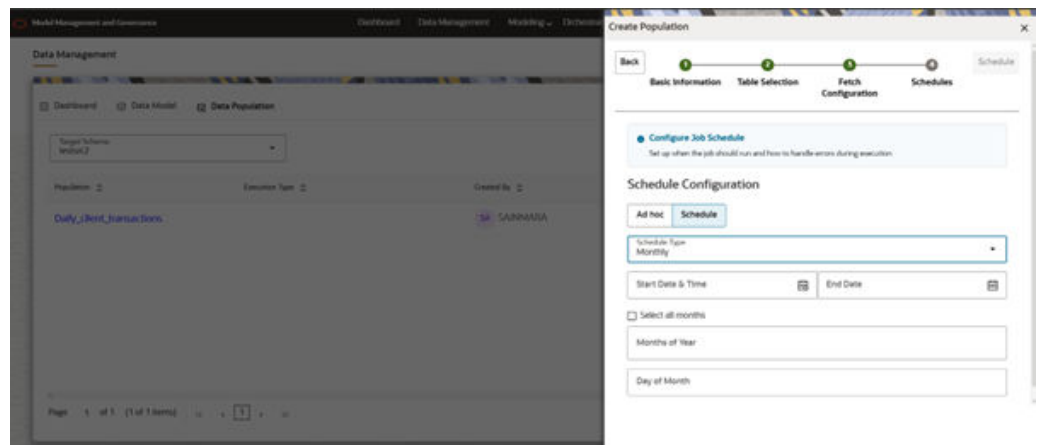


Figure 10-27 Schedule Cron Expression

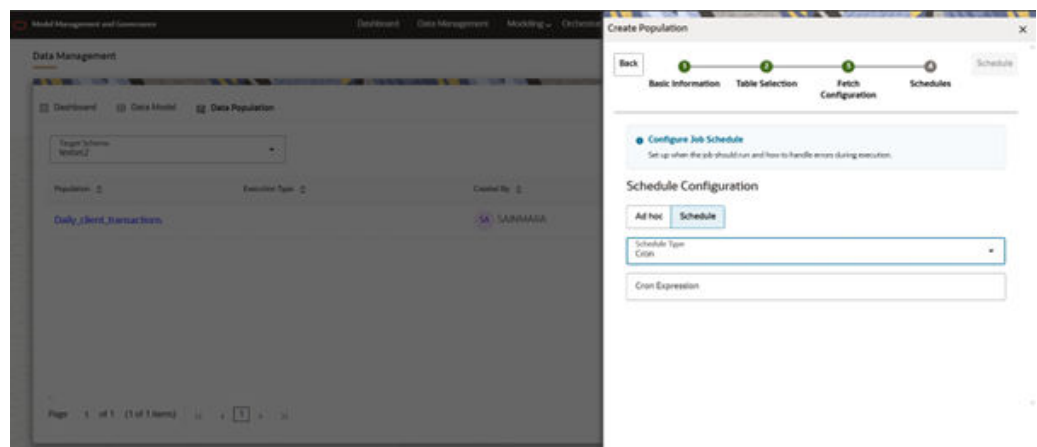
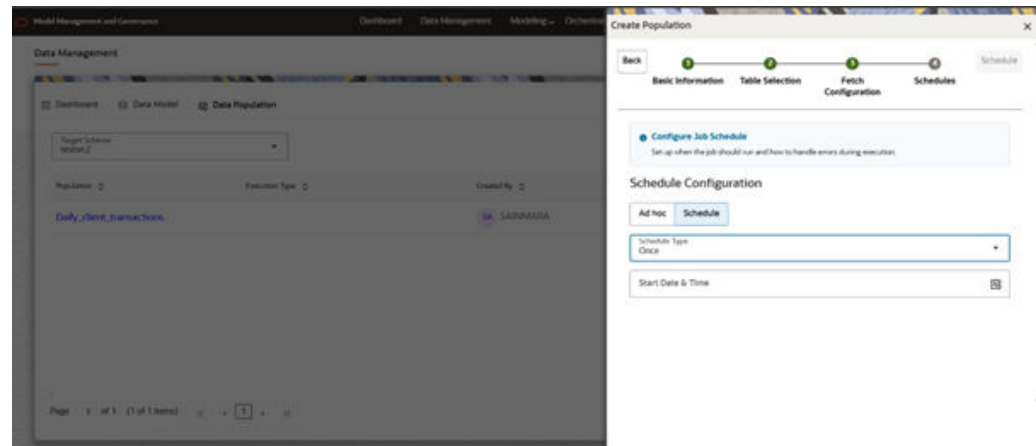


Figure 10-28 Schedule Once options



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.

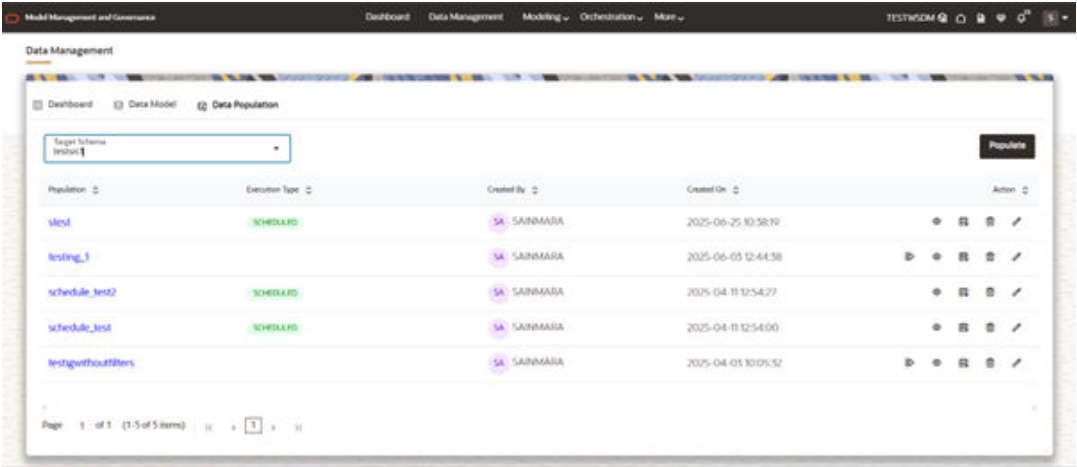
SAVE or SAVEANDRUN Mode Actions

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

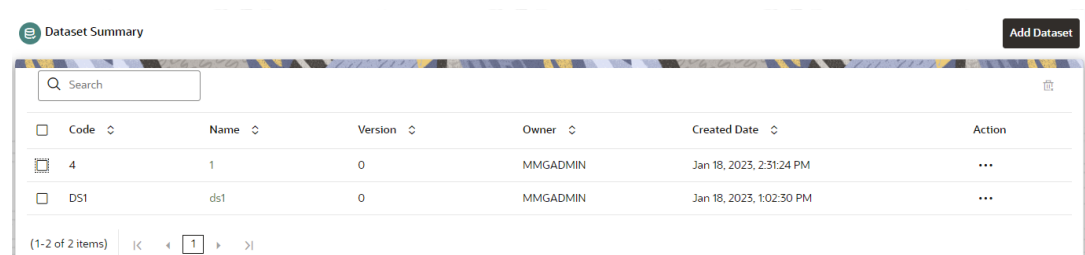
The Dataset Summary page gives access to the various Dataset functions such as create, view, and delete.

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. At the top right is an 'Add Dataset' button. Below it is a search bar. The main content is a table with columns: Code, Name, Version, Owner, Created Date, and Action. There are two rows of data. The first row has Code '4', Name '1', Version '0', Owner 'MMGADMIN', and Created Date 'Jan 18, 2023, 2:31:24 PM'. The second row has Code 'DS1', Name 'ds1', Version '0', Owner 'MMGADMIN', and Created Date 'Jan 18, 2023, 1:02:30 PM'. At the bottom left of the table, it says '(1-2 of 2 items)' and there are navigation arrows.

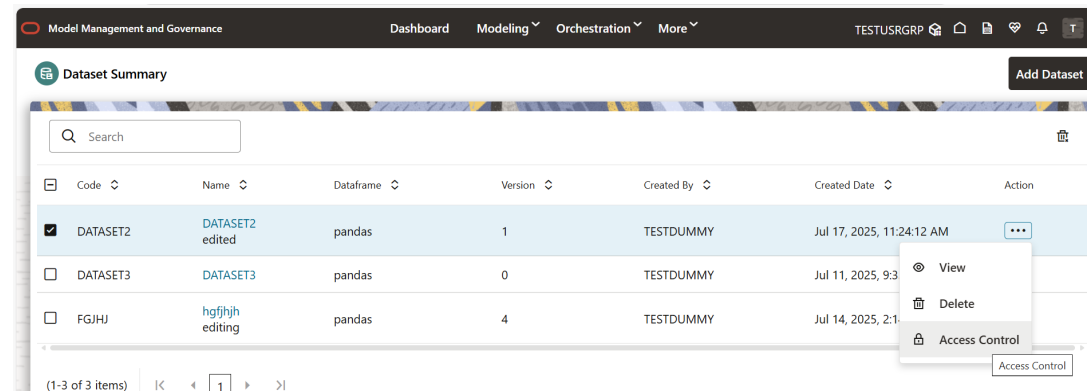
Code	Name	Version	Owner	Created Date	Action
4	1	0	MMGADMIN	Jan 18, 2023, 2:31:24 PM	...
DS1	ds1	0	MMGADMIN	Jan 18, 2023, 1:02:30 PM	...

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

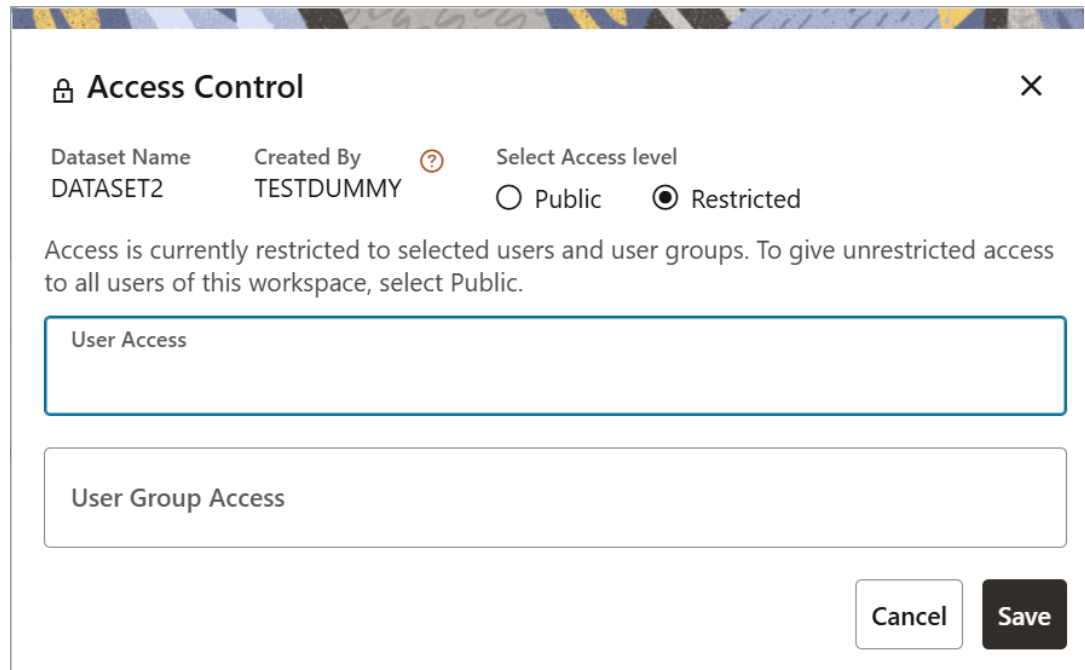
Table 11-1 Fields and icons on the Dataset Summary page

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

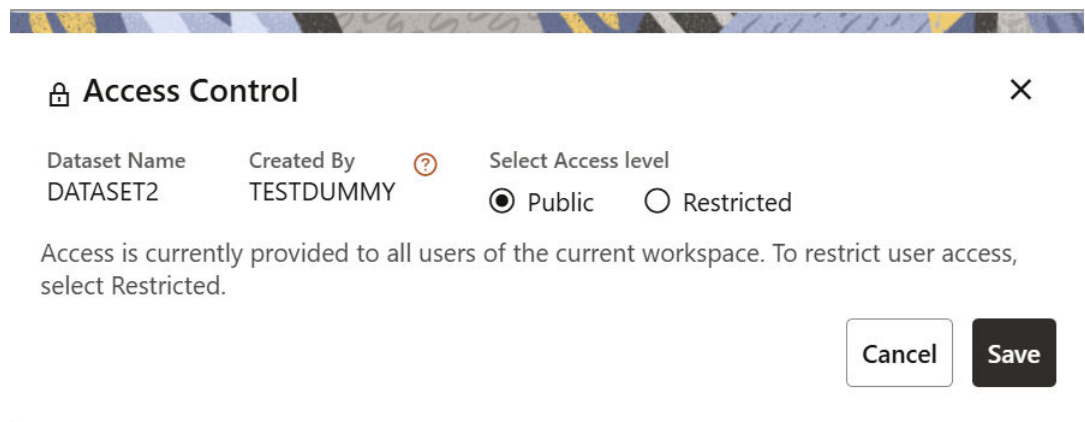


Access Control Pop has two Access Level Radio button: Restricted Mode and Public Mode.

Figure 11-3 Restricted Mode

The dialog box is titled "Access Control" with a close button (X) in the top right corner. It contains the following fields and controls:

- Dataset Name:** DATASET2
- Created By:** TESTDUMMY
- Select Access level:** Two radio buttons are present: "Public" (unselected) and "Restricted" (selected).
- Text:** "Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public."
- User Access:** A text input field with the placeholder text "User Access".
- User Group Access:** A text input field with the placeholder text "User Group Access".
- Buttons:** "Cancel" and "Save" buttons are located at the bottom right.

Figure 11-4 Public Mode

The dialog box is titled "Access Control" with a close button (X) in the top right corner. It contains the following fields and controls:

- Dataset Name:** DATASET2
- Created By:** TESTDUMMY
- Select Access level:** Two radio buttons are present: "Public" (selected) and "Restricted" (unselected).
- Text:** "Access is currently provided to all users of the current workspace. To restrict user access, select Restricted."
- User Access:** A text input field with the placeholder text "User Access".
- User Group Access:** A text input field with the placeholder text "User Group Access".
- Buttons:** "Cancel" and "Save" buttons are located at the bottom right.

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. Inside the dialog, there are three fields: "Dataset Name" with the value "DATASET2", "Created By" with the value "TESTDUMMY" and a help icon (?), 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." Underneath the message is a dropdown menu labeled "User Access". The dropdown menu is open, displaying a list of user groups: MMGUSER, SYSADMN, DUMMY1, DUMMYDELETE, and MMGADMIN.

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

- 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

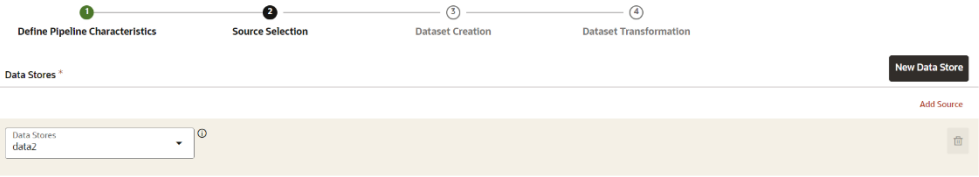
This section allows you to define the source of data. Here, you can choose the data structures from an existing datasources. Click on the icon next to Data Source field to view the tables/ data source information associated with the data source.

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



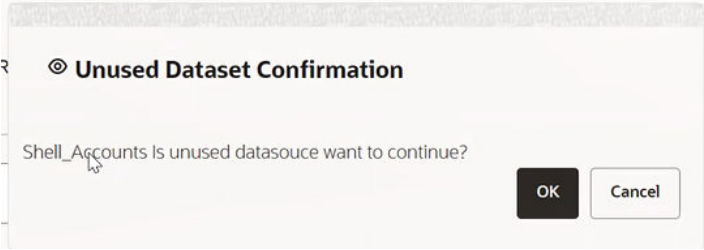
Note

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 **Deleteto** delete added Data Source.

- Click **Next** to go to the next step or click **Previous** to go back to previous step.

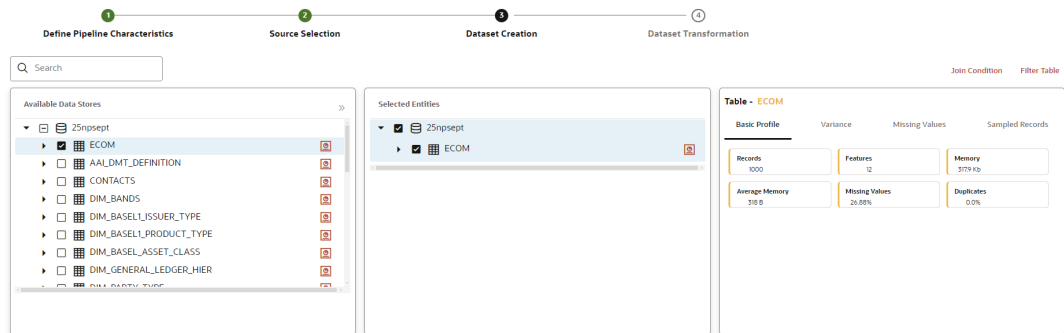
Create a Dataset

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

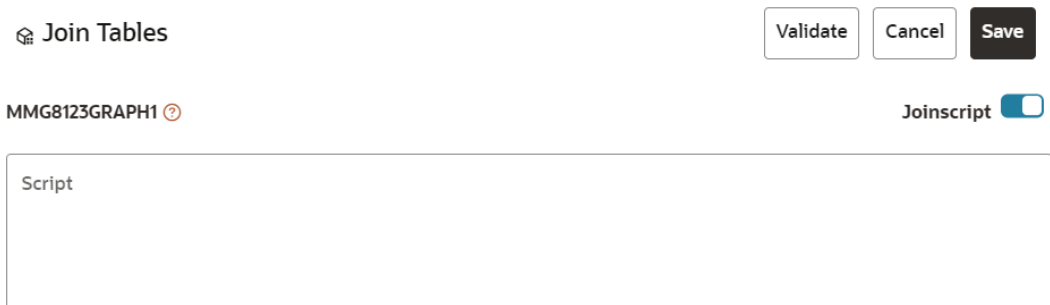
Select the Column name of Table from the drop-down list. You can use multiple join conditions by using icon.

Note

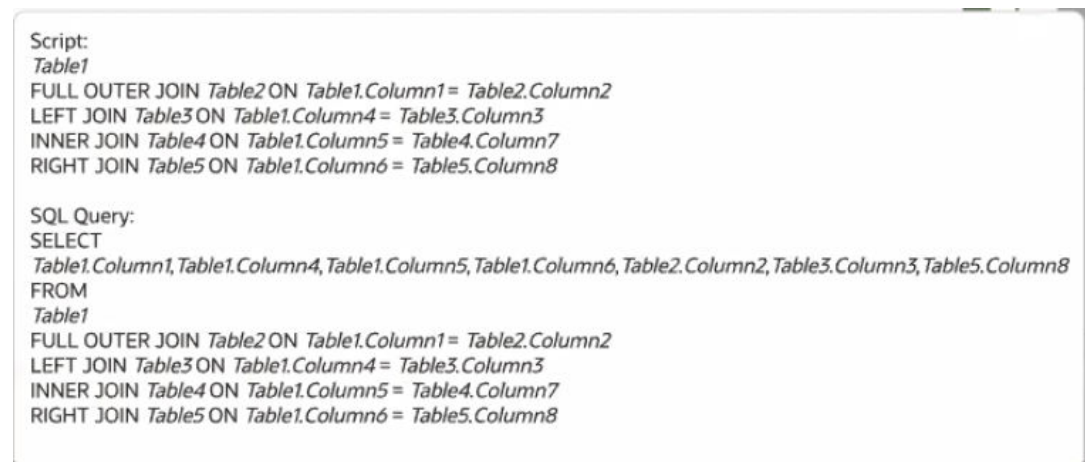
In the current release, only INNER JOIN is present.

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

Basic Profile	Variance	Missing Values	Sampled Records
Profile -DATABASECHANGELOGLOCK_MMG			
Records 1	Features 4	Memory 176 B	
Average Memory 176 B	Missing Values 50.0%	Duplicates 0.0%	

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

Transformation Validate Cancel Add

Type
Feature Transform

Select Transform
Custom Transform

Custom Transform

Data Cleaning

- Duplicates Removal
- Impute Missing Data
- Outlier Removal
- Filter Features

Data Transformation

- Encode Categorical Features
- Encode Datetime Features
- Encode Cyclical Features

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=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	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. 'fmax ' denotes the maximum possible value of the cyclical feature data.
5.	Impute Missing Data	This function imputes missing data within a feature. For numerical features, there are 4 methods for imputing missing data. simple - imputes with mean, median, most_frequent values based on chosen arg value using the SimpleImputer in sklearn. const - fills the missing values with the value given in the arg knn - imputes using the KNNImputer in sklearn with k value given in arg mice - imputes using the IterativeImputer in sklearn For non-numerical data, missing values can be imputed using the 'const' method by replacing all missing values with the value given in arg
6.	Feature Scaling	This function is used to scale multiple selected numerical features using the StandardScaler in sklearn
7.	Dimensionality Reduction	This function performs PCA on selected numerical features to reduce the dimensionality using sklearn.decomposition.PCA module. The number of output features can be specified using dim field. The names of the output features' names can be specified in the fields 'Physical Feature Name' and 'Feature Name'
8.	Outlier removal	This function is used to remove outliers present in a feature based on the specified zscore value. Non-numerical features are label encoded before removing the outliers.
9.	Duplicates Removal - Data Frame	This function removes all duplicate rows in the dataframe.
10.	Duplicates Removal - Feature	This function removes all duplicate rows within a specified subset of features and consequently removes those rows from the data frame

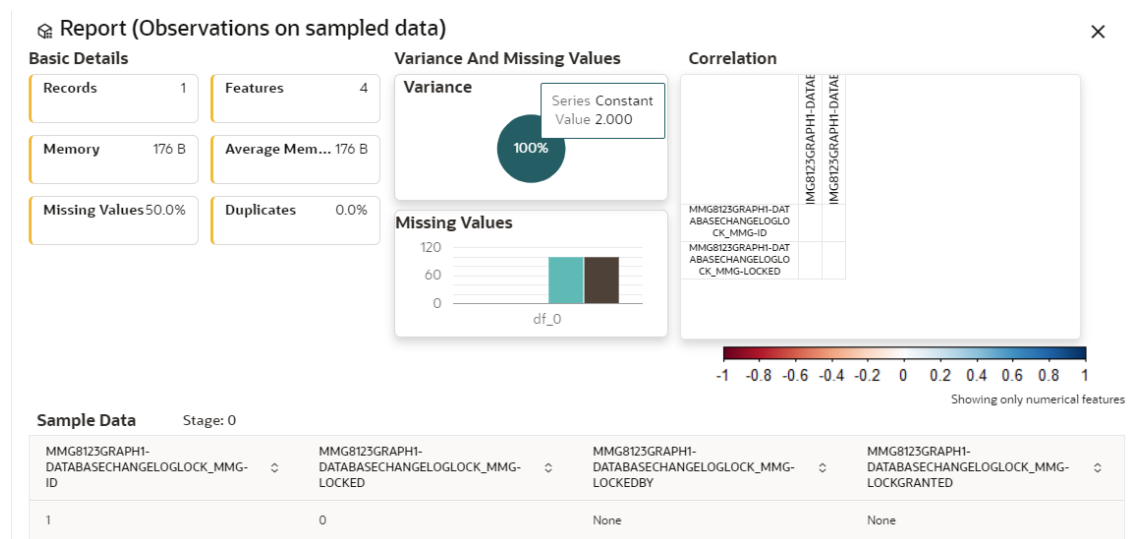
Table 11-5 (Cont.) Transformations

No.	Transformation	Function
11.	Filter Features	This function is used to filter the data frame based on conditions specified on features Operations allowed : >, >=, =, !=, <, <=, isin 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

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.

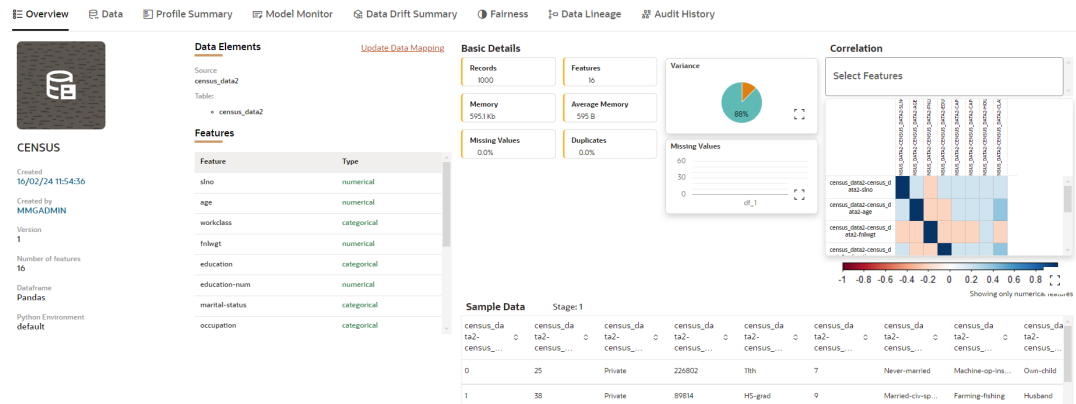
View a Dataset

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

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:

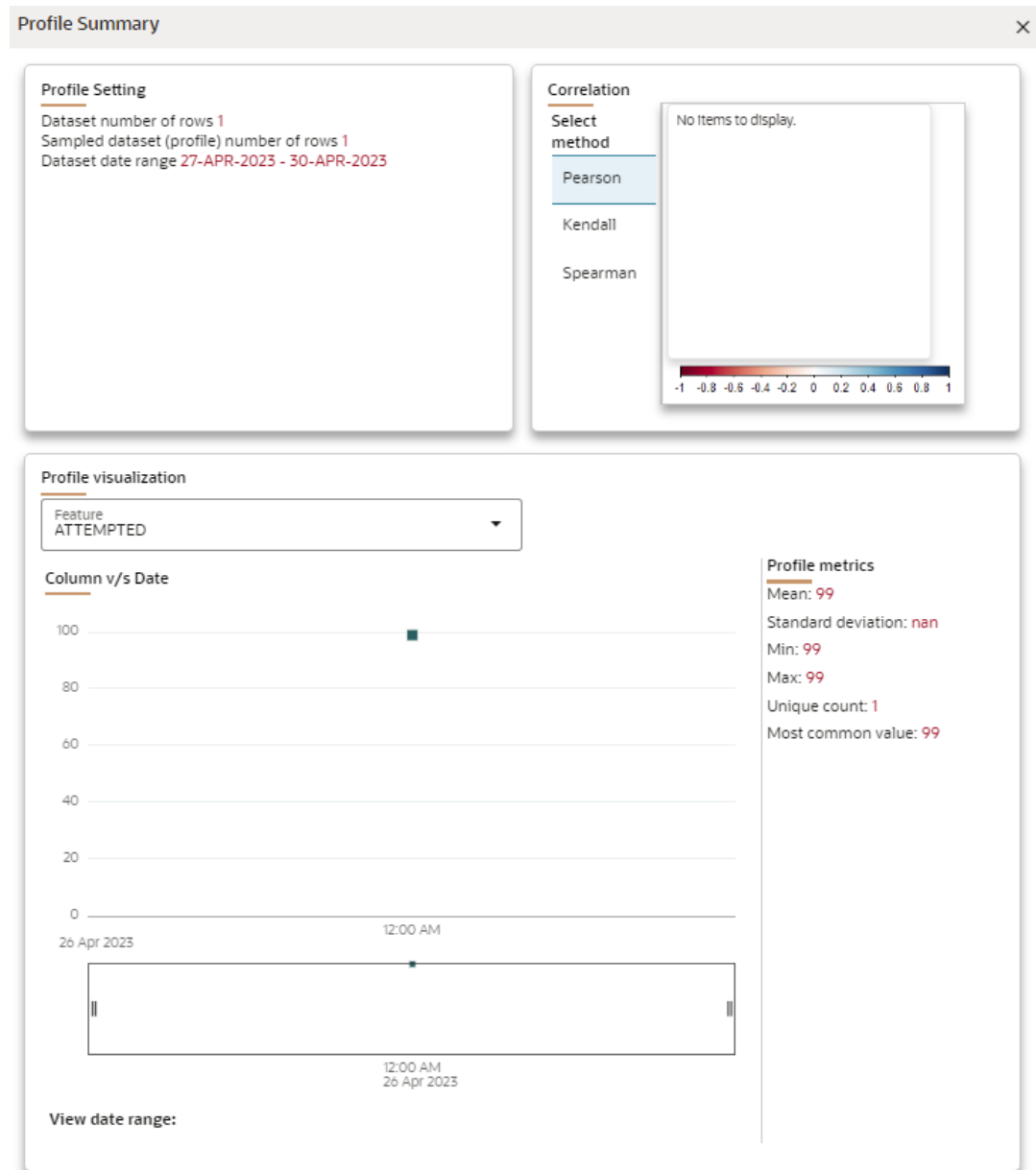
- 1. Navigate to the **Dataset Summary** page.
- 2. Click **Next** to the corresponding Dataset and select **View**.
The Overview page is displayed.
- 3. Navigate to **Data** tab.

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

The screenshot shows the 'Dataset Profiles' page with tabs for 'Profile Summary', 'Model Monitor', 'Data Drift Summary', and 'Data Lineage'. The 'Profile Summary' tab is active. It features a search bar, 'Analyze Profiles', 'Generate Profile', and 'Stop Profiling' buttons. Below is a table of profiles:

Data Profile Id	Creation Datetime	Dataset Range	Action
☒ 1_DATAPROFILER	26-APR-2023 07:42:22 AM	START - 26-APR-2023	...
☐ 2_DATAPROFILER	26-APR-2023 07:48:29 AM	26-APR-2023 - 26-APR-2023	...
☐ 3_DATAPROFILER	26-APR-2023 01:00:04 PM	26-APR-2023 - 26-APR-2023	...
☐ 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

The screenshot shows the 'Profiler Settings' dialog box. The 'Basic Information' section is expanded, showing a 'Columns' drop-down menu (labeled 'Required'), a 'Threshold' input field set to '0.1', and a 'Filter Column' drop-down menu (labeled 'Required'). There is a '+ Add Column' button and a 'Generate First Profile' toggle switch. The 'Schedule' section is partially visible below.

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:

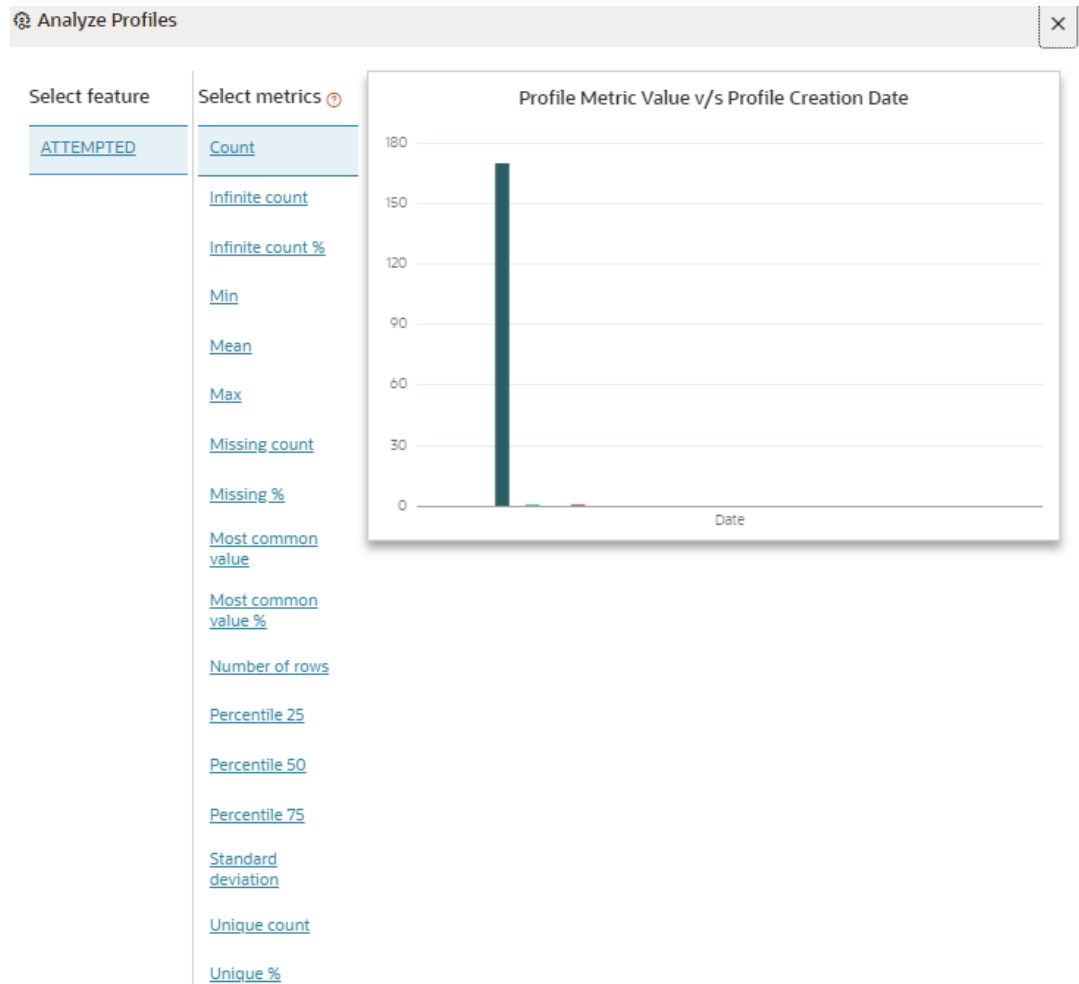
- a. In the **Profiler Settings** screen, select the Schedule Type as **Daily**.
- b. Select the start date on which you want to run the profiler.
- c. Select the end date on which you want to stop your schedule.
- d. Enter the time at which you want to run the profiler.
- e. Select the month on which 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

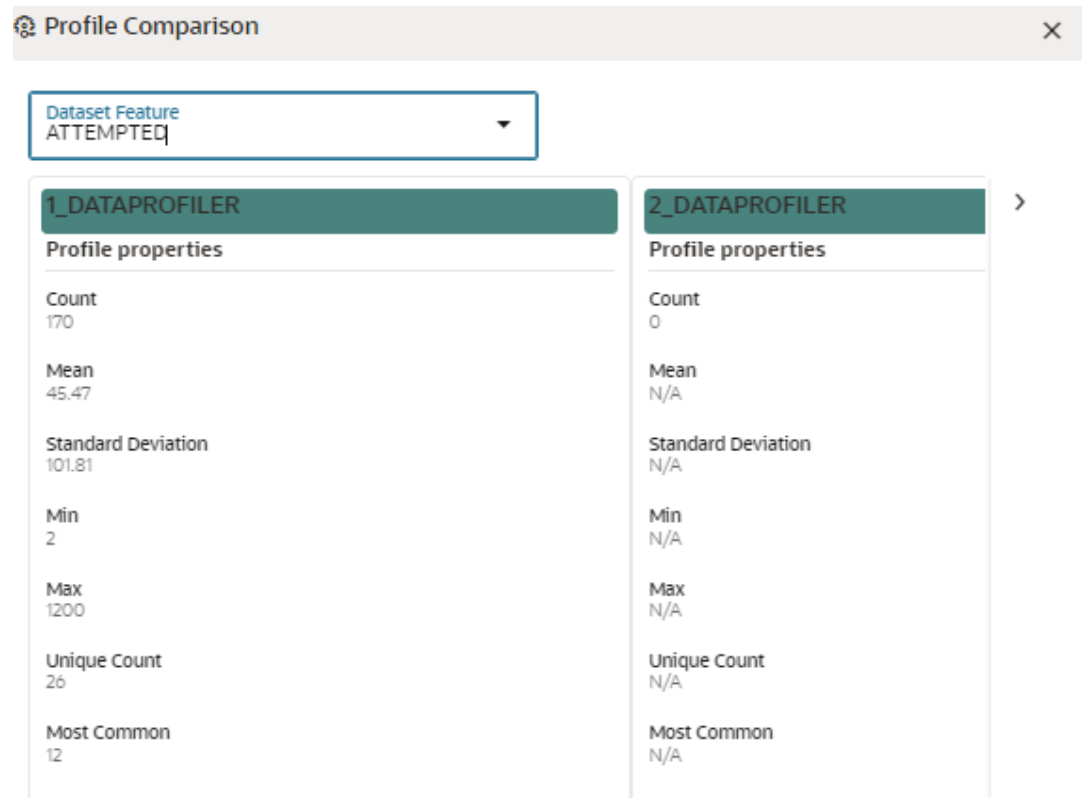
To compare the data between two data profiles, perform the following steps:

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



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	On	MBAGSAML	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.

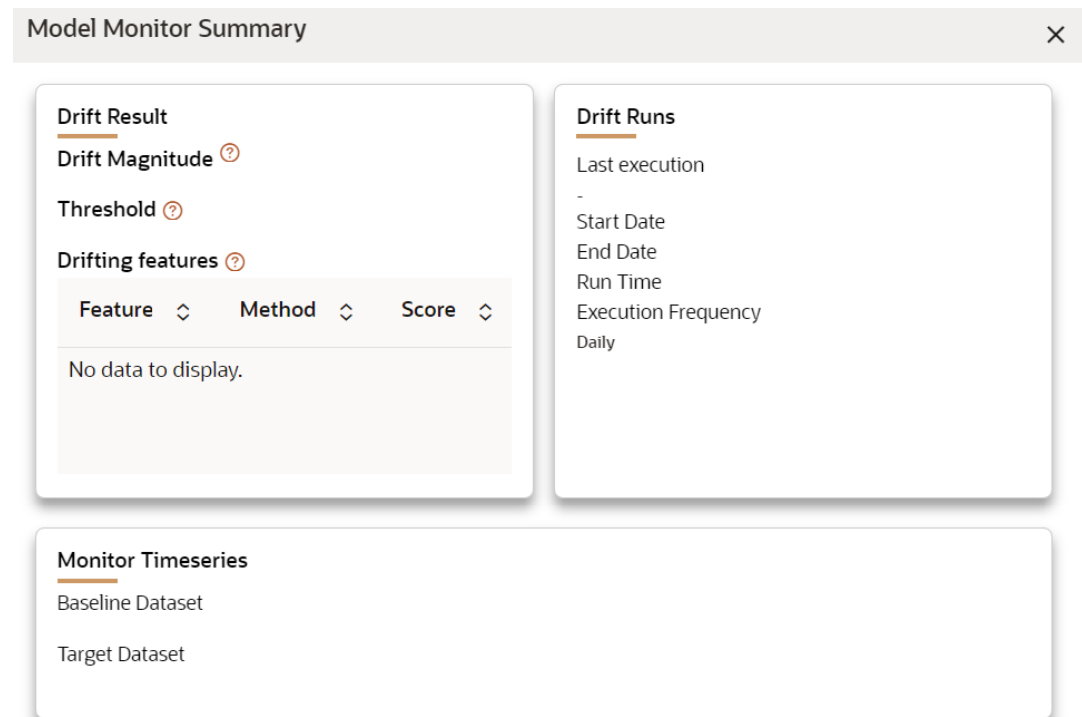
2. Click **Stop Model Monitoring**.
A confirmation pop-up window is displayed.
3. Click **Yes**.
The Model Monitoring is stopped.

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

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

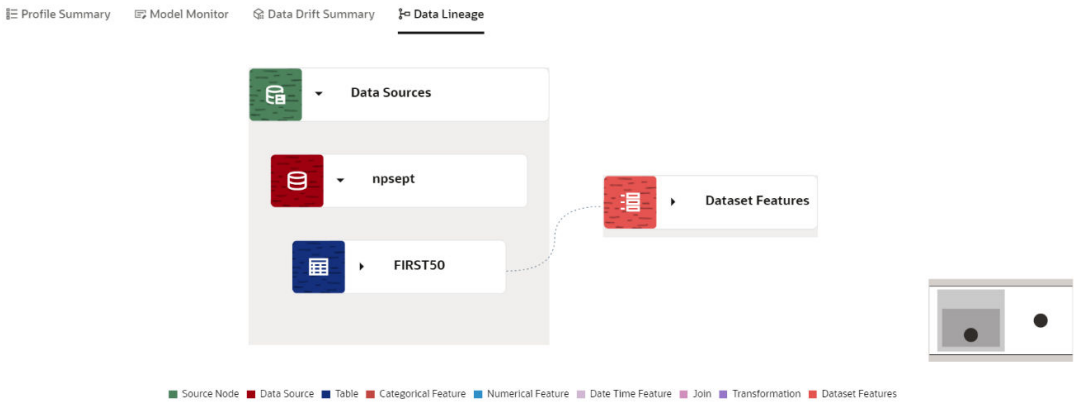
Data Lineage provides a summary view or visualization of Data Sources and its associated dataset and features in a graphical format. Also displays the complete data flow from start to finish.

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:

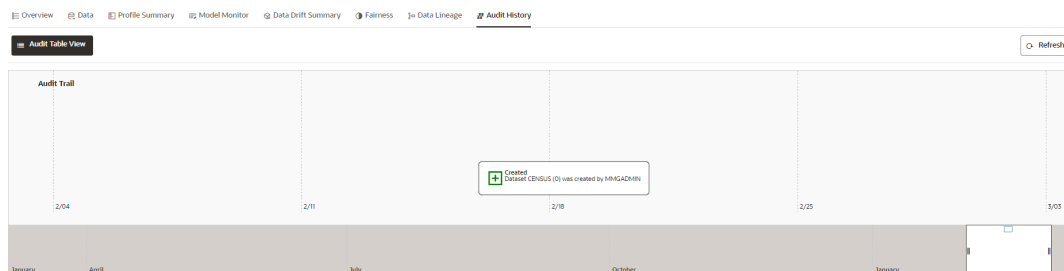
- 1. Click **Launch Workspace** next to corresponding Workspace to Launch Workspace to display the **Dashboard** window with application configuration and model creation menu.
- 2. Navigate to **Dataset Summary** page.
- 3. Click More Options icon next to corresponding Dataset and select View.
The **Overview** screen is displayed.
- 4. Navigate to **Audit History** page.

Figure 11-36 Audit History

The screenshot shows the 'Audit History' tab selected. Below the navigation bar, there's a table with columns: Status, Type, Name, User, Comments, Time. The table has one row with data: Created, Dataset, CENSUS, MMGADMIN, Dataset CENSUS (2) was created by MMGADMIN, 16 Feb 2024, 17:16:39. There's a search bar and a 'Refresh' button at the top right.

Status	Type	Name	User	Comments	Time
Created	Dataset	CENSUS	MMGADMIN	Dataset CENSUS (2) 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/ftpshare/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" error while 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	from mmg.datasets import list_tags list_tags(dscore)	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	from mmg.datasets import delete_df delete_df(dscore, tag)	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	from mmg.datasets import list_datasets list_datasets()	

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	from mmg.datasets import fetch_ds df = fetch_ds(dscore,tag)	dscore = dataset code - string tag - string
Fetch New Snapshot of Dataset	from mmg.datasets import fetch_ds df = fetch_ds(dscore)	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>	df = dataframe to be cached - pandas dataframe tag - string

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>	tag - string

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:

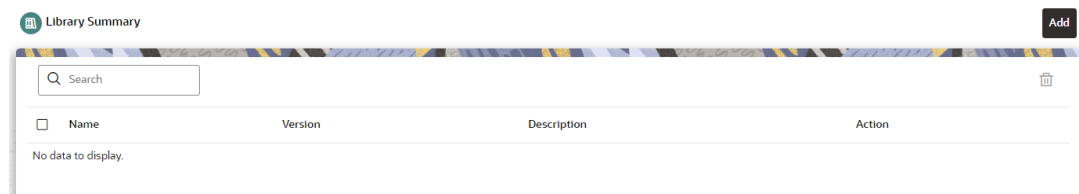
```
pip show scikit-learn
```

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

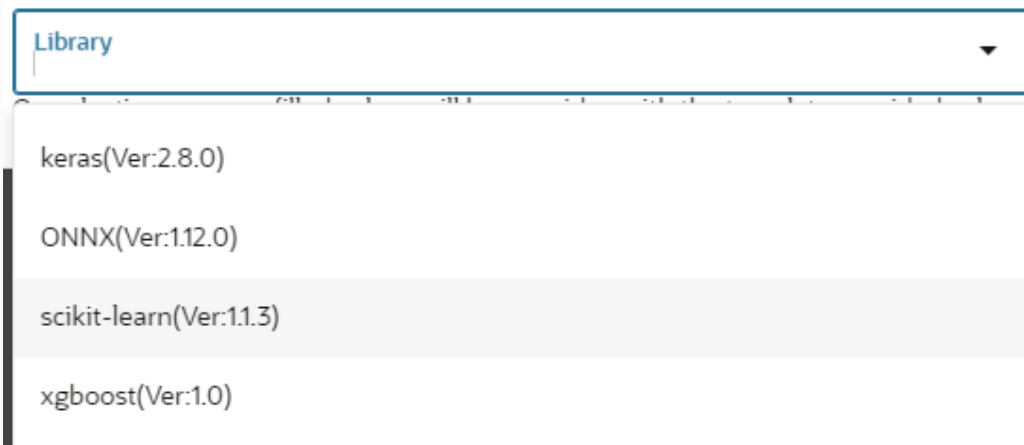
The User must check the details provided above from the home page of the library being captured.

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

Use Template

The screenshot shows a web interface with a dropdown menu titled "Library". The dropdown is open, displaying a list of libraries with their versions: "keras(Ver:2.8.0)", "ONNX(Ver:1.12.0)", "scikit-learn(Ver:1.1.3)", and "xgboost(Ver:1.0)". The "scikit-learn(Ver:1.1.3)" option is currently selected and highlighted with a grey background.

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

Edit Library

Name
keras

Version
2.8.0

Description
Tensorflow Keras Modelling library

Signatures

Import Signature
from tensorflow import keras

Load Signature
keras.models.load_model(\$MODEL_DIR_PATH)

Train Signature
keras.fit(\$df_X,\$df_Y)

Save Signature
\$MODEL.save(\$MODEL_DIR_PATH)

Infer Signature
\$MODEL.predict(\$DF)

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

Delete Objects

Are you sure you want to delete selected model library? This process can not be undone.

☒ Code

☒ xgboost(Ver:1.0)

Name	Description
xgboost	Xgboost Library for Modeling

To view the Techniques that will be deleted with Library, click

These items are not consumed by any Model Catalogs.

Cancel

Delete

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.

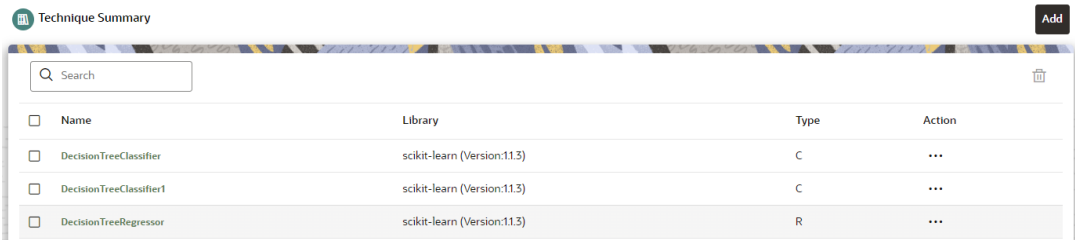
To delete multiple model libraries select all the model libraries that you want to delete and click the icon on the page header.

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



2. In the **Technique Summary** page, click **Add**.
- The **Add Technique** page is displayed.

Figure 11-44 Add Technique page

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.

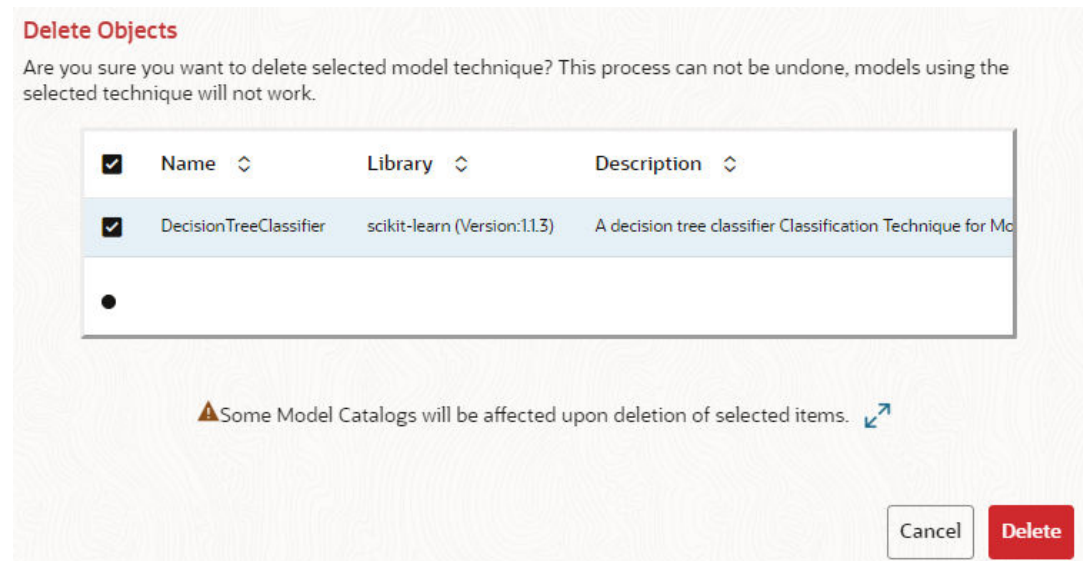
Delete a Model Technique

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** feature allows you to manage your Model Catalogs. 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

On the **Objective Summary** page, use the **Show Empty Objectives** checkbox to toggle the display of empty objectives. By default, empty objectives are hidden. You can select **Show Empty Objectives** to display them. After you successfully add a new objective, **Show Empty Objectives** is automatically enabled.

Add a Model Catalog

To add a Model Catalog:

1. In the **Model Catalog Summary** screen, click **Add**.

The **Add Model Catalog** screen is displayed.

Figure 11-47 Add Model Catalog

Add Model Catalog

Are you creating a generic catalog?

☐ Yes ☒ No

Select Dataset Required

Select Target Variable Required

Code Required

Name Required

Type Required

Description

Cancel Create

2. Select either *Yes* or *No* in the **Are you creating a generic catalog?** section. By default, the *No* option is selected on this window. If you select *Yes*, the following options appears:

Figure 11-48 Add Model Catalog Window with the Yes Option Selected

Add Model Catalog

Are you creating a generic catalog?

☒ Yes ☐ No

Code Required

Name Required

Type Required

Description Required

Cancel Create

3. Select the desired dataset from the **Select Dataset** drop-down list. This option is only available if *No* is the selected option in the **Are you creating a generic catalog?** section. For more details on datasets, see [Dataset](#) section.

4. In the **Select Target Variable** field, select one of the column/feature as the target variable for the models that needs to be trained/uploaded. This option is only available if *No* is the selected option in the **Are you creating a generic catalog?** section.

5. Enter a code for the model catalog in the **Code** field.
The code should be unique for each model catalog.

6. Enter a name in the **Name** field.

7. Select the type from the **Type** drop-down list. The available options are:

Classification: Classification is the process of finding a function that 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 the 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 in this field is not **Classification** or **Regression**, then select this option.

8. Enter the description in the **Description** field.

9. Click **Create**.

The Model Catalog is created and displayed in the **Model Catalog Summary** screen.

View a Model Catalog

To view a Model Catalog:

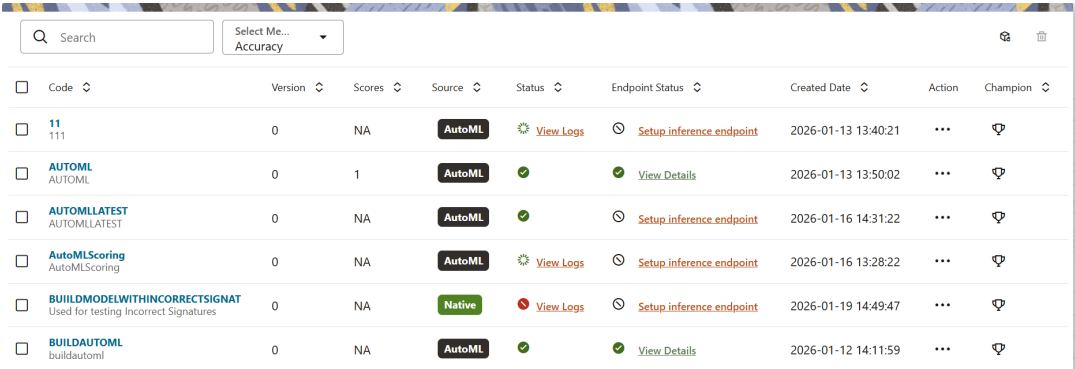
1. Navigate to the **Model Catalog Summary** page.
2. Click **Action** and select **View**.

OR

Click the desired model.

The following page is displayed.

Figure 11-49 View Model Page



☐	Code ↕	Version ↕	Scores ↕	Source ↕	Status ↕	Endpoint Status ↕	Created Date ↕	Action	Champion ↕
☐	11 111	0	NA	AutoML	View Logs	Setup inference endpoint	2026-01-13 13:40:21	...	
☐	AUTOML AUTOML	0	1	AutoML		View Details	2026-01-13 13:50:02	...	
☐	AUTOMLATEST AUTOMLATEST	0	NA	AutoML		Setup inference endpoint	2026-01-16 14:31:22	...	
☐	AutoMLScoring AutoMLScoring	0	NA	AutoML	View Logs	Setup inference endpoint	2026-01-16 13:28:22	...	
☐	BUILDMODELWITHINCORRECTSIGNAT Used for testing Incorrect Signatures	0	NA	Native	View Logs	Setup inference endpoint	2026-01-19 14:49:47	...	
☐	BUILDAUTOML buildautoml	0	NA	AutoML		View Details	2026-01-12 14:11:59	...	

Click a model to view it's details.

Figure 11-50 Model Details Drawer

The screenshot shows a 'Model Details' drawer window with a close button (X) in the top right corner. Below the title bar are five tabs: 'Basic Details', 'Parameters', 'Scores', 'Serve', and 'Logs'. The 'Basic Details' tab is selected and underlined. It contains six fields arranged in two columns:

Field Name	Value
Model Code	BIASMITIGATION
Description	environment_py312
Library Name	scikit-learn (Version:1.5.0)
Technique Name	SVC
Created By	MMGADMIN
Created Date	2026-01-20 14:16:47

At the bottom right of the drawer is a 'Close' button.

3. If a model has failed, the **View Log** link appears in the **Status** column of the **Model Catalog Summary** page. When you click this link, the **Model Details** window appears. Navigate to the **Logs** tab to view the reason for the failure and the path to the file that failed.

Upload a Model

To upload a model:

1. In the **View Model Objective** page, click **Add** and then select **Upload Model**.
The **Upload Model** page is displayed.

Figure 11-51 Upload Model Drawer

Upload Model

Select Library

Select Technique

Model Code

Description

Python Runtime default

Model File Name

Upload from FTPSHARE

Import File

Cancel Upload Model

2. Select a library from the **Select Library** drop-down list. When a library is selected, the **Select Technique** drop-down list appears
For more details on the model library, see the [Model Libraries](#) section.
3. Select the required technique from the **Select Technique** drop-down list.
For more details on the model technique, see the [Model Technique](#) section.
4. Enter the model code and description for the model catalog in the **Model Code** and **Description** fields.
5. Select a python runtime from the **Python Runtime** drop-down list.
6. Enter the name of model file with the extension in the **Model File Name** field.
7. Import the model by using the **Import Archive File** option. The file format must be in be a zip format.

OR

Enable the **Upload from FTPSHARE** option to import the model. The file path that is entered is the zip file path that is present on the server relative to ftpshare path, including the file.

For example: <installation path>/temp/models/model.zip

8. Click **Upload Model**.

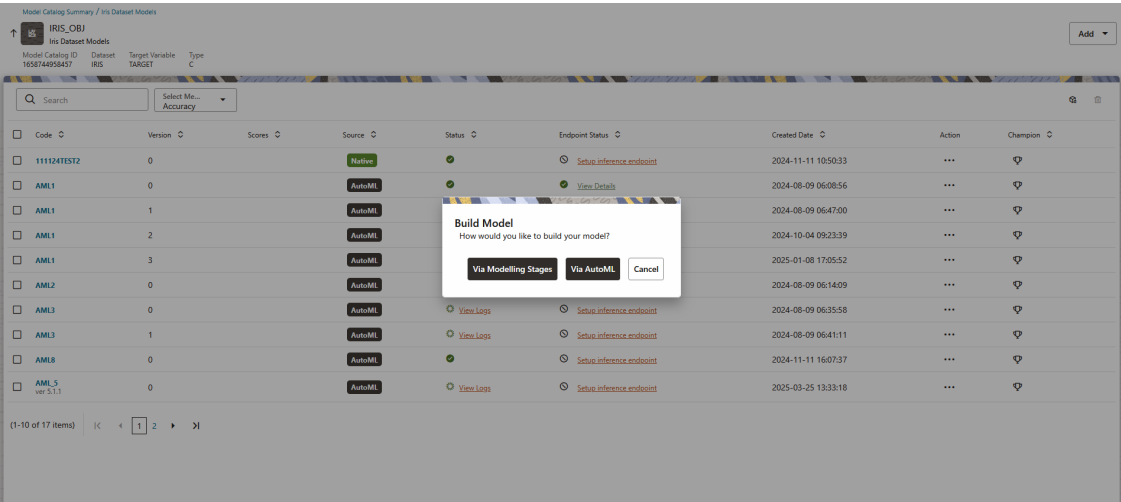
The model is uploaded successfully.

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 11-52 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 11-53 Model Details page

1 Model Details 2 Pre-Processing Stage 3 Model Training Stage 4 Model Validation Stage 5 Model Summary

Select Library Required

Select Technique

Model Code Required

Description

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 11-54 Pre-Processing Stage screen

1 Model Details 2 Pre-Processing Stage 3 Model Training Stage 4 Model Validation Stage 5 Model Summary

Preprocessing

1. Enter the data in the Pre-Processing text box as shown in an example is provided here.

Figure 11-55 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 11-56 Model Training Stage screen

Parameter	Value	Placeholder	Enabled
kernel (String)	poly	\$kernel\$	☒
degree (Integer)...	3	\$degree\$	☒
C (Float)	1.1	\$C\$	☒
shrinking (Boolean)		\$shrinking\$	☐
probability (Boolean)	True	\$probability\$	☒
gamma (String)		\$gamma\$	☐
C (Float)		\$C\$	☐

Generate Train Signature

```
SVC(C=1.1, 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.

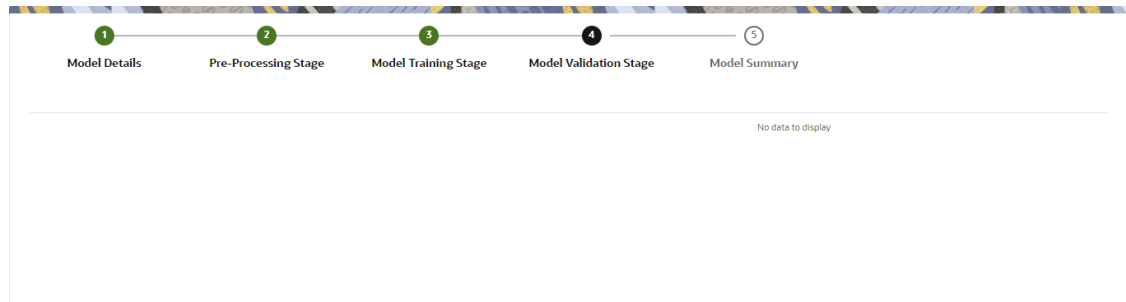
2. Click **Generate Train Signature**.

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 11-57 Model Validation Stage screen

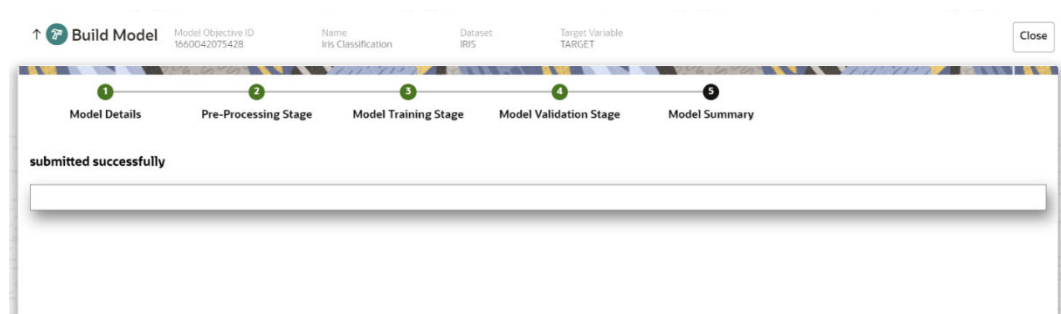
The Validation is work in progress.

- 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 11-58 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`

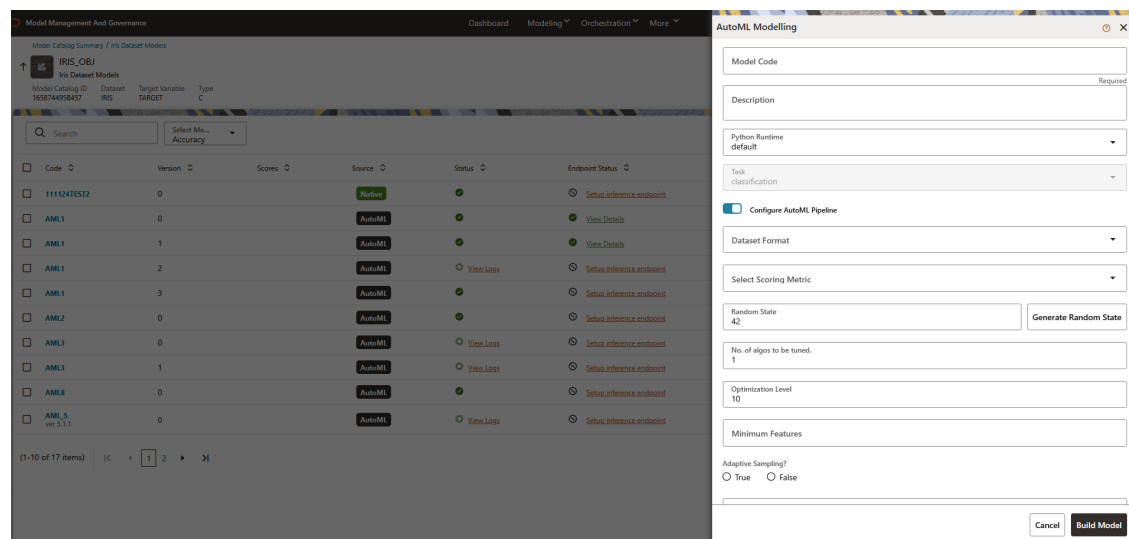
The following is the command to install AutoML: `python3 -m pip install oracle-automlx[classic]==25.2.1`

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 11-59 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.

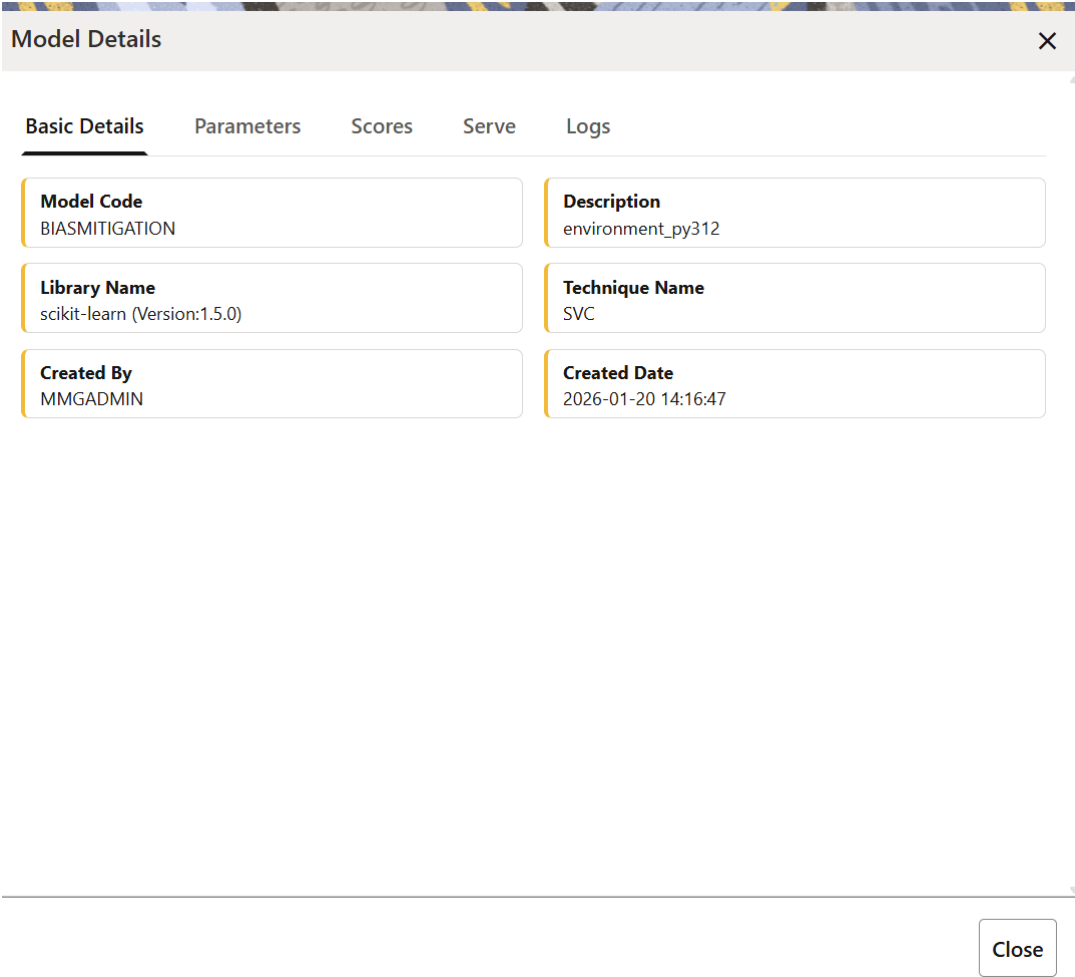
5. Click on the **Build Model**.
An AutoML model will be created and displayed in the **Model Objectives** page.

View the Details of an AutoML Model

To view details of an AutoML model, perform the following steps:

1. On the **View Model Catalog** page, click **Action** next to corresponding AutoML model and select **View Details**.
The AutoML model details are displayed.

Figure 11-60 AutoML Model Details Drawer



The following image illustrates the AutoML Summary tab

Figure 11-61 AutoML Summary Tab

Model Details	
AutoML Configurations	AutoML Summary
▼ SVC	
Name C	Value 1
Name seed	Value 7
Name reproducible	Value false
Name cache_size	Value 800
Name inference_mode	Value predict
Name class_weight	Value balanced
Name predict_proba	Value false
Name sample_weight	Value
Name gamma	Value 0.000030517578125
Name selected_features_names	Value PETAL_LENGTH

2. Click the tabs to view the following details:
 - **Basic Details:** This tab displays the basic details of the AutoML model catalog such as the model code, description, library name, technique name, created by, and the created date.
 - **Parameters:** This tab will display the task and type details in the **AutoML Configurations** tab if the **Configure AutoML Pipeline** was not enabled in the **AutoML Modeling** window. If the **Configure AutoML Pipeline** option was enabled in the **AutoML Modeling** window then it will display additional configuration details. The **AutoML Summary** tab displays the model techniques that are used when building the AutoML model along with other configuration details.
 - **Scores:** This tab displays information about the latest evaluation score, evaluation history, fairness score.
 - **Serve:** This tab displays the endpoint details of a deployed AutoML model.

- **Logs:** This tab displays the log information of the selected model. In case a model has failed, you can view the reason for the failure.
3. Click the **Score Details** tab to view the Latest score, and Scoring details of the AutoML model.

Download a Model

Perform the following steps to download a model from the **View Model Catalog** page:

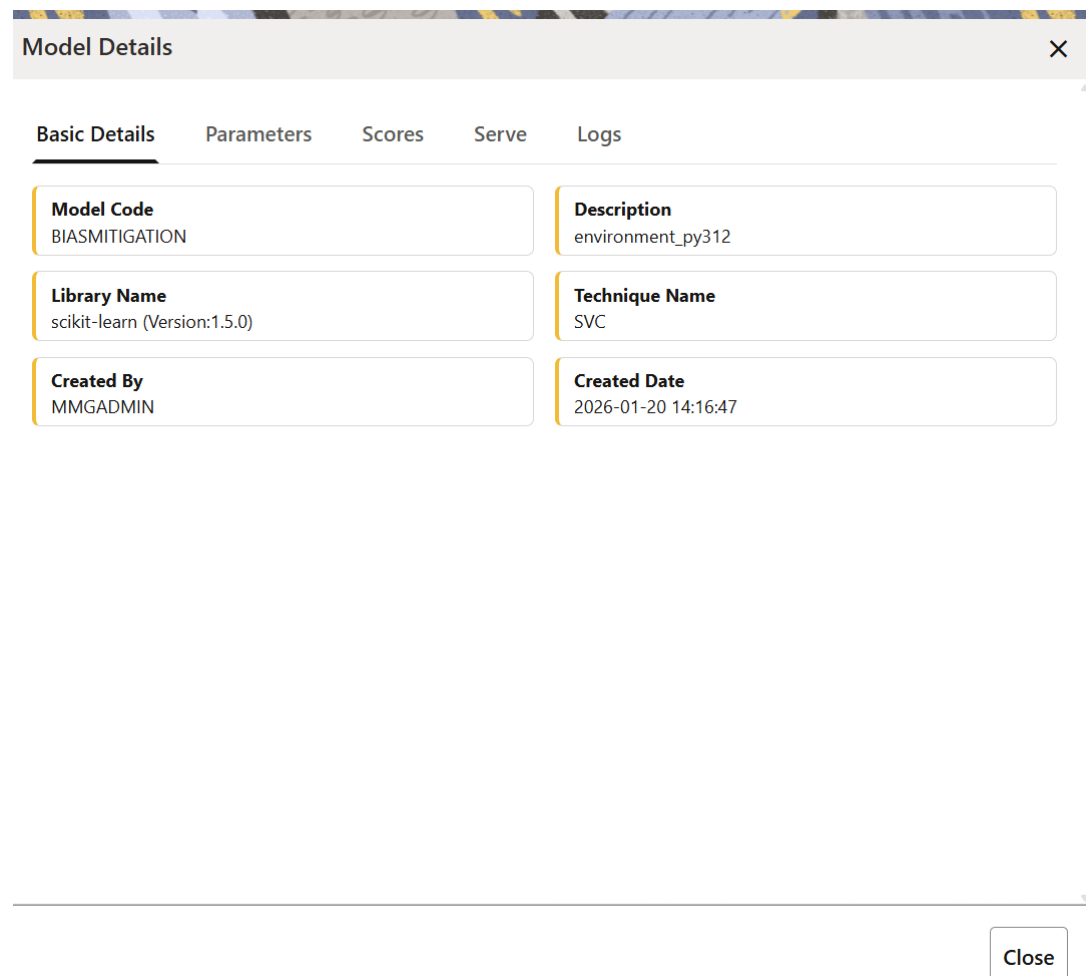
1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Download**.
The selected model will be downloaded in a .zip format.

View the Details of a Model

To view details of a model, perform the following steps:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **View Details**.
The model details are displayed.

Figure 11-62 Model Details Drawer



2. Click the tabs to view the following details:
 - **Basic Details:** This tab displays the basic details of the model catalog such as the model code, description, library name, technique name, created by, and the created date.
 - **Parameters:** This tab displays the preprocessing details of the model in the **Preprocessing** tab. The **Hyperparameter Details** tab displays the different names, placeholder, type, and values that are used during the training of the selected model.
 - **Scores:** This tab displays information about the latest evaluation score, evaluation history, fairness score.
 - **Serve:** This tab displays the endpoint details of a deployed model.
 - **Logs:** This tab displays the log information of the selected model. In case a model has failed, you can view the reason for the failure.
3. Click the **Score Details** tab to view the Latest score, and Scoring details of the model.

Delete a Model

To delete a model:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Delete**. A confirmation message appears, asking if you want to delete the selected model.
2. Click yes to delete the selected model.

Score a Model

Perform the following steps to score a model from the **View Model Catalog** page:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Score Model**.
The **Score Model** drawer appears.

Figure 11-63 Score Model Drawer

Score Model [X]

Training Dataset: IRISDS

Model Catalog ID: 1768311621082

Select Scoring Dataset [v] Required

Scoring Metrics [v] Required

Conda Environment: default [v]

Average Type [v]

Do you want to split the dataset for scoring?

☐ Yes ☒ No

[Cancel] [Score Model]

2. Select the scoring dataset from the **Select Scoring Dataset** drop-down menu.
3. Select a scoring metric from the **Scoring Metrics** drop-down list.
4. Select a conda environment from the **Conda Environment** drop-down list.
The selected Conda environment is sourced from the model context and is passed through to the backend for each operation so that the appropriate environment can be activated before executing the Python code. The Conda environment can be selected for models that are built from model catalog or model pipeline.
5. Select the average type from the **Average Type** drop-down list.
6. Select either Yes or No in the **Do you want to split the dataset for scoring** section as per your requirement.
7. Click **Score Model**.

Model Bias Mitigation

Perform the following steps to score a model from the **View Model Catalog** page:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Bias Mitigation**.
The **Model Bias Mitigation** drawer appears.

Figure 11-64 Model Bias Mitigation

Model Bias Mitigation [X]

Training Dataset: IRISDS

Model Catalog ID: 1768311621082

Select Fairness Metric [v] Required

Select Scoring Metric [v] Required

Select Protected Features [v] Required

Conda Environment: default [v]

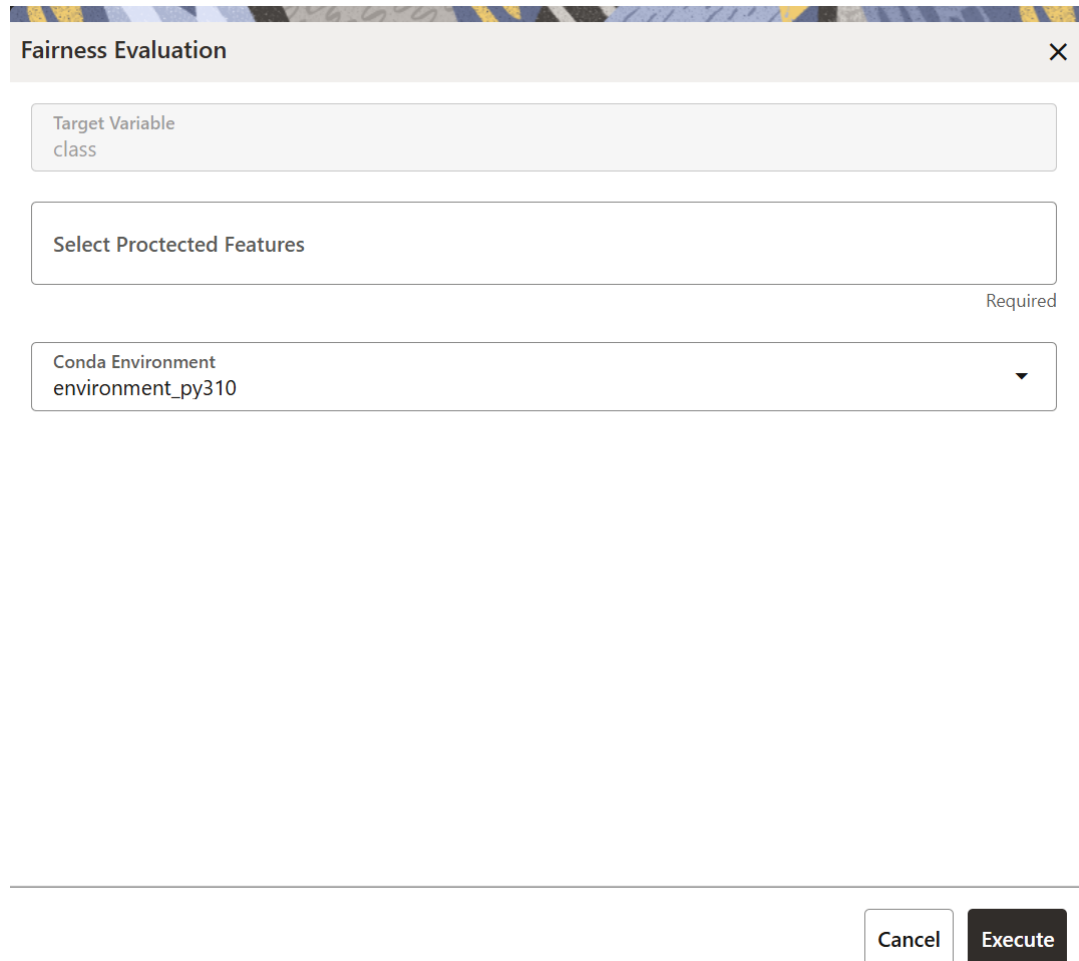
[Cancel] [Execute]

2. Select the fairness metric from the **Select Fairness Metric** drop-down menu.
3. Select a scoring metric from the **Select Scoring Metrics** drop-down list.
4. Select a protected feature from the **Select Protected Features** drop-down list.
5. Select a conda environment from the **Conda Environment** drop-down list.
The selected Conda environment is sourced from the model context and is passed through to the backend for each operation so that the appropriate environment can be activated before executing the Python code. The Conda environment can be selected for models that are built from model catalog or model pipeline.
6. Select the average type from the **Average Type** drop-down list.
7. Click **Execute**.

Fairness Evaluation

Perform the following steps to score a model from the **View Model Catalog** page:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Fairness Evaluation**.
The **Fairness Evaluation** drawer appears.

Figure 11-65 Fairness Evaluation

The image shows a 'Fairness Evaluation' dialog box with a title bar and a close button (X). It contains three input fields: 'Target Variable class', 'Select Protected Features', and 'Conda Environment environment_py310'. The 'Select Protected Features' field is marked as 'Required'. At the bottom right, there are 'Cancel' and 'Execute' buttons.

Fairness Evaluation

Target Variable
class

Select Protected Features

Required

Conda Environment
environment_py310

Cancel Execute

2. Select a protected feature from the **Select Protected Features** drop-down list.
3. Select a conda environment from the **Conda Environment** drop-down list.
The selected Conda environment is sourced from the model context and is passed through to the backend for each operation so that the appropriate environment can be activated before executing the Python code. The Conda environment can be selected for models that are built from model catalog or model pipeline.
4. Click **Execute**.

Deploy a Model

Perform the following steps to score a model from the **View Model Catalog** page:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Deploy Model**.
The **Deploy Model** drawer appears.

Figure 11-66 Deploy Model

Deploy Model

▼ **Basic Information**

Endpoint

☒ New Endpoint ☐ Existing Endpoint

Endpoint Code

Endpoint Port
2000 Required

[Check Port Status](#)

Description

Authentication Mode

☒ Key-based Authentication ☐ No Authentication

> **Scoring Script**

Cancel Deploy

2. Select either *New Endpoint* or *Existing Endpoint* from the **Endpoint** section.
3. Enter the endpoint code in the **Endpoint Code** field.
4. Enter the value of the endpoint port in the **Endpoint Port** field.
5. Enter a description in the **Description** field.
6. Select either *Key-based Authentication* or *No Authentication* from the **Authentication Mode** section.
7. In the **Scoring Script** section, either enter the scoring script or drag and drop the script into the **Upload Scoring Script** section.
8. Click **Execute**.

Set the Model as Champion

You can set the trained models as champion model.

To set a champion model:

1. In the **View Model Catalog** page, click **Action** next to corresponding model and select **Set Champion**.

A confirmation message appears, asking if you want to set the selected model as a champion.

2. Click **Update**.

The model is set as champion for the selected model objective.

Delete a Model Catalog

To delete a Model Catalog:

1. Navigate to **Model Catalog Summary** page.
2. Select the model that you want to delete and then click the **Delete** icon. A confirmation message appears asking you if you want to delete the model catalog.
3. Confirm the deletion and the selected model catalog is 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:

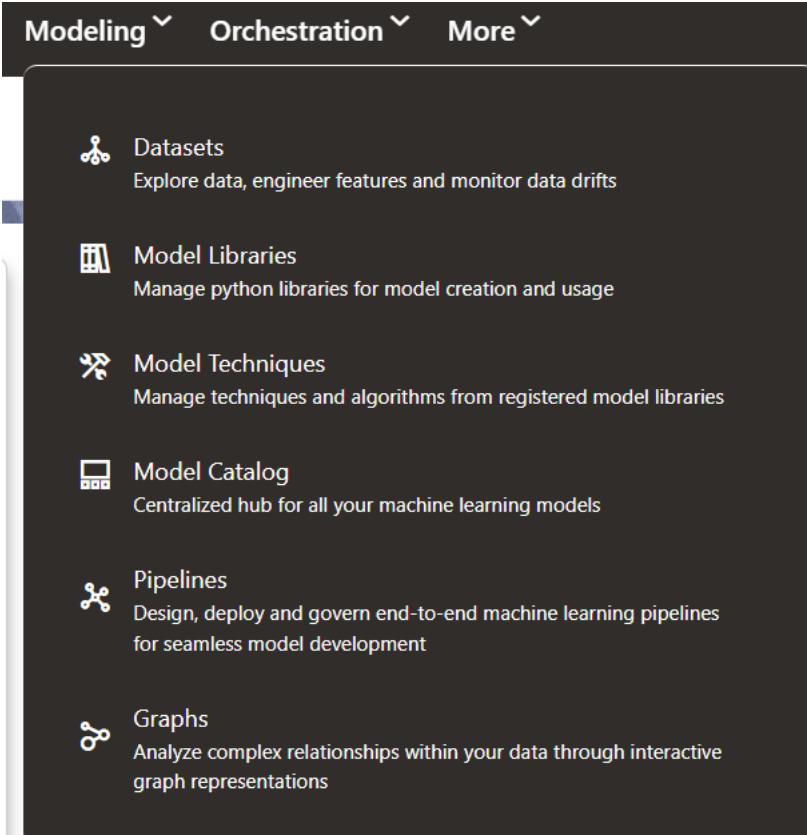
- To create a model, your user profile must be mapped to the Modeler Group.
- To create a model, a workspace must be deployed.
- To approve and deploy a model, your user profile must be mapped to the Modeling Administrator Group.

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.
3. 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.

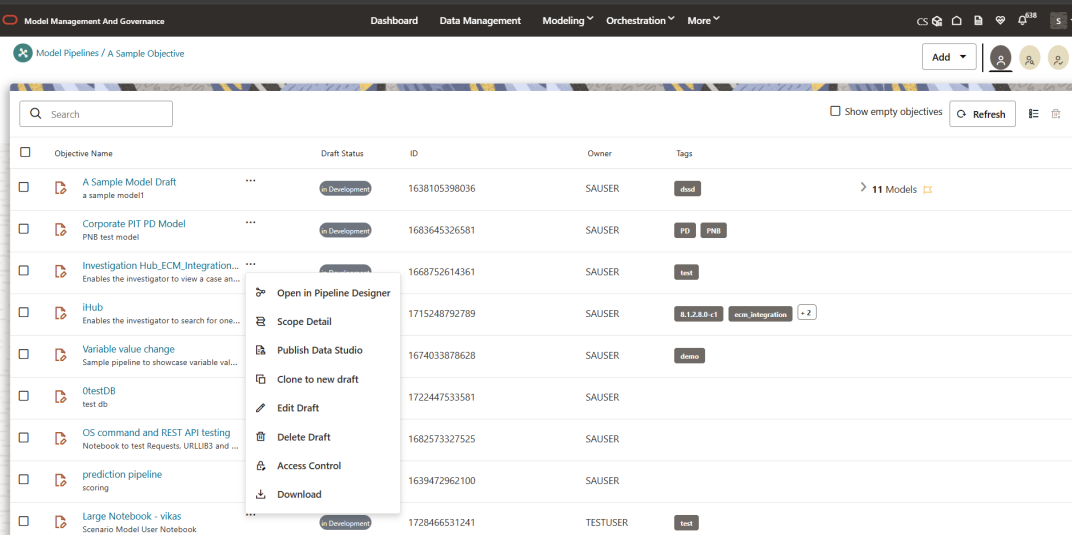
Figure 11-67 Model Pipelines page

Model Pipelines Add Icons

☐	Objective Name	ID	
☐	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 POX for AIF4AML	*** AIF0000019	

Click the **More** icon adjacent to an objective to view all the available actions.

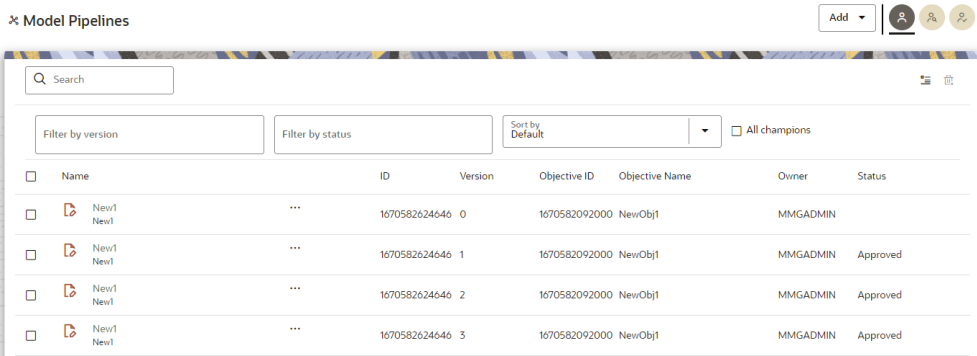
Figure 11-68 Actions for an Objective



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 11-69 Model Pipelines



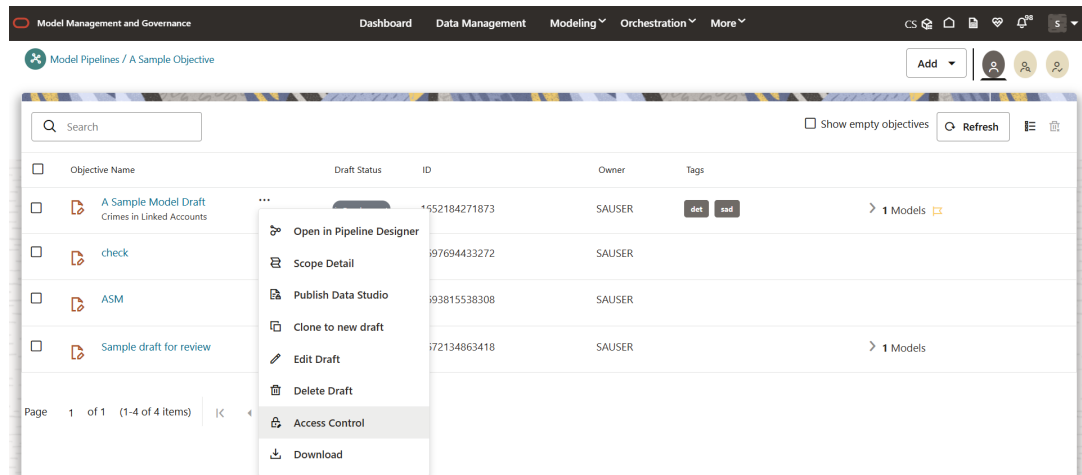
The following table provides descriptions for the fields and icons on the Model Pipeline page.

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: • 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. When you are creating a Model by using the extraction widget, you might want to increase the ZEPPPELIN_LIMIT_INTERPETER_OUTPUT in the Python Interpreter. For information on how to do this, see the question on how to increase the ZEPPPELIN_LIMIT_INTERPETER_OUTPUT in the Python Interpreter in the Frequently Asked Questions topic of the OFS MMG Installation Guide. However, if you want to know about the cycle of model creation, see the Manage Models 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 11-70 Access Control Menu option

The **Access Control** window has two radio buttons: **Public** and **Restricted**.

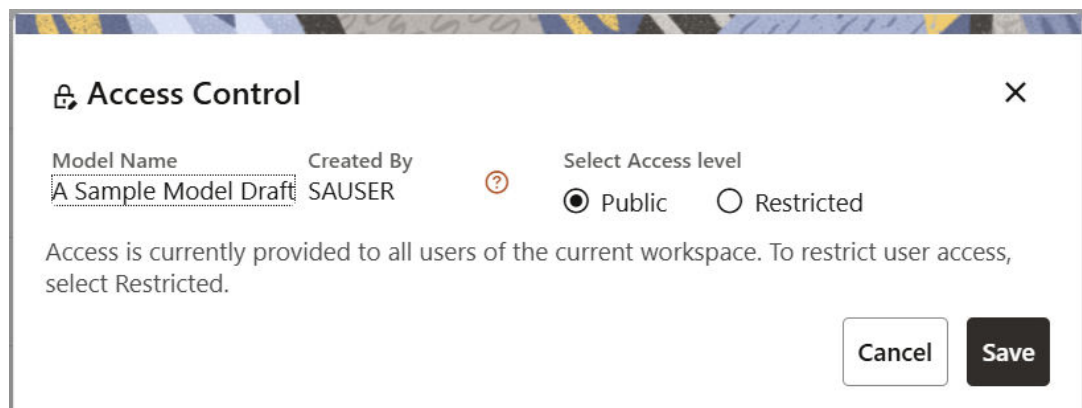
Figure 11-71 Public Mode

Figure 11-72 Restricted Mode

The screenshot shows the 'Access Control' dialog box. At the top, it displays 'Model Name' as 'A Sample Model Draft' and 'Created By' as 'SAUSER'. A tooltip points to the 'SAUSER' text, stating 'SAUSER has created this object and will always have access to it'. Below this, there is a 'Select Access level' section with two radio buttons: 'Public' and 'Restricted'. The 'Restricted' option is selected. A message below the radio buttons states: 'Access is currently restricted to selected users and user groups. To give unrestricted access to all users of this workspace, select Public.' There are two large text input fields labeled 'User Access' and 'User Group Access'. At the bottom right, there are 'Cancel' and 'Save' buttons.

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 11-73 Access Control User Dropdown

Access Control ×

Model Name Created By ? Select Access level

A Sample Model Draft SAUSER ☐ 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

More results available, please filter further.

- CSLEARNER
- DSTEST
- JVELE**
- MMGADMIN
- KAFKAREADER
- MMGDEVADMIN
- NANDHA

- **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 11-74 Access Control User Dropdown

The screenshot shows the 'Access Control' dialog box. At the top, it displays 'Model Name: A Sample Model Draft' and 'Created By: SAUSER'. Below this, there is a section for 'Select Access level' with two radio buttons: 'Public' (unselected) and 'Restricted' (selected). 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.' Below the message is a 'User Access' dropdown menu. The dropdown is open, showing a list of users: CSLEARNER, DSTEST, JVELE (highlighted), MMGADMIN, KAFKAREADER, MMGDEVADMIN, and NANDHA. A scroll bar is visible on the right side of the dropdown list.

- 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:

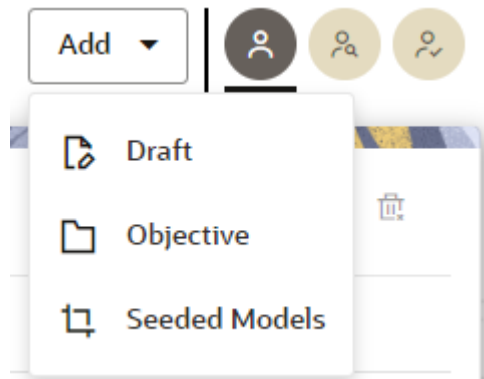
- Create Objective (Folders)
- Create Draft Models
- Publish Models (Scoring)
- View Model Details
- Compare Models
- Understand Model Governance
- Request Model Acceptance
- Review Models and Move to Approve or Reject
- Approve Models and Promote to Production
- Deploy Models in Production and Make 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 11-75 Objective Option



4. Enter details in the **Name**, **Description**, and **Tags** fields.

Figure 11-76 The Add Objective Window

 A screenshot of a modal window titled 'Add Objective' with a close button (X) in the top right corner. The window contains three text input fields: 'Objective Name' (with a 'Required' label to its right), 'Description', and 'Tags'. At the bottom right of the window are two buttons: 'Cancel' and 'Save'.

5. Click **Save**.

Create an Objective by Supplying the Custom Objective ID and Name

Whenever a new objective is created, the application generates a unique ID that is associated with the objective. An objective can also be created by supplying the custom objective ID and name via an API.

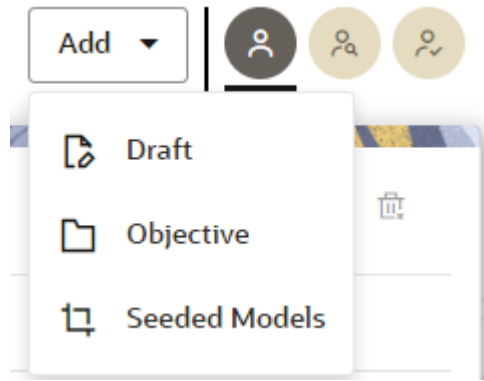
Create Draft Models

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 11-77 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.

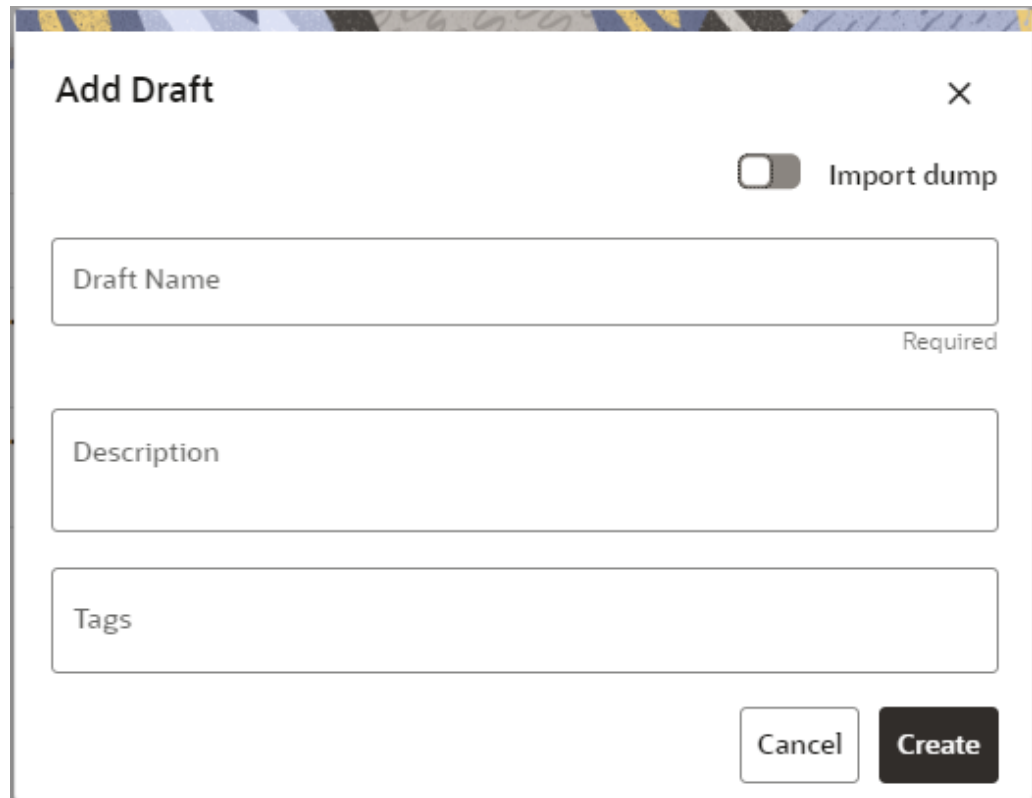
OR

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 11-78 Model Details- Create New Model



Add Draft ×

☐ Import dump

Draft Name Required

Description

Tags

Cancel 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 11-79 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 11-80 Objective Details

Add Draft ✕

☐ Import dump

Objective Details ✕

Objective
ROOT

Objective Name

Description

Tags

Required

Draft Details [Pull from Workspace](#) [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 11-81 Pull from Workspace

Add Draft ×

Objective
ANCHOR_TAG ▼ + ☐ Import dump

Draft Details [Pull from Workspace](#) [Use Template](#)

Workspace ▼

Objective ▼

Model Pipeline ▼

Add comments Cancel Pull

- **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 11-82 Use Template

Use Template

Templates ▼

⚠ On selecting, any pre-filled values will be overridden with the template provided values.

On selecting, any pre-filled values will be overridden with the template provided values.

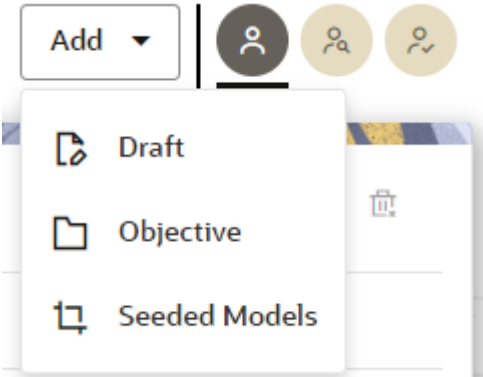
Select a template from the **Templates** drop-down list. Note that any pre-filled values will be overridden with the values from the template that has been selected from the **Templates** drop-down list.

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 11-83 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 11-84 Seeded Models

Seeded Models							
☐	Objective Path	Model Name	Model Description	Version	Seeded File Name		
☐	Test/NewDraft	NewDraft	Supervised learning analysis of loan application frauds	0	CS_1670477976040_0.zip		
☐	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.

Note

The function, Model Report View-All, is mapped by default to the role,MMGMDLRPTREAD, and the groups MDLREV, MDLAPPR, and MDLUSR. If this function is unmapped, then the **Generate Model Report** button will be disabled. You must map this function to enable the **Generate Model Report** option.

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

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

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.

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.

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 pipeline using pipeline:

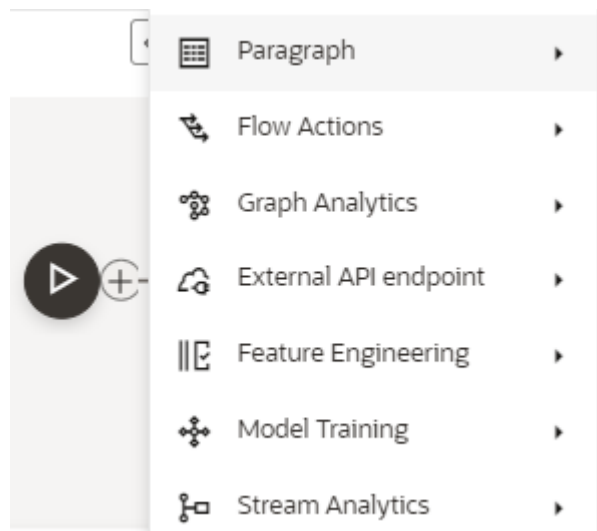
1. Navigate to the **Pipeline Designer** page.

The **Start** widget is displayed by default in the canvas screen.

2. Hover over the **Start** widget and new nodes can be added by using the **Add** button in the **Pipeline Canvas**.

When the **Add** button is clicked, a menu opens that contains all the widgets sectioned into different categories. The spectra diagram builder is based on row and column positioning.

Figure 11-85 Create a Pipeline menu



Note

Click each menu option to select the required widgets.

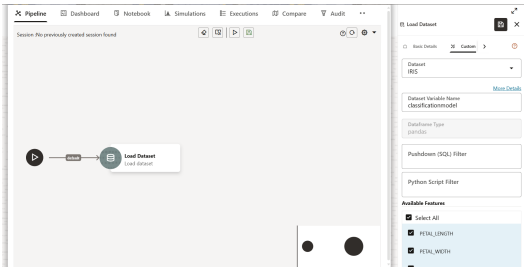
You cannot edit or delete the **Start** widget. Whenever a new draft is created, not by importing dump files, the default paragraph that is created is converted into a **Start** widget. The visibility of code or result or the title in the notebook of this node will be invisible.

Whenever the notebook is opened, the script execution, including the workspace attachment will happen in this node.

Table 11-17 Widgets

Widget	Description
Load Dataset Widget	Push Down Filtering (SQL Filters) and Python Script Filtering is available for the Load Dataset widget in the pipeline.

Figure 11-86 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]
```

Start Widget

Publish, download, import, and promoting a model to production can be done via this widget. When you publish a model from the **Model Summary** page, you can select the **Start** widget paragraph from the list of paragraphs and then the start widget will appear in the published model.

Table 11-17 (Cont.) Widgets

Widget	Description
Exploratory Data Analysis (EDA) Widget	This widget provides a one-click, automated evaluation of any dataset within OFS MMG. It provides a rapid insight into the structure, quality, and characteristics of data, and generates visual summaries, statistics, and distribution plots without requiring any coding or a manual setup. This widget automatically generates the following analyses for a dataset: • Missing Value & Schema Summary – Creates a table that lists each feature or column. – Shows the number and percentage of missing values per column. – Displays the inferred data type (e.g., numeric, categorical, text). – This helps users to quickly identify columns with significant missing data or non-standard data types that may need cleaning or attention. • Numeric Feature Analysis (For Columns Detected as Numeric) – Distribution Plots: Histograms combined with a smooth density (KDE) curve display how data points are spread across ranges of values. – Box Plots: Allows quick detection of outliers, medians, and interquartile ranges. – Purpose: Enables users to see which values are common, detect skewness or unusual patterns, and spot potential anomalies. • Correlation Heatmap – Displays a color-coded matrix showing how strongly pairs of numeric features are correlated. – Helps in detecting strong associations, redundancy (collinearity), or identifying which variables might influence each other. • Categorical Feature Analysis (For any Column Detected as Categorical or Text:) – Frequency Plots: Bar charts that display how often each category appears within the column. – Purpose: Visualizes dominant and rare categories, helping to find class imbalance or need for further grouping.

3. Click the node to add the basic details.

The **Basic Details** page is displayed.

Figure 11-87 Basic Details for the Paragraph

Basic Details

Script

Activity Name
PARAGRAPH8b64bd7e-31aa-4db3-905f-d56f9a36c518

Description

Task Type
Default

Track Output
☐

```
%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 the following details:

Table 11-18 Adding Details

Field	Description
Activity Name	Enter the activity name
Description	Enter the description of the 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 as ON in case you want to execute the paragraph and view the result from the Dashboard tab.
Script	Displays 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, whose behavior is based on the execution of the user. It has a script that evaluates *True* or *False* and based on the value, it chooses either of them. Additionally, you can also add a new node that comes with a default link. The default path will be executed irrespective of the result.

- To execute a node, click the **Execute** icon. If no active or a selected session is available, then the **Session Required** dialog box appears, informing you that it cannot be executed with no selected session and you need to select a valid session or create a new one. Either click the **Create New** button to create a new session, or click the **Cancel** button, select a session and then click **Execute**.

Note

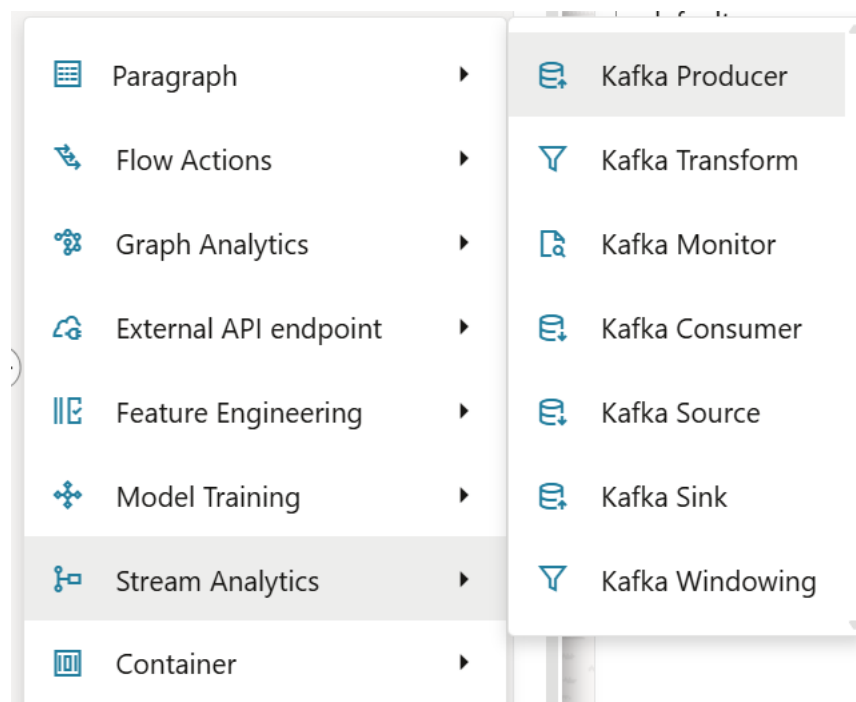
The **Session Required** dialog box will not appear if the session is active or if the session has been selected.

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 11-88 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

① 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 11-89 Kafka Source

Kafka Source

Basic Details Custom Parameters

Input Topic Required

Auto Offset Reset Required

Enable Auto Commit Required

Consumer Timeout(ms) Required

Group ID

Transformation Script

Table 11-19 Mandatory Parameters

Parameter	Description
Input Topic	Select the input topic.
Auto Offset Reset	Select either of the two options: • 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/ofsaaapp/ 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 11-90 Kafka Sink

Kafka Sink

Basic Details

Custom Parameters

Output Topic

Required

input_dataframe

Required

Table 11-20 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 11-91 Kafka Consumer

Kafka Consumer

Basic Details

Custom Parameters

Input Topic

Required

[More Details](#)

Auto Offset Reset

Required

Enable Auto Commit

Required

Consumer Timeout(ms)

Required

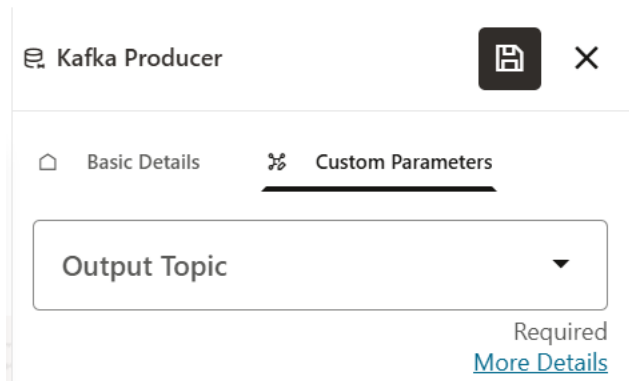
Group ID

Table 11-21 Mandatory Parameters

Parameter	Description
Input Topic	Select the input topic.
Auto Offset Reset	Select either of the two options: • 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: <code>import pandas as pd data_out= { "Msg": data_prev} pd.DataFrame(data_out).to_csv("/scratch/ofsaaapp/kafka/output/data_out.csv", sep='\t')</code>

Kafka Producer

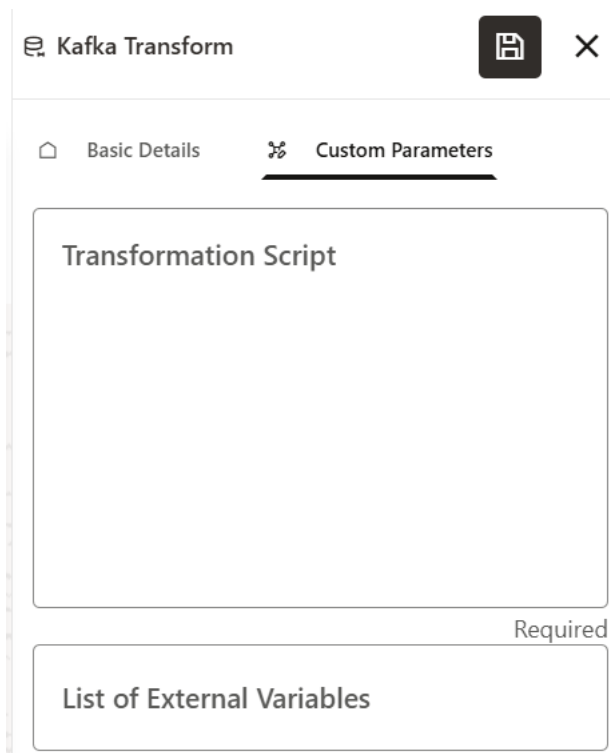
When you click the **Kafka Producer** widget, the following manadatory parameters must be provided:

Figure 11-92 Kafka Producer

- **Output Topic:** Provide the output topic.

Kafka Transform

When you click the **Kafka Transform** widget, the following mandatory parameters must be provided:

Figure 11-93 Kafka Transform

- **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 11-94 Kafka Monitor

Kafka Monitor

Basic Details Custom Parameters

Time Duration(ms) Required

Output Partition To Monitor Required

Output Variable Name Required

Table 11-22 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 11-95 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 11-23 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 11-23 (Cont.) Kafka Windowing Fields

Field	Type	Values / Rules
Note:		
•	–	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

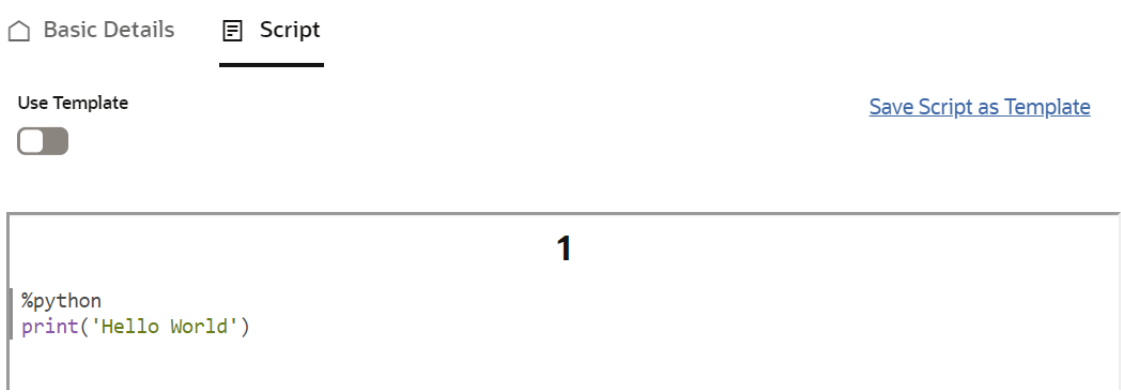
The following features will be implemented in a future release:

- – **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 11-96 Script template



To add a script template:

1. In the **Script** tab, Click **Save Script as Template** link.

Figure 11-97 Script tab

Script Name

Script name should be unique

Required

Description

Description

Script Type

Default

Script Content

%python
print('Hello World')

Cancel Save

2. Enter the following details:

Table 11-24 Creating Script Template

Field	Description
Script Name	Enter the Script Name
Description	Enter the description of Script

Table 11-24 (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.

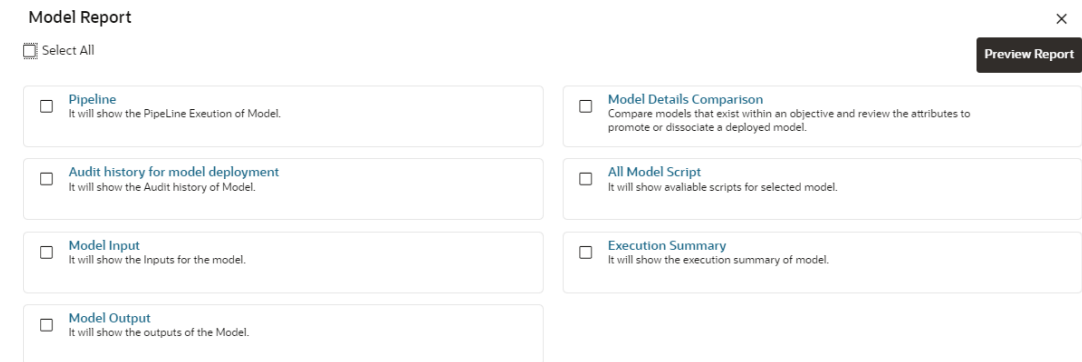
- Clone to new draft: Allows to create a new draft with the same pipeline.
- Overwrite existing draft: Allows to overwrite the existing draft with the current published model.

Model Report

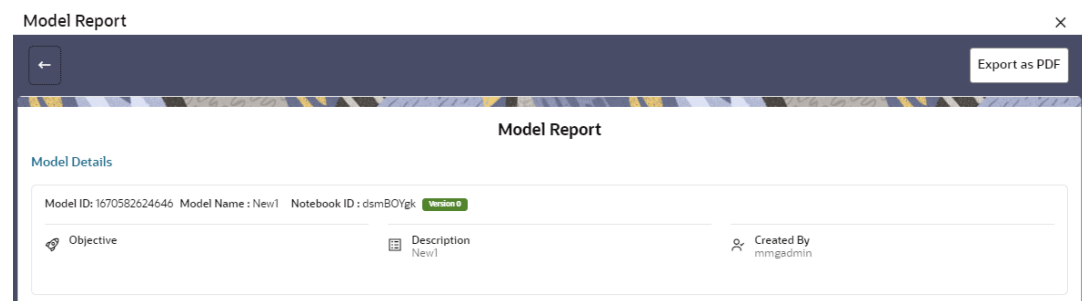
This option allows you to view the Model Report and download the same in PDF format.

To view and download report:

1. Open a Model in Pipeline Designer.
2. Click **Generate Model Report**.
The Model Report window is displayed.

Figure 11-98 Model Report window

3. Select the Parameters to generate the report.
4. Click **Preview Report**.
The report is displayed based on selected parameters.

Figure 11-99 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

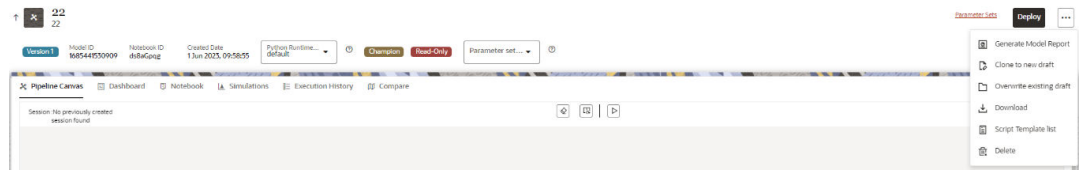
This option is used to delete the Model.

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 11-100 Published Model in Pipeline Designer

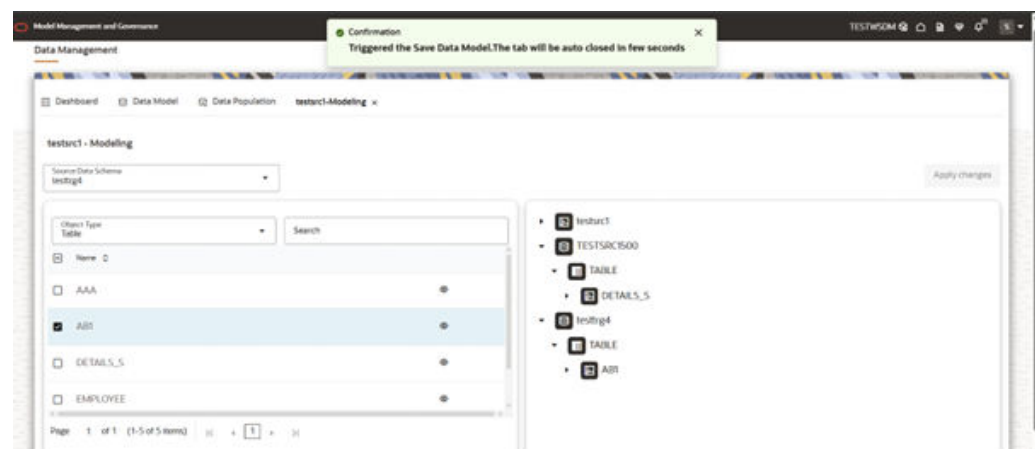
2. Click **More Actions** icon and select **Clone to new draft** or **Overwrite existing draft** option.

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. Sometime, 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 11-101 Confirmation message

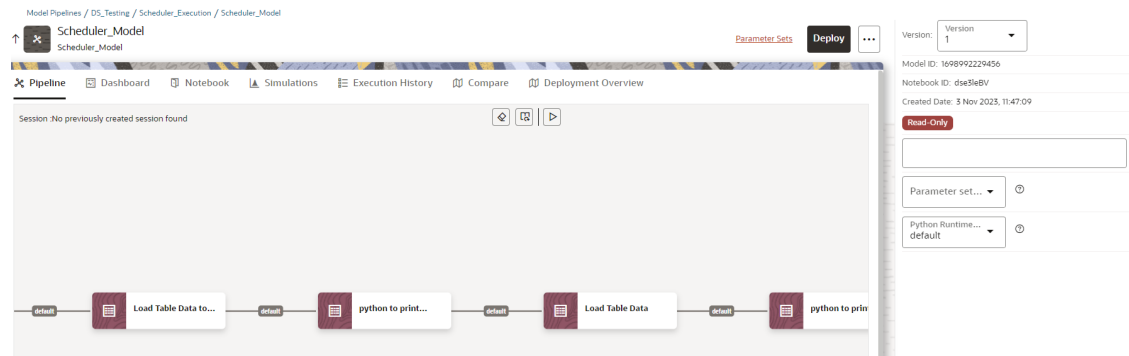
- If user clicks **Yes**, then execution thread will be closed for this given session time.
- If user clicks **No**, then execution thread will be valid for this given kill time and run in the background.

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 11-102 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 11-103 Execute Pipeline

The screenshot shows the 'Execute Pipeline' window. It includes a 'Links' section with checkboxes for 'Training', 'Experimentation', and 'Scoring'. Below this is the 'Execution Parameters' section with a table for defining parameters. The 'System Parameters' section shows pre-defined parameters like '\$FICMISDATES' and '\$BATCHRUNID\$'. The window concludes with 'Cancel' and 'Execute' buttons.

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.

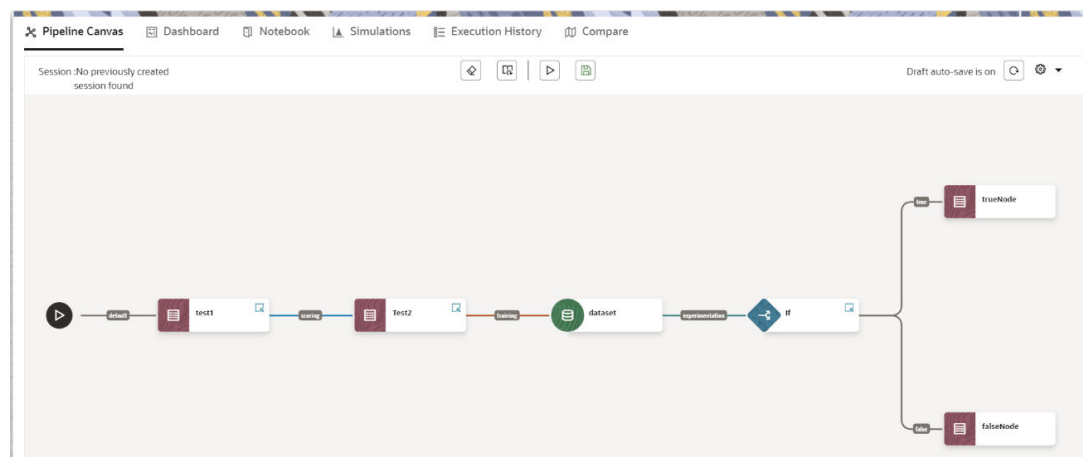
Note

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.

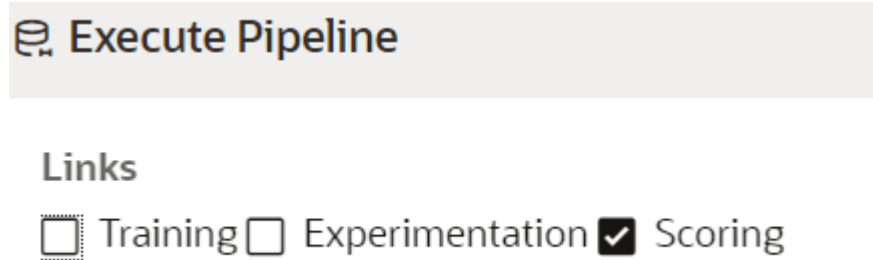
For example, refer to below figure.

Figure 11-104 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 11-105 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 11-106 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 11-25 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 11-25 (Cont.) Field and its description for the Publish pop up window

Field or Icon	Description
Attach Parameter set	If this option is enabled: - 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	If this option is enabled: - 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.

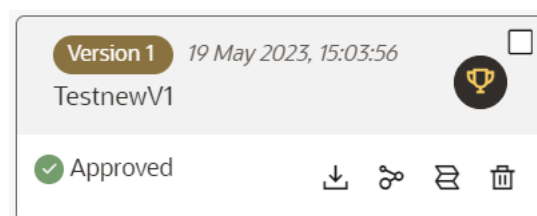
3. Click **Publish**.

View Model Details

You can view model information for deployed models, models that require approval, and so on.

To view model details:

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.

Figure 11-107 Mouse over the Model

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.

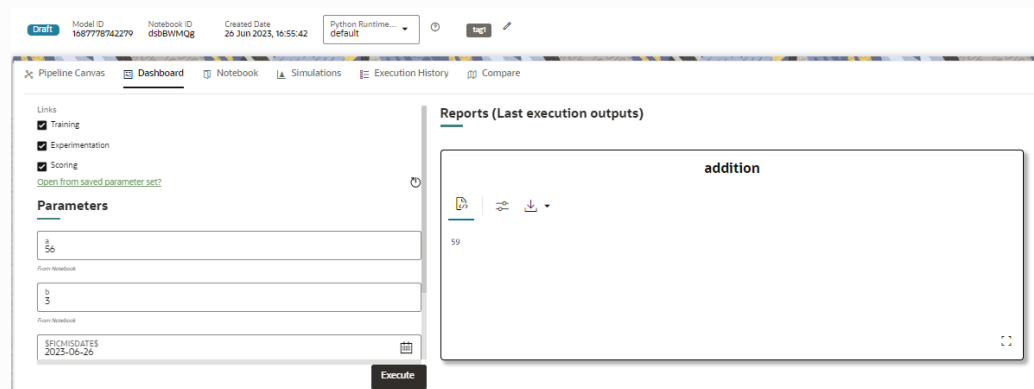
Dashboard

This section of Pipeline Designer allows you to execute the models and also displays the execution output of a model if the widgets are saved with the Track output option enabled.

Note

For *Promoted* models, the **Dashboard** is the default tab.

Figure 11-108 Dashboard



This tab contains two options for execution:

- **Batch:** This provides all the required inputs upfront and executes the pipeline end-to-end in a single run.
- **Interactive:** This executes the pipeline stage-by-stage; the run pauses at each stage to prompt the required inputs for that stage before continuing further.

Note

In an Interactive execution, when two or more paragraphs within the same stage are configured to run in parallel, the input prompt may not list the parameters for all the parallel paragraphs, for example, parameters for *Para2* may not be displayed. As a workaround, avoid parallel executions for paragraphs that require interactive inputs, or move the paragraphs to separate stages so that inputs are collected sequentially.

Note

There is no Cancel button in Settings tab of reports in the Dashboard tab. You can press 'Escape' key to close the Settings.

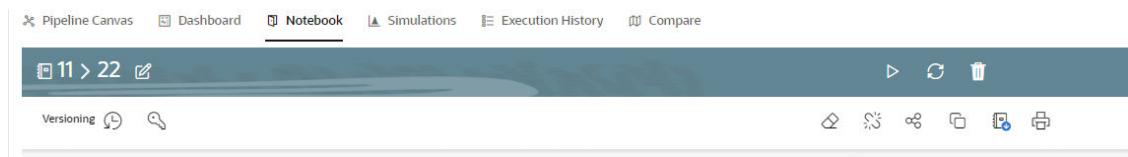
Notebook

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 11-109 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 11-26 List of APIs

Name	Description	Usage Example	Note
<code>getcwd</code>	Returns the current working directory path within MMG FTPSHARE for the active workspace.	<code>getcwd()</code>	The <code>getcwd()</code> function returns the path inside the attached workspace that is within FTPSHARE. It does not reflect the actual cwd 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.
<code>move</code>	Moves a file or directory to another location.	<code>move(src='/source/path', dst='/destination/path')</code>	
<code>copy</code>	Copies a file with metadata from one location to another.	<code>copy(src='/source/path', dst='/destination/path')</code>	
<code>copyfile</code>	Copies a file without its metadata.	<code>copyfile(src='/source/file', dst='/destination/file')</code>	
<code>copytree</code>	Recursively copies a directory tree to the destination.	<code>copytree(src='/source/dir', dst='/destination/dir', dirs_exist_ok=True)</code>	
<code>listdir</code>	Lists all files in the specified directory.	<code>listdir(path='/your/dir')</code>	
<code>remove</code>	Removes a single file from the specified path.	<code>remove(path='/file/to/remove')</code>	
<code>rename</code>	Renames a file within the current workspace.	<code>rename(src='/old/filename', dst='/new/filename')</code>	

Limitation:

- **%shell:** This is not enabled by default. Shell execution is generally considered a security risk and is disabled unless explicitly required and approved as per internal security guidelines.
- **%spark:** This is the Data Studio default configuration (spark.json) and enables both %spark and %pyspark; however, the **Add** button is enabled only for %pyspark by default. Enable %spark only if it is required by the application or customer and is approved.
- **%jdbc:** This is the Data Studio default configuration and enables the **Add** button for %mysql but not for %jdbc. Enable %jdbc only if it is required by the application or customer and is approved.

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 11-110 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 11-111 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 11-112 Run Stats

The screenshot displays the 'Run Stats' interface for two different runs. On the left, there are 'Run time parameters' including 'a', 'b', 'num12', 'num22', 'sub11', and 'sub12', each with a red 'x' icon. Below these are environment variables: '\$FICMISDATE\$', '\$BATCHRUNID\$', and '\$TASKID\$'. The main area shows two panels, each titled 'Choose from parameter sets'. Each panel contains a 'Parameter sets' section with a 'Default (From notebook)' dropdown. Below this, there are input fields for 'a', 'b', 'num12', 'num22', 'sub11', and 'sub12'. A 'Save parameter set' button is located below the input fields. At the bottom of each panel, there are fields for 'MISDATE' (2023-07-12), 'Batch Run Identifier' (Batch_simulation_ftb5fbb3-5ea6-4665-a24b-08ff9t), and 'Task Identifier' (Task_0).

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.

Executions

This section of the Pipeline Designer displays the history of the executed pipelines.

Note

The **Add File Output** button in the execution history tab opens the file manager at a specific path depending on the Save dataframe widget in the **Pipeline** window. When the Save dataframe widget is not available in the pipeline, it displays that the path does not exist from file manager.

Figure 11-113 Executions

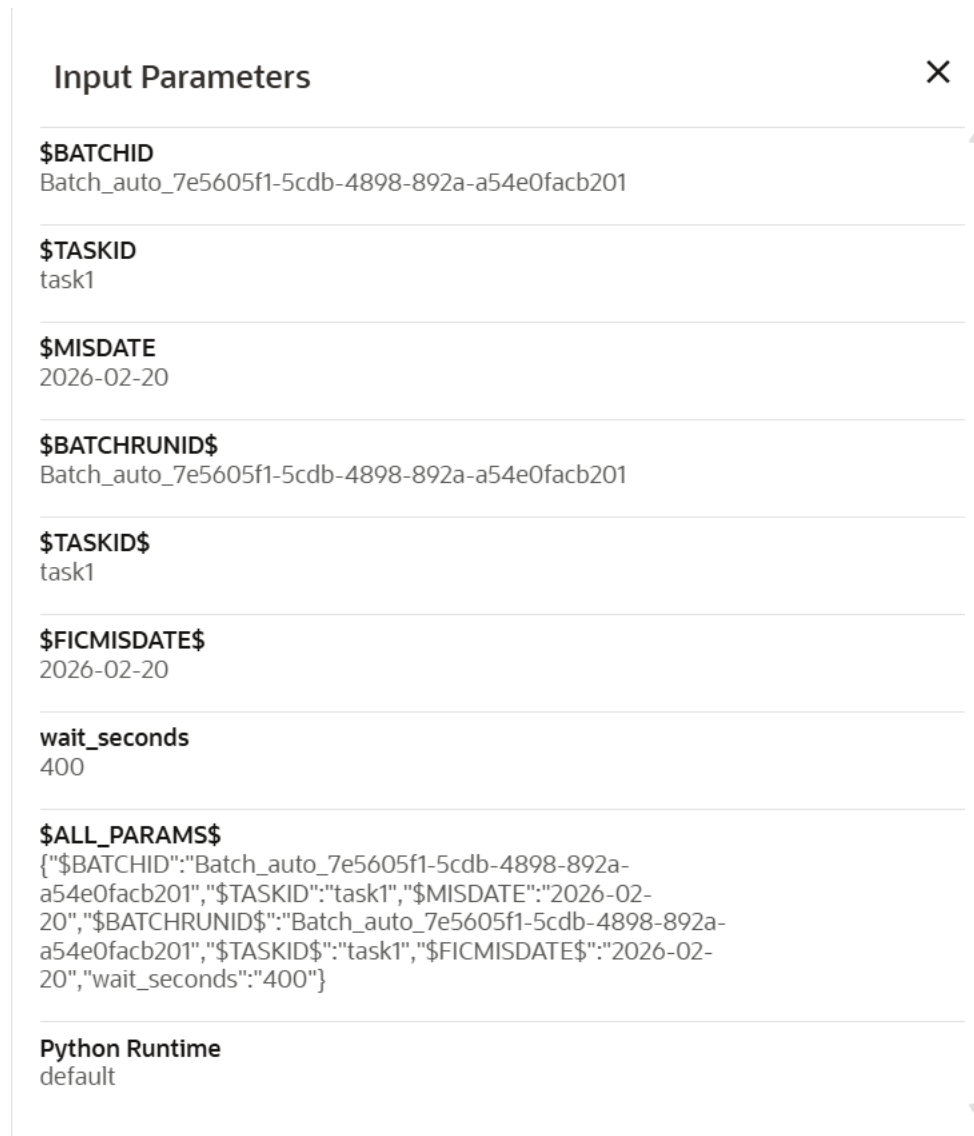
The screenshot shows the 'Executions' tab in the Pipeline Designer. At the top, there is a breadcrumb trail: 'Model Pipelines / ObjectiveTest / Model_Execution'. Below this, there is a 'Model_Execution' section with a 'Model_Execution For testing model exec' label and a 'Publish' button. The main area is a table with columns: 'Batch Run Identifier', 'Inputs', 'Task Identifier', 'User', 'Status', 'Outputs', 'Canvas view', 'MISDATE', and 'Start Time'. The table contains one row with the following data: 'Batch_auto_7e5605f1-5cdb-4898-892a-a54e0facb201', 'Custom', 'task1', 'MMGADMIN', 'In Progress', 'Outputs', 'Canvas view', '2026-02-20', and 'Feb 20, 2026, 1:18'. The table is paginated, showing 'Page 1 of 1 (1 of 1 items)' and a '10' items per page selector.

Runs Tab

You can compare and refresh the executions by clicking the **Compare** and **Refresh** icons.

Click the link in the **Inputs** column adjacent to an execution to open the Input Parameters window. This window displays the input parameters details of the execution.

Figure 11-114 Input Parameters window



The screenshot shows a window titled "Input Parameters" with a close button (X) in the top right corner. The window contains a list of parameters and their values, separated by horizontal lines. The parameters are:

- \$BATCHID**: Batch_auto_7e5605f1-5cdb-4898-892a-a54e0facb201
- \$TASKID**: task1
- \$MISDATE**: 2026-02-20
- \$BATCHRUNID\$**: Batch_auto_7e5605f1-5cdb-4898-892a-a54e0facb201
- \$TASKID\$**: task1
- \$FICMISDATE\$**: 2026-02-20
- wait_seconds**: 400
- \$ALL_PARAMS\$**: {"\$BATCHID": "Batch_auto_7e5605f1-5cdb-4898-892a-a54e0facb201", "\$TASKID": "task1", "\$MISDATE": "2026-02-20", "\$BATCHRUNID\$": "Batch_auto_7e5605f1-5cdb-4898-892a-a54e0facb201", "\$TASKID\$": "task1", "\$FICMISDATE\$": "2026-02-20", "wait_seconds": "400"}
- Python Runtime**: default

In the Outputs column, click either of the following icons:

- Show Outputs:** When you click this icon, the **Output Details** window appears, displaying the *Executed-only* paragraph outputs. By default, this option is enabled, you can disable this option based on your requirements. From the **View Type** drop-down list, you can select either *Paragraph List View* or *Notebook View*. In the **Output Visibility** section, select either **Tracked** or **All**. By default, the **Tracked** option is selected and displays the outputs tracked from the time of the execution.

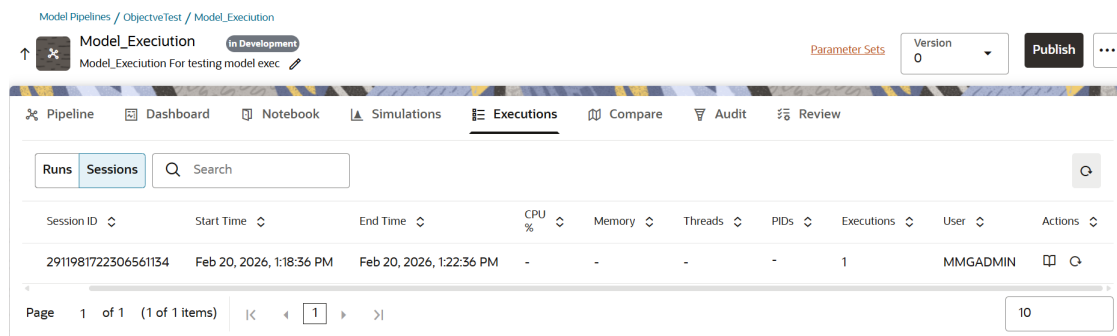
- **Output File:** When you click this option, the File Manager window appears. For more information on the File Manager feature, see the [File Manager](#) topic.

Click the **View in canvas** icon in the **Canvas View** column to view the executions via the Pipeline View window.

Sessions Tab

Click the **Sessions** tab to view the sessions information:

Figure 11-115 Sessions Tab



- **Cancel:** Click this icon to cancel a session.
- **Refresh:** Click this icon to refresh the page.
- The form displays the session information via the following columns:
 - Session ID
 - Start Time
 - End Time
 - CPU %
 - Memory
 - Threads
 - PIDs
 - Executions
 - User
 - Actions - This column contains the following icons:
 - * **Open Notebook:** When you click this icon, the **Session Notebook** window appears.
 - * **Refresh:** Click this icon to refresh the session.

Note

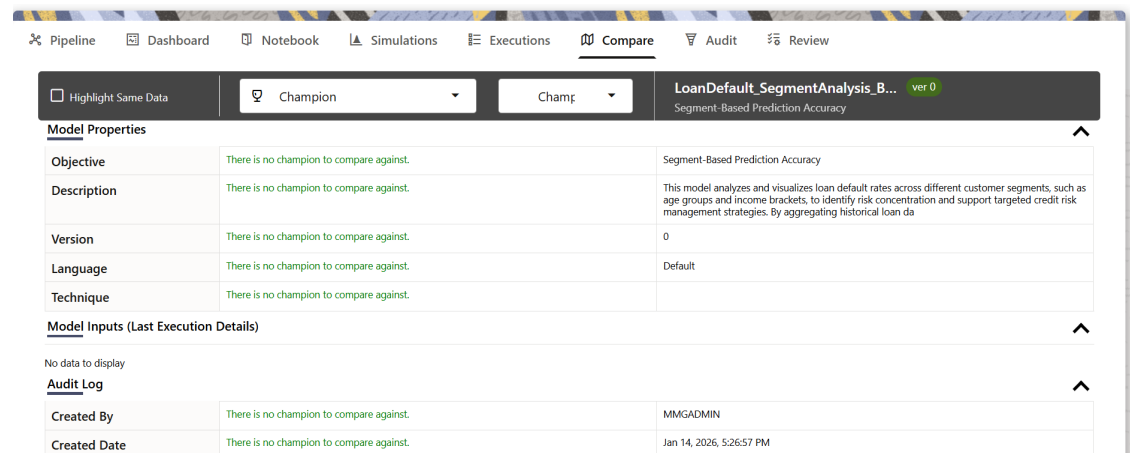
The data that appears in the CPU%, Memory, and PIDs columns are fetched via the data-studio APIs.

Compare

The **Compare** tab in the Model Pipeline Designer enables you to evaluate and contrast different model pipelines and their versions within a given objective. This feature is now more flexible as you are no longer limited to comparing only the champion, the best-performing pipeline, but can select any pipeline and version available from the current or descendant objectives.

To compare a selected pipeline against a champion version, navigate to **Pipeline Designer** window and click **Compare** tab.

Figure 11-116 The Compare Tab



This tab contains two drop-down lists:

- **Pipeline:** Select the desired available pipelines in the current objective and its descendants. The pipeline selector includes all pipelines from the current objective and its descendants
- **Version:** Select the desired version for the selected pipeline. Additional versions of the same pipeline can be selected but the champion version will not be duplicated in the selection list.

This tab also displays the following comparison details:

- Model Properties
- Model Inputs (Last Execution Details)
- Audit Log
- Model Script
- Model Output (Last execution outputs)

Note

If a model pipeline is newly created and only its initial version, *ver 0* or *Draft 0*, exists, with no champion or other versions, the **Compare** tab remains in an inactive, informational state. The **model/version** dropdowns may appear empty or disabled and *Draft 0* may not be selectable for comparison because there is no alternative version. In the comparison grid, all property rows display the message: *There is no champion to compare against*. To perform a comparison, create another version and/or designate a champion.

The following image illustrates an example of how this feature works when you select a pipeline and a champion version to compare:

Figure 11-117 Comparing a Selected Pipeline against a Champion Version

✕ Pipeline Dashboard Notebook Simulations Executions Compare Audit Review		
☐ Highlight Same Data	Analysis_Banking2024_v1	v1
		LoanDefault_SegmentAnalysis_Banking2024_v1 ver 0
Model Properties		
Objective	Segment Based Prediction Accuracy	Segment Based Prediction Accuracy
Description	This model analyzes and visualizes loan default rates across different customer segments, such as age groups and income brackets, to identify risk concentration and support targeted credit risk management strategies. By aggregating historical loan data	This model analyzes and visualizes loan default rates across different customer segments, such as age groups and income brackets, to identify risk concentration and support targeted credit risk management strategies. By aggregating historical loan da
Version	1	0
Language	Default	Default
Technique		
Model Inputs (Last Execution Details)		
Status of execution	COMPLETED	
Start time	Feb 2, 2026, 8:05:48 PM	
End time	Feb 2, 2026, 8:05:58 PM	
Python Runtime	environment_py310	
\$FICMISDATES	2026-01-30	
\$TASKIDS	task5	
\$TASKID	task5	
\$ALL_PARAMS	{"\$BATCHID":"Batch_Model1_2026-01-30_1769776477098_4","\$TASKID":"task5","\$SMISDATE":"2026-01-30","\$BATCHRUNID":"Batch_Model1_2026-01-30_1769776477098_4","\$TASKID":"task5","\$FICMISDATES":"2026-01-30"}	
\$MISDATE	2026-01-30	
\$BATCHID	Batch_Model1_2026-01-30_1769776477098_4	
\$BATCHRUNIDS	Batch_Model1_2026-01-30_1769776477098_4	
Audit Log		

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.

To import the parameter sets:

- Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:

```
./model_pipeline_utility.sh PARAM IMPORT TARGET_WORKSPACE_NAME LOGIN_USER  
LOCALE SOURCE_ZIP_FILE_NAME OBJECTIVE_ID OVERWRITE_FLAG[true|false] [v]
```

Table 11-27 Fields and icons on the Sandbox Summary page

Field	Description
PARAM	This is used for referring to the parameter.
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

Table 11-27 (Cont.) Fields and icons on the Sandbox Summary page

Field	Description
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:
`./model_pipeline_utility.sh PARAM EXPORT WORKSPACE_NAME LOGIN_USER LOCALE
PARAMETER_SET_NAME OBJECTIVE_ID [v]`

Table 11-28 Fields and icons on the Parameter Set page

Field	Description
PARAM	This is used for referring to the parameter.
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 Model Draft

After creating the Draft Models, publish the Notebooks, which have the Model script.

You can publish a model draft to Data Studio from the Pipelines flow. Publishing can be initiated from either of the following locations:

- From the pipeline or draft screen.
- From the **Model Summary** screen by using the **Actions** icon (three dots)
- From **Model Summary** (Actions menu)

To publish from the **Model Summary** screen:

1. Navigate to the **Model Summary** screen.
2. Click the **Actions** icon (three dots).
3. Select **Publish Model Draft**.

Note

(Limitation): When you are publishing from the **Model Summary** screen, the link information (for example, training and scoring links) is not retained because the **Publish** action is executed from the summarized content (selected paragraphs). As a result, the training or scoring links are defaulted to *default* in the published output. To preserve the configured links, publish the model draft from the pipeline or draft screen.

To publish:

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 11-118 Publish Model

Publish Model

Model Name

Testnew

Technique

Description

Run Version

Script

☐	ID ↕	Title ↕	Script ↕
☐	ds5a1kd3	Start widget	## Do not delete this paragraph. Also, ensure that this is the first paragraph in the notebook. You may use 'Move Up/ Move Down' from Setting

Publish

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

Table 11-29 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.
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

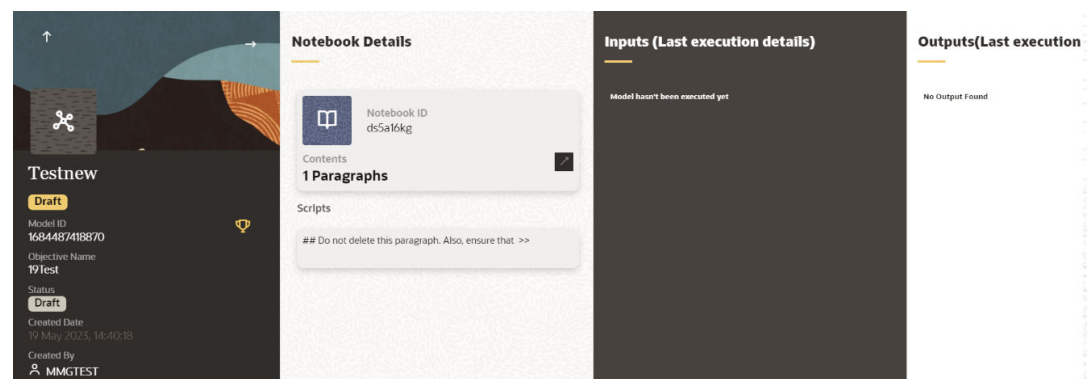
To view the scope details of the model:

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 11-119 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 `NEXTGENMF_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

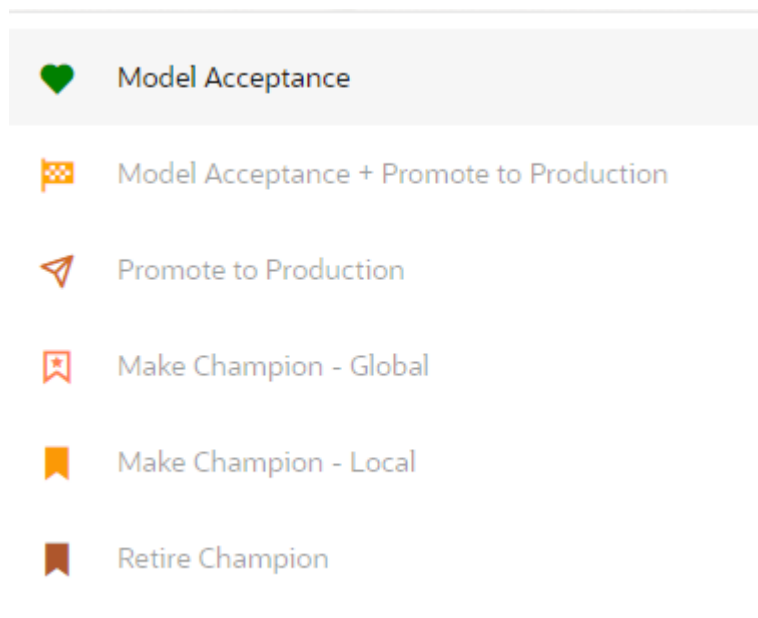
You must restart the application everytime you update this value.

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 11-120 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:

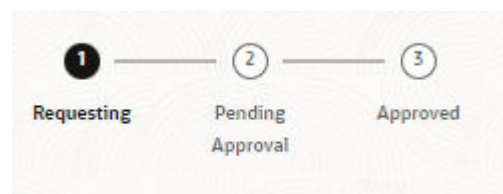
1. Click **Model Pipeline** to display the **Model Pipeline** window.
The window displays folders that contain models and model records in a table.
2. Click on the Deploy icon on the Model to open it.
3. If a model is a champion, the icon is displayed on it.
4. To make a model a champion from this window, 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 11-121 Model Deployment

The screenshot shows the 'Model Deployment' window. On the left, model details are listed: ID I685441530909, Version 2, Objective 11, Description 22, and Created By MMGADMIN on 7 Aug 2023, 18:22:12. A champion icon is visible. On the right, a progress indicator shows three steps: 1. Requesting (active), 2. Pending Approval, and 3. Approved. Below this, there are input fields for 'Reviewer' and 'Approver', both marked as 'Required', and a 'Comments' field with an 'Attach' icon.

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 11-122 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.

① 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.

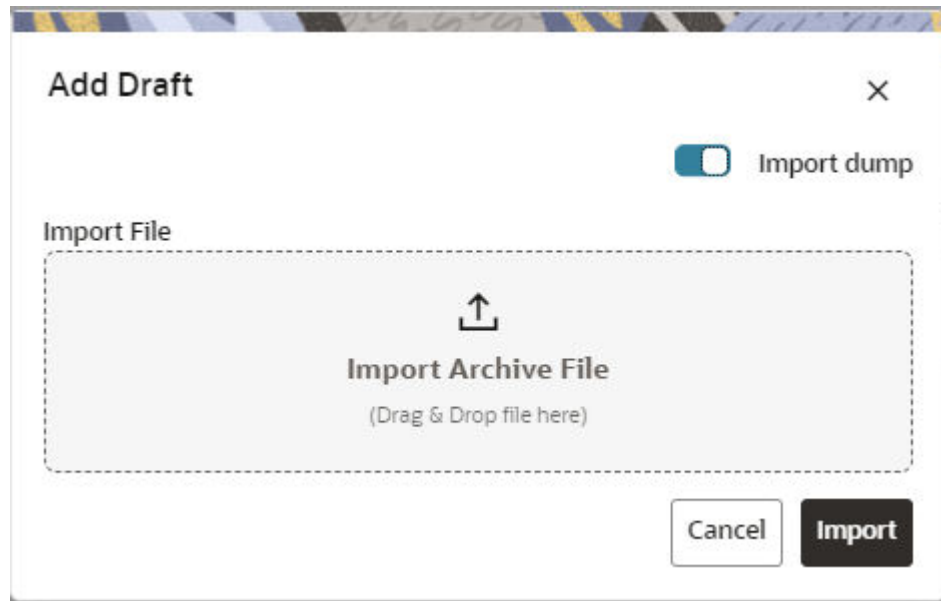
The import of model data lets you reuse and extend on model creation. This topic is part of the procedure of creating Draft Models and after creating a new model using this method, see the [Create Draft Models](#) section for instructions on how to proceed further.

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 11-123 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.

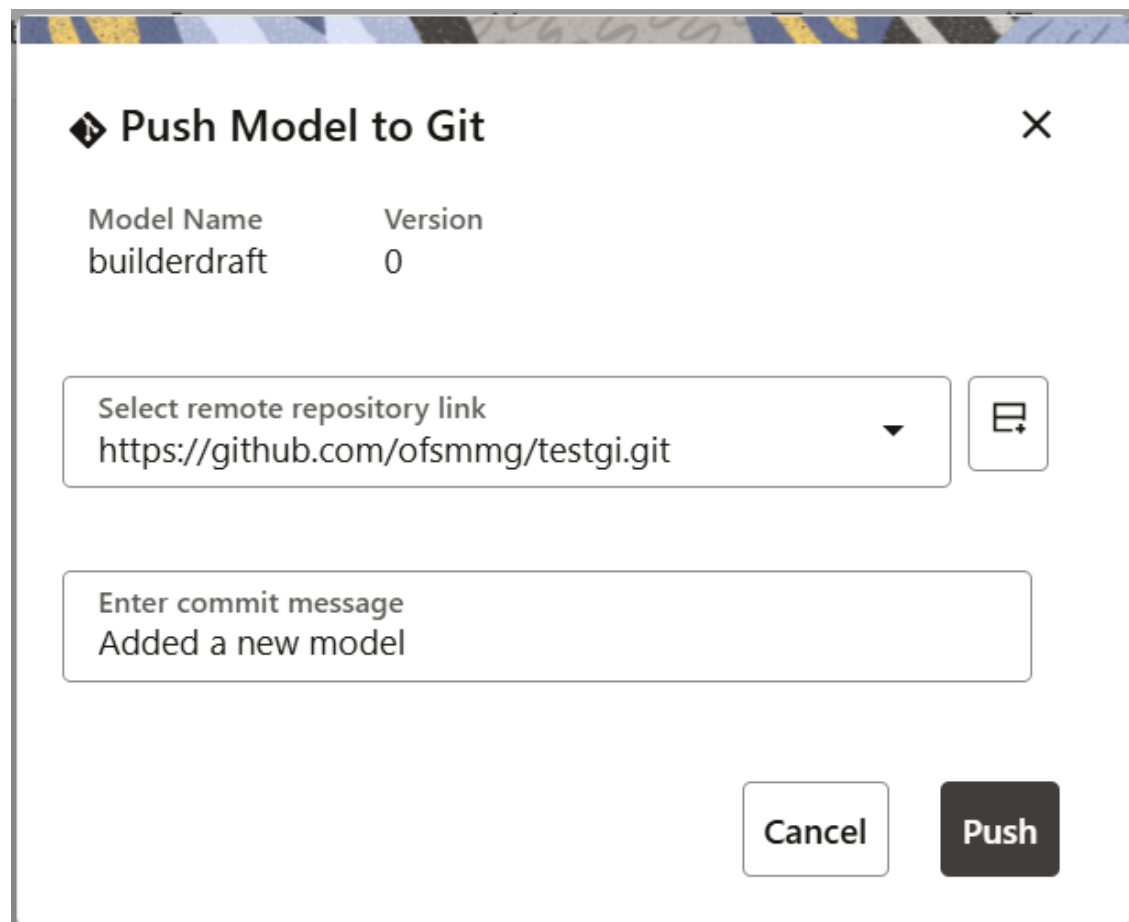
Git Functionalities

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 11-124 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 11-125 Git Pull

Pull Model From Git X

Model Name: devTest0 Version: 0

Select remote repository link: <https://github.com/ofsmmg/testgi.git> [Copy]

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 11-126 Git Clone

Clone Model From Git X

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 11-127 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 11-128 Task Parameters

Task Parameters 				
Parameter	\$RUNSKY\$	Value	RUNSKY	
Parameter	\$BATCHDATE\$	Value	Batch Date	
Parameter	\$BATCHRUNID\$	Value	BATCHRUNID	

Note

Fields marked with * are mandatory fields.

The Scheduler Service feature provides a system-generated runtime parameter **RUNSKY**, \$RUNSK\$, to support the delta processing of the Data Pipeline and Match Rules. **RUNSKY** is generated during execution and is propagated from the batch to downstream tasks or components so they can process specific execution context. You will see \$RUNSK\$ listed along with other runtime parameters such as; **RUNSKY** and is not user-editable. This is populated by the scheduler service at runtime.

RUNSKY lifecycle follows the BatchRunID behavior.

- **Rerun:** \$RUNSK\$ changes. A new **RUNSKY** is generated for the rerun execution.
- **Restart:** \$RUNSK\$ remains the same. The restart continues the same execution context.
- **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 11-129 Populate Workspace screen

Parameter	Value	
\$BATCHDATES\$	Batch Date	☒
\$BATCHRUNID\$	BATCHRUNID	☒
Global Filter		☒
Table Filters		☒
Additional Parameters		☒

Header Parameters

Parameter	Value	
user	aaaiuser	☒

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".

Data Pipeline Execution

For standalone Data Pipeline definitions, you must pass the following parameters as optional during execution from the scheduler service, after selecting the graph data pipeline as component.

- \$<Runtime Parameter1>=<Value>
- \$<Runtime Parameter2>=<value>
- \$RUNTYPE\$=PROD

- \$batchRunType\$=run
- \$BATCHTYPE\$=DATA
- \$JOBNAME\$=<DataPipeline_Name>

Export/Import/Retire Models via Utility

You can import and export the models between Sandbox to Production or vice versa using the Utility.

To import the models:

- Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:

```
./model_pipeline_utility.sh IMPORT TARGET_WORKSPACE LOGIN_USER LOCALE
FILE_NAME OBJECTIVE_ID OVERWRITE_FLAG
```

Table 11-30 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
Objective_ID	The ID of the objective.

Note

For the OVERWRITE_FLAG, you enter the 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_pipeline_utility.sh MODEL EXPORT WORKSPACE_NAME LOGIN_USER LOCALE
MODEL_ID MODEL_VERSION [v]
```

Table 11-31 Fields and icons on the Sandbox Summary page

Field	Description
Export	Command to export the Model from the Workspace

Table 11-31 (Cont.) Fields and icons on the Sandbox Summary page

Field	Description
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.

Before you retire, ensure the Models are in the published state.

1. Navigate to <MMG_PACK>OFS_MMG/bin and execute in the following format:
`./model_pipeline_utility.sh MODEL RETIRE WORKSPACE_NAME LOGIN_USER LOCALE
MODEL_ID MODEL_VERSION [v]`

Table 11-32 Fields and icons on the Sandbox Summary page

Field	Description
Retire	Command to retire the Model.
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

The migration utility allows administrators to import objects into any target workspace, including a Production workspace. If objects are imported directly into the Production workspace, then a few objects may appear in a non-final state (for example, *Pending Approval* or *Invalid*) until the required approvals or promotions are completed. Hence, you must plan the imports accordingly and then validate the object status after importing.

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

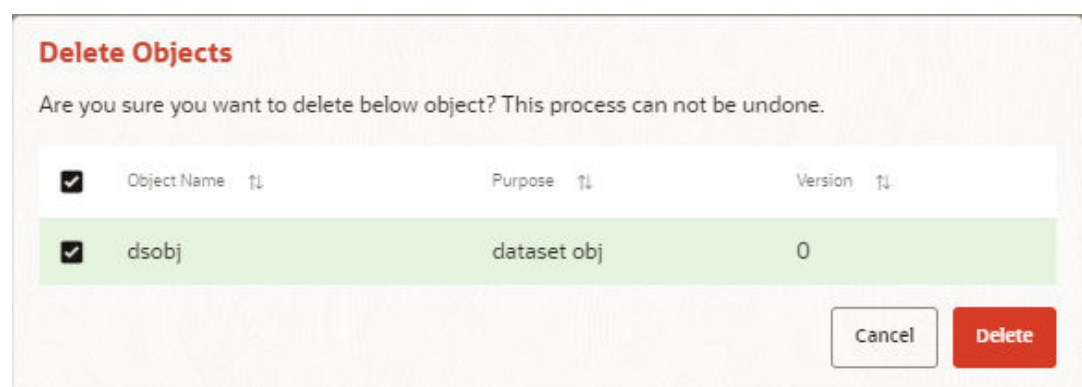
To delete Objectives and Models that exist in the Objectives, follow these steps:

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 11-130 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:

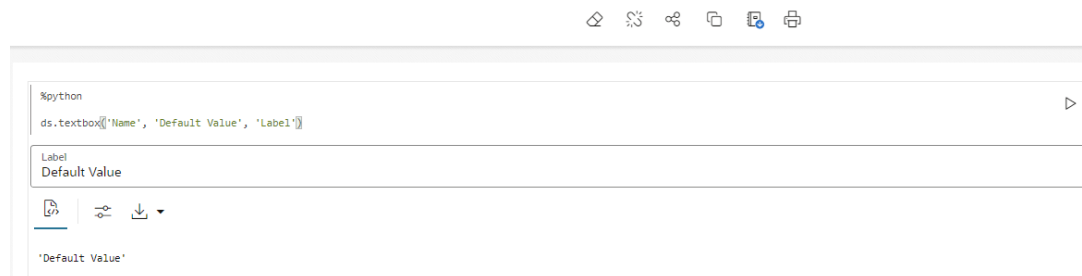
- Click **Add Python Paragraph** in Model Studio to add a Paragraph in the Notebook and fetch the runtime parameters supported in Model Studio runtime as shown in the following example. This is available for all the interpreters.

For example, enter as follows:

```
%python
ds.textbox('Name', 'Default Value', 'Label')
```

Click **Play** to run the script and display.

Figure 11-131 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 11-132 Model Pipeline Session screen

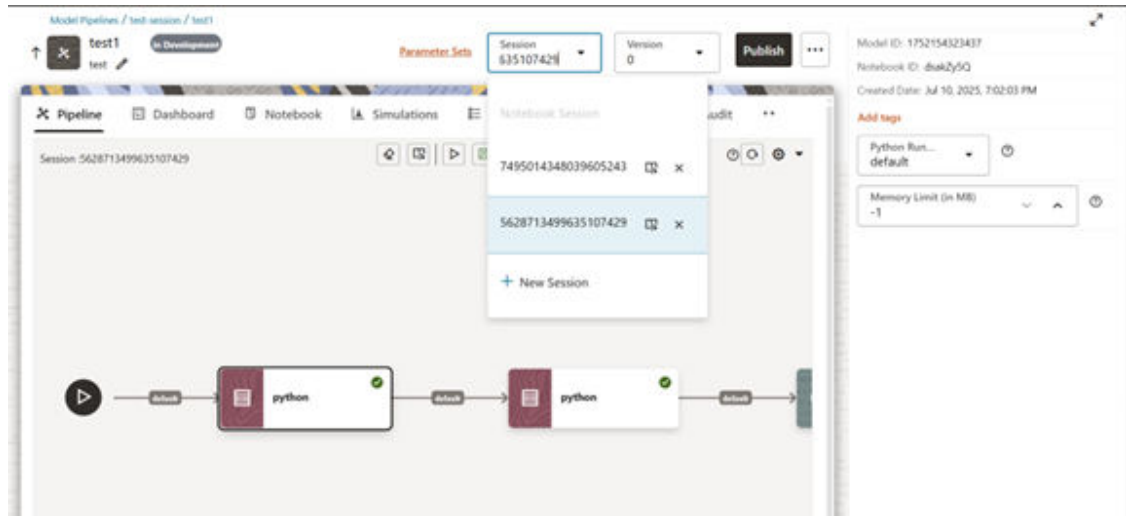
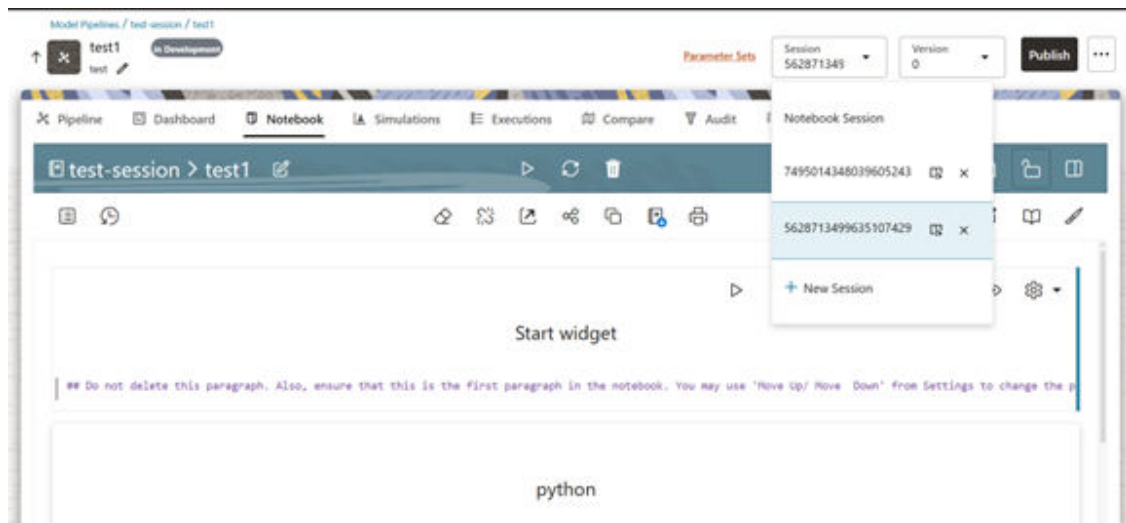


Figure 11-133 Session drop-down screen with multiple sessions shared between Pipeline Canvas and Notebook



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 11-134 Add session

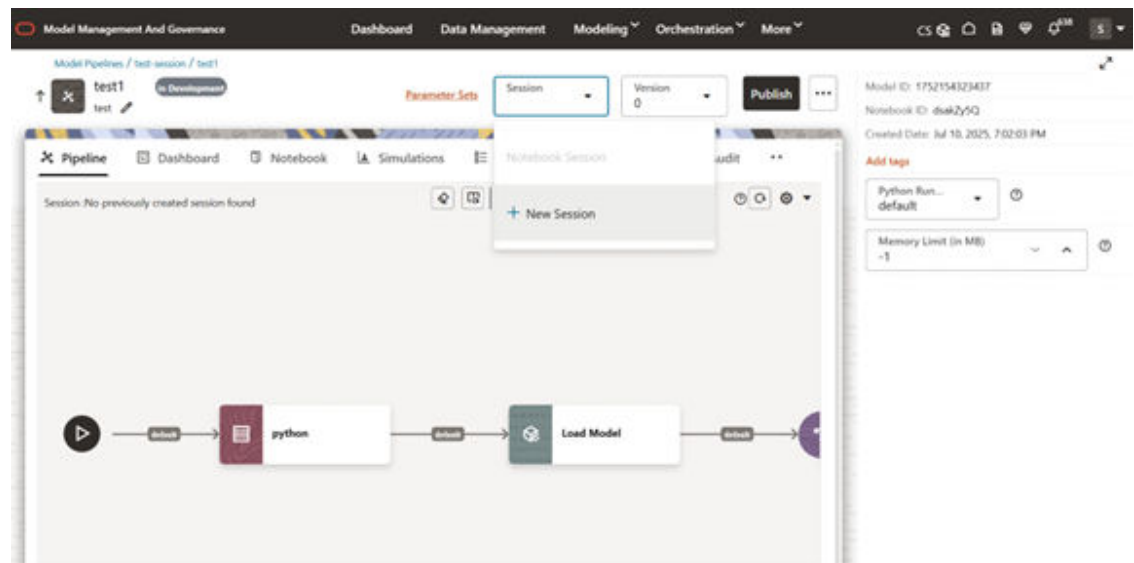
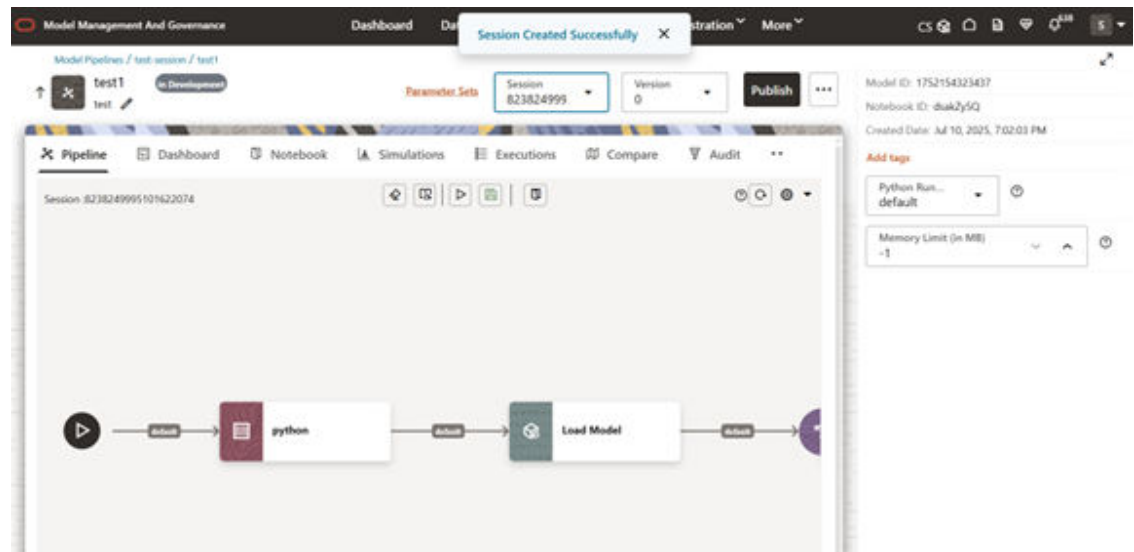


Figure 11-135 Add session created



Clearing a Session

Clearing a Session In the Session drop-down, click the cross icon beside the session to delete it.

Figure 11-136 Clearing a Session screen

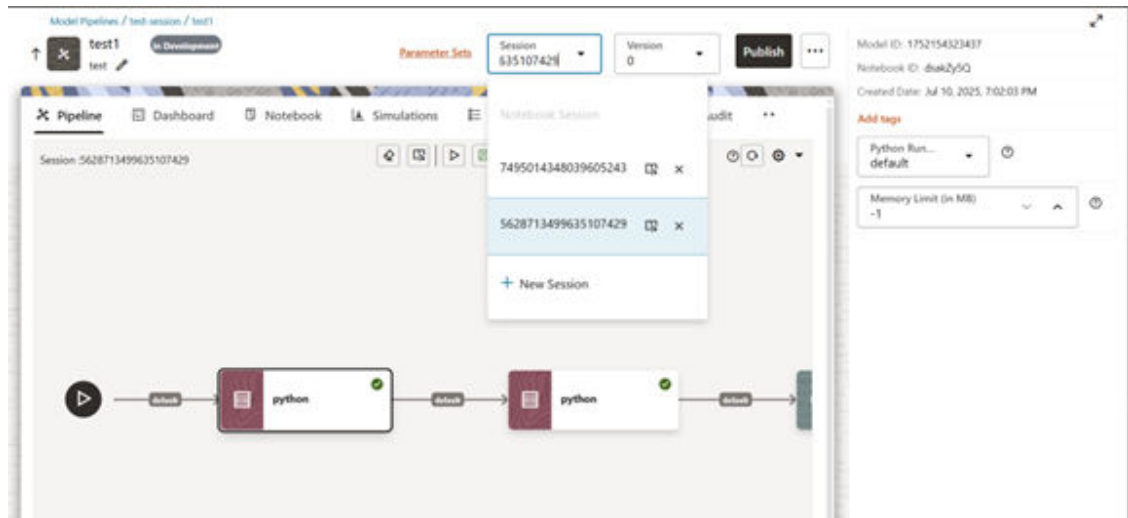
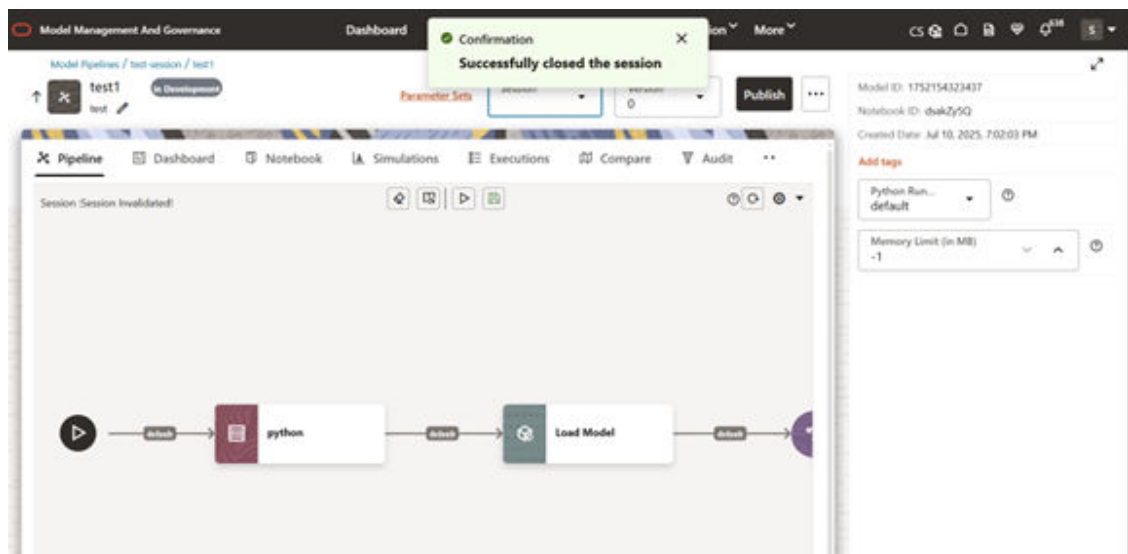


Figure 11-137 Closing the Session screen



Reinitializing a Session

Click the re-initialize icon next to a session in the drop-down to reload its state.

Figure 11-138 Reinitializing a Session

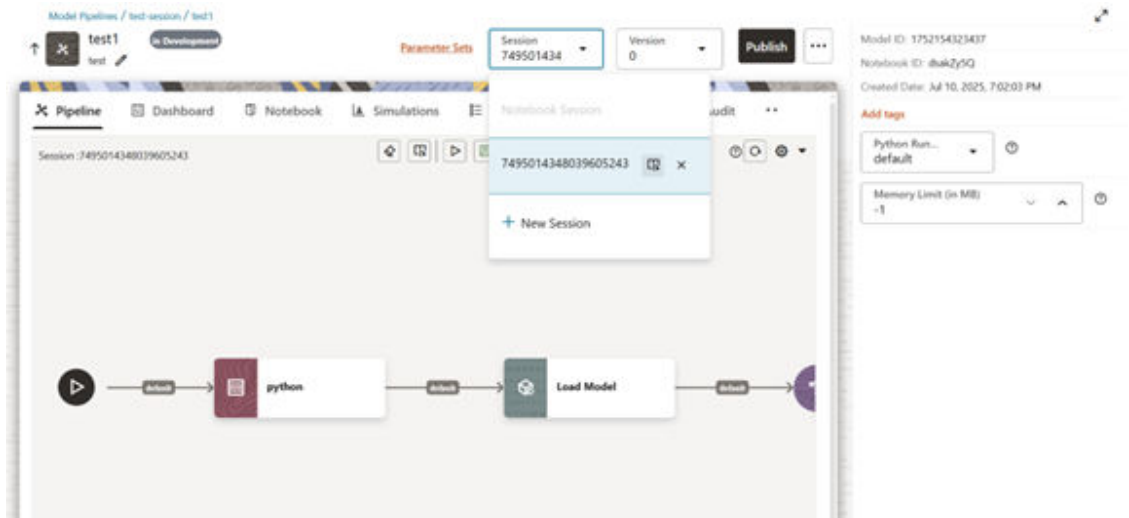
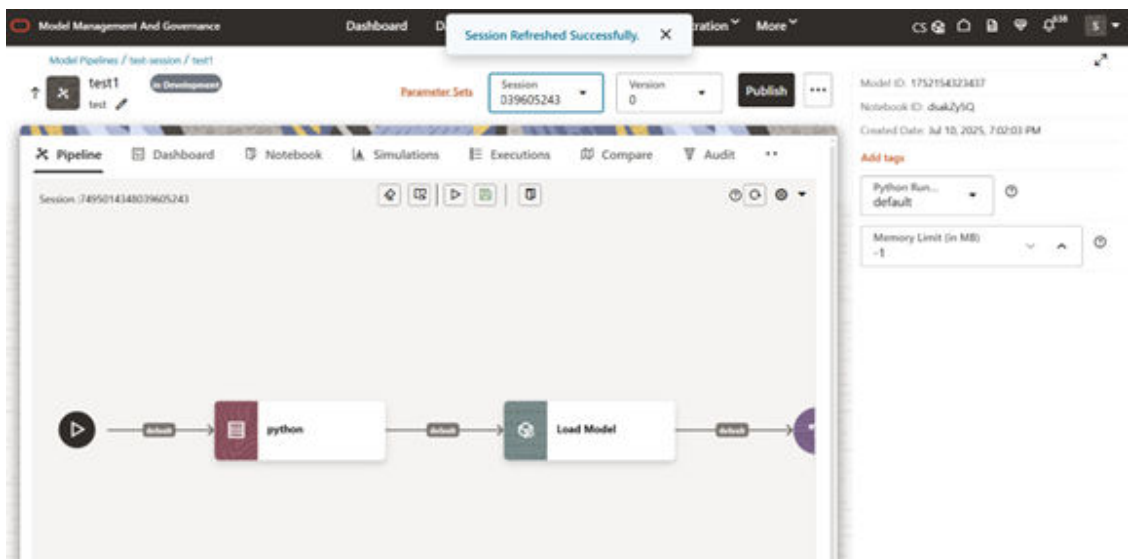


Figure 11-139 Refreshing a Session



Switching Between Notebook and Pipeline

Once a session is selected, you can move between Notebook and Pipeline tabs while keeping outputs in sync.

Figure 11-140 Switching between Notebook and Pipeline

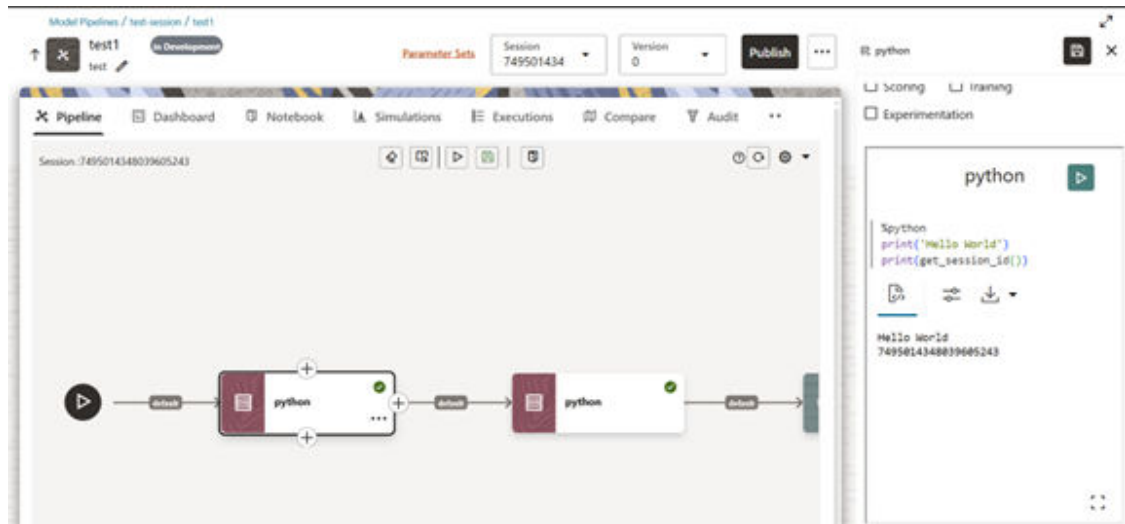
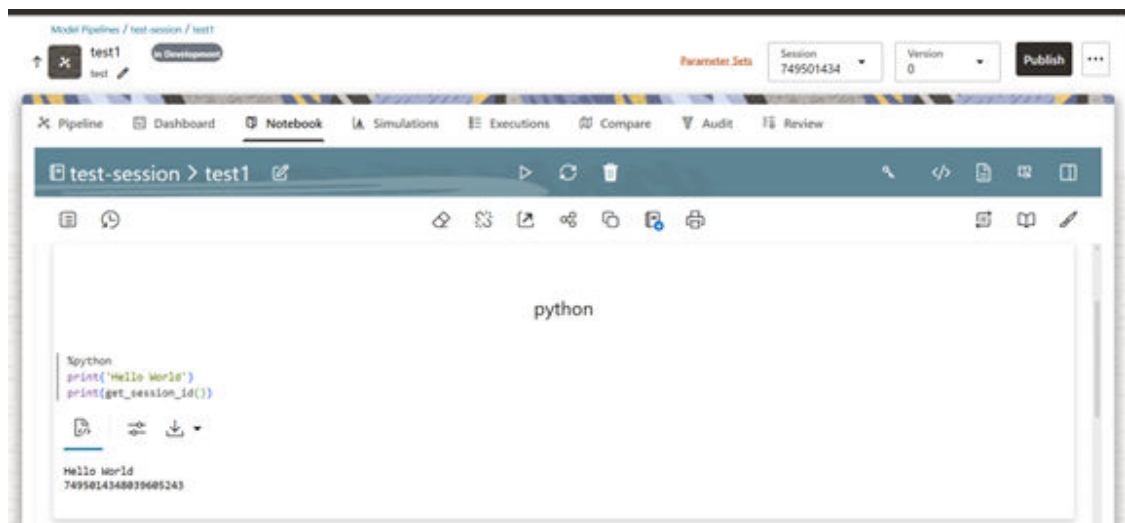


Figure 11-141 Notebook screen



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.

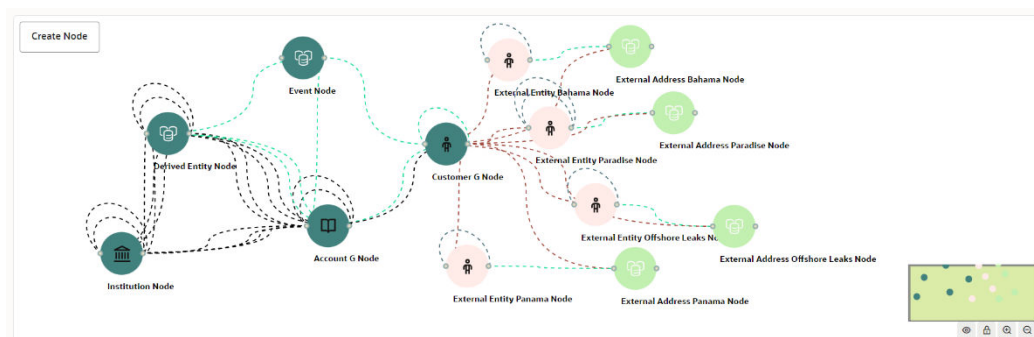
The Graph Pipeline functionality allows users to quickly create and configure a graph for use in advanced analytics. It can also manage and schedule the tasks required to run populate the graph on a periodic basis.

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 11-142 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 11-33 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:

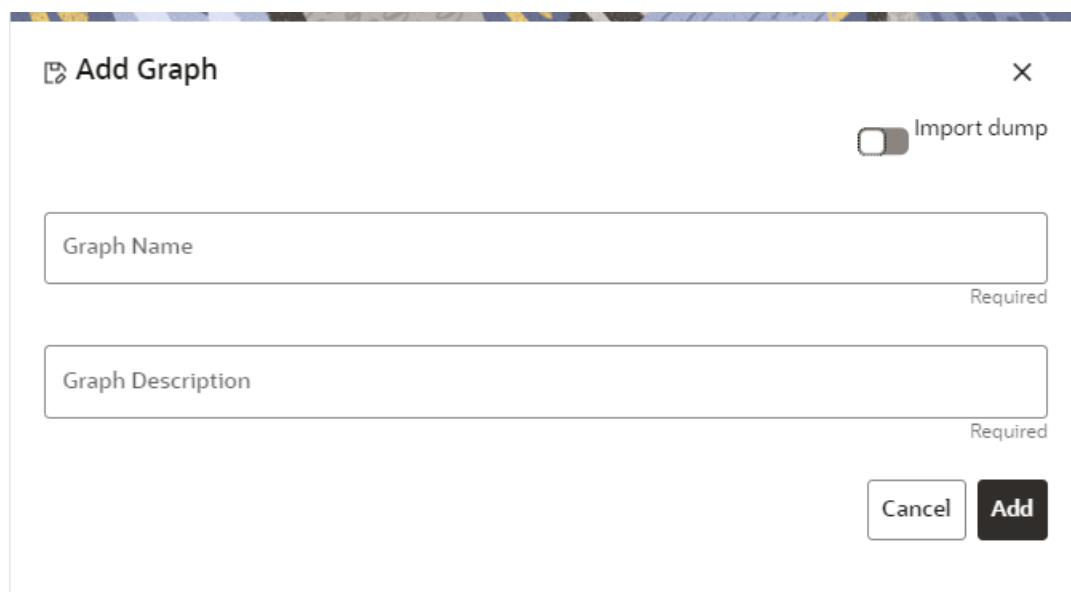
1. Navigate to Graph Summary page.

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 11-143 Add Graph window



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 11-144 Node Details

Node Details

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)

Advanced Features

Manage Pipeline(s)

Search

Create
Attach

Pipeline Name
Action

No data to display.

Page 1 (0 of 0 items)

- In the **Logical Name** field, enter a name for the Node or click **Setting** .
The Setting pop-up window is displayed.

Figure 11-145 Setting

Setting
Select Icon
Select Color

Cancel OK

- 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 11-146 Add Node Modeler Attribute(s) screen

Add Node Modeler Attribute(s)

Logical Name

Required

Data Type

Required

Maximum Length of Text

Required

Cancel OK

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 11-147 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 11-148 Advanced Features

Node Details

> Basic Information

☒ Advanced Features

Column Identifier

▼

Required

> Manage Pipeline(s)

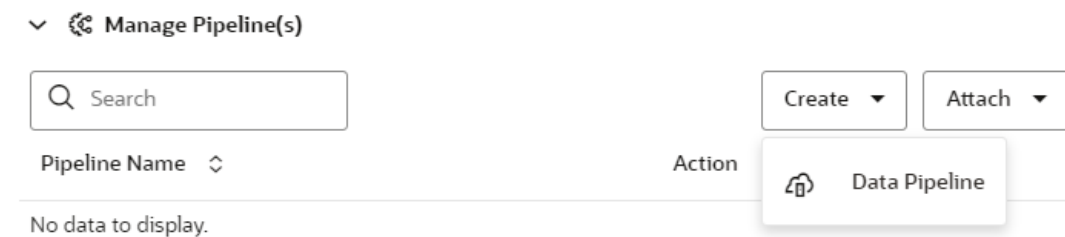
Under the **Manage Pipeline(s)** group, you can either create a new data pipeline or attach the existing pipeline.

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

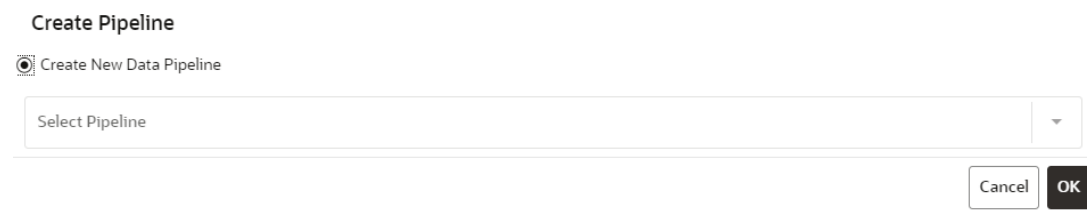
The newly created data pipeline is not automatically added to the Node. You must attach the pipeline using the **Attach** drop-down option.

The selected data pipeline is displayed under the Manage Pipeline(s).

Figure 11-149 Manage Pipeline(s)

10. Click **Create** drop-down and select **Data Pipeline**.

The **Create Pipeline** window is displayed.

Figure 11-150 Create Pipeline

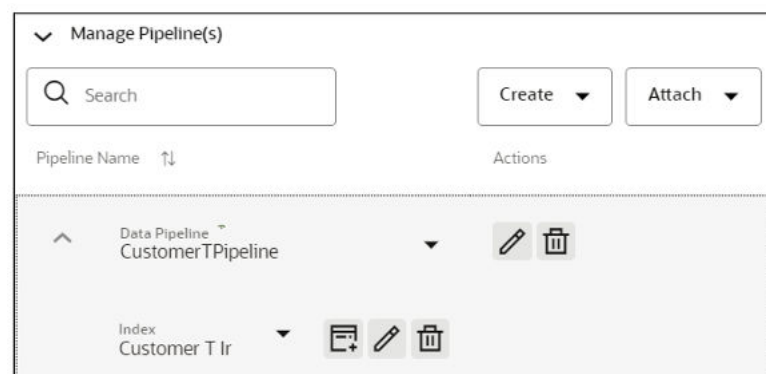
11. Select the **Create New Data Pipeline** option and then click **Ok**.

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

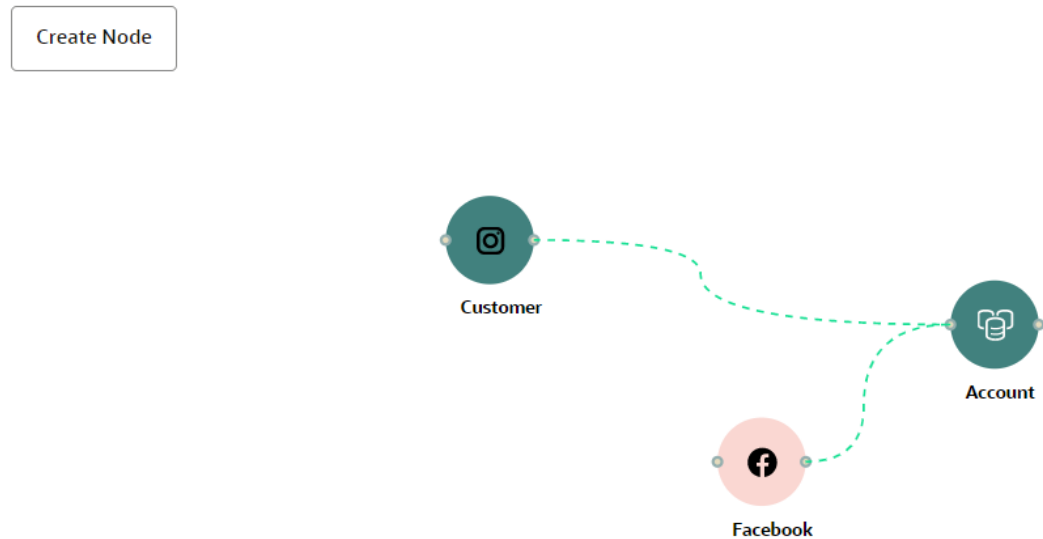
The Persist of the Data pipeline of the corresponding node should be defined with the prescript. See the Prescript Condition section for more details.

Figure 11-151 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.

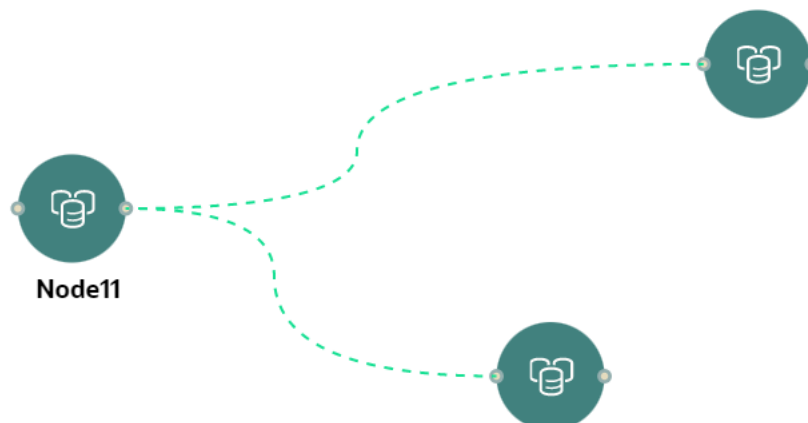
Figure 11-152 Graph with Node and Edge



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 11-153 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 11-154 Edge Details

Edge Details [X]

▼ **Basic Information**

Logical Name Required

Logical Description Required

Edge Modeler Attributes Section Is Similarity Edge ☐

Logical Name	Data Type	Format	Action
No data to display.			

(0 of 0 items) |< < 1 > >|

> **Advanced Features**

▼ **Manage Pipeline(s)**

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 11-155 Setting

Setting Select Edge Select Color [X]

— - - - -

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 11-156 Add Edge Modeler Attribute(s) screen

Add Edge Modeler Attribute(s)

Logical Name

Required

Data Type

Required

Maximum Length of Text

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 11-157 Search

Q Search

+

Logical Name

↕

Data Type

↕

Format

↕

Action

Test

Text

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 11-158 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.

9. 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.

10. Enter the retention count and select the retention duration from the drop-down.

The available retention durations are Days, Weeks, and Months.

Figure 11-159 Retention Durations tab

Retention Policy ?

Retention Count
5

Retention Duration
Days

11. 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 11-160 Pluggable Attribute

Pluggable Attribute

Is Pluggable ☒

Pluggable Attribute Required

12. 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 11-161 Key Attributes

Key Attributes

Column Identifier Required Start Identifier Required End Identifier Required

13. 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 11-162 Manage Pipeline(s) for Edge

Manage Pipeline(s)

Search

Pipeline Name ⌵ Action

No data to display.

Create ⌵ Attach ⌵

Data Pipeline

Match Rules

14. Click **Create** drop-down and then select the Data Pipeline.

The **Create Pipeline** window is displayed.

Figure 11-163 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 11-164 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 11-165 Create Pipeline

▼ Manage Pipeline(s)

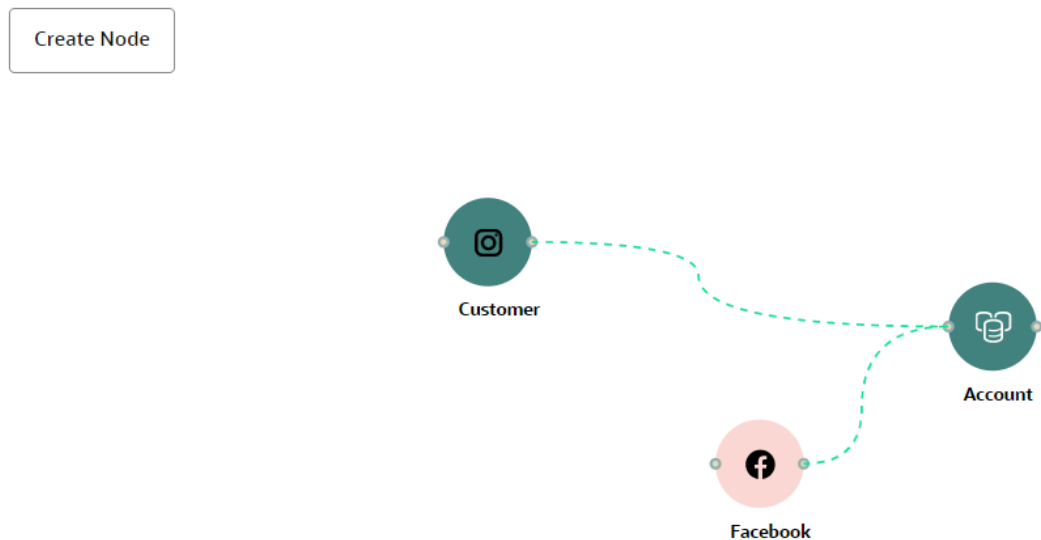
Create ▼

Attach ▼

Pipeline Name	↕	Actions
Data Pipeline [*]	▼	
Account Trail 1		

18. Click OK.

An Edge is created between the Customer, Facebook, and Account as shown in the below example.

Figure 11-166 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 11-167 Graph page

Graph Details [X]

Graph Name
Graph001

Graph Description
Graph001 Test Delta

Should batch be created in read-only mode for scheduler screen ? ☒

Max age of old session ⓘ

Retention Count
7

Retention Duration
Days ▼

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.

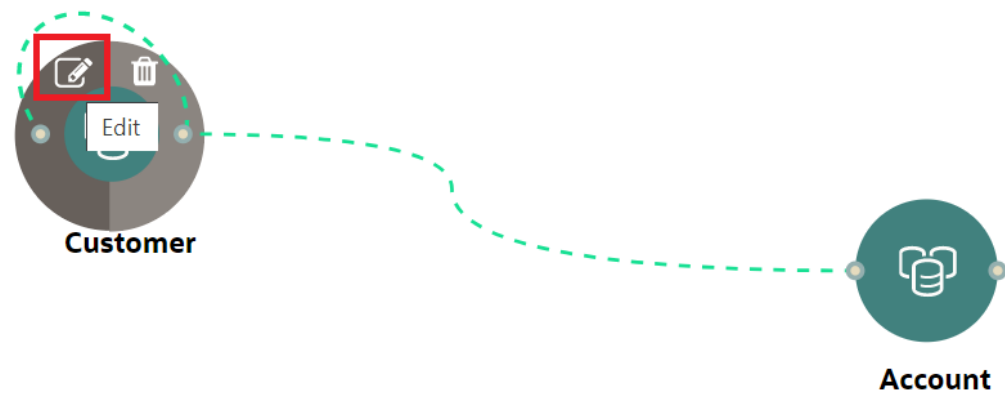
Edit a Node

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 11-168 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:

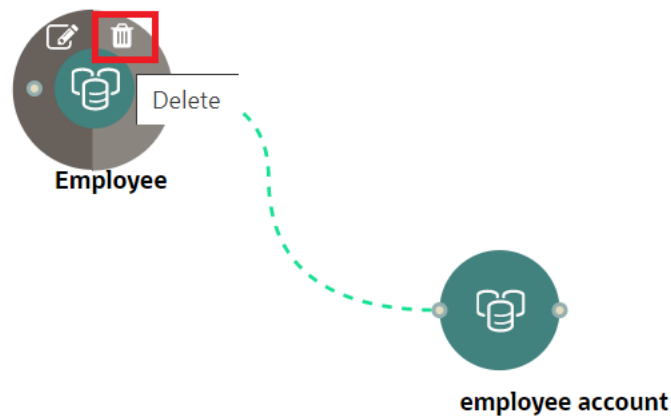
If the edge has two match rules and one is attached, and another match rule is not attached to the edge details. You need to remove attributes in both math rules for this case. To remove the deleted attribute from the Node Details page in the unattached match rule, perform the following:

- 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 11-169 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 11-170 Manage Schedule screen

Manage Schedule [Close]

Basic Information

Schedule Name Required

Schedule Type

☒ Once ☐ Daily ☐ Weekly ☐ Monthly ☐ Cron

Start Date
02/26/2023 [Calendar Icon]

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 11-171 Add Optional Parameters

Add Schedule
Cancel
Previous
Add

Selected Task(s)
TestCustomer

Optional Parameters
+

Task : TestCustomer | Populates : TestCustomer | Type : datapipeline

Key GraphDB	Value GraphDB	
Key BDATM	Value BDATM	
Key \$RUNTYPE\$	Value NULL	
Key \$BATCHRUNTYPE\$	Value NULL	
Key \$BATCHTYPE\$	Value 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 11-34 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	
\$RUNTYPE\$	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

User Guide
G52879-05
Copyright © 2014, 2026, Oracle and/or its affiliates.

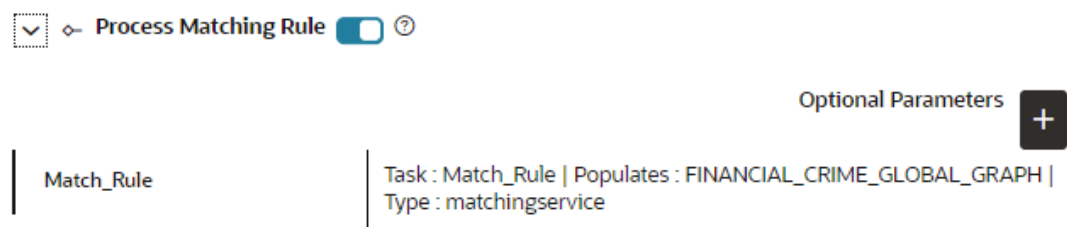
April 17, 2026
Page 162 of 166

Table 11-34 (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 11-172 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 11-35 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 11-35 (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 11-173 Refresh Graph

The screenshot shows a user interface for configuring a 'Refresh Graph' task. At the top, there is a toggle switch labeled 'Refresh Graph' which is currently turned on, accompanied by a question mark icon. Below this, the task name 'Refresh_Graph' is displayed. To the right, there is a section titled 'Optional Parameters' with a plus icon to add more. The task details are shown as 'Task : Refresh_Graph | Populates : FINANCIAL_CRIME_GLOBAL_GRAPH | Type : refresh'. Below the details, there are two input fields: 'Key' with the value 'graphDatasource' and 'Value' which is empty. A 'Required' label is positioned below the 'Value' field. A trash icon is located to the right of the 'Value' field.

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 11-174 Audit History

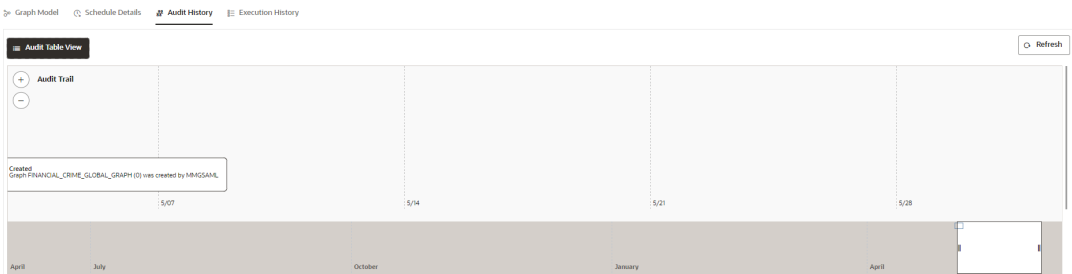


The screenshot shows the 'Audit History' tab selected in the top navigation bar. Below the navigation bar, there is a 'Refresh' button. The main content area displays a table with the following columns: Status, Type, Name, User, Comments, and Time. A single row is visible with the following data: Status: Created, Type: Graph, Name: FINANCIAL_CRIME_GLOBAL_GRAPH, User: MMGSAML, Comments: Graph FINANCIAL_CRIME_GLOBAL_GRAPH (0) was created by MMGSAML, Time: 2 May 2023, 17:40:12.

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 11-175 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.

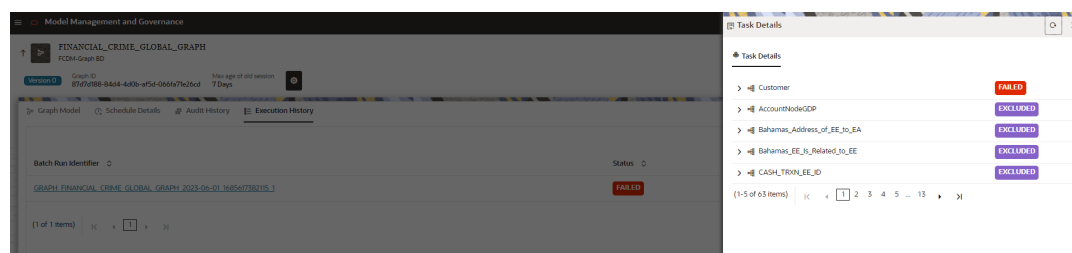
- Click on the graph for which you want to view the execution history.

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 11-176 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":""
}

```

① 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.

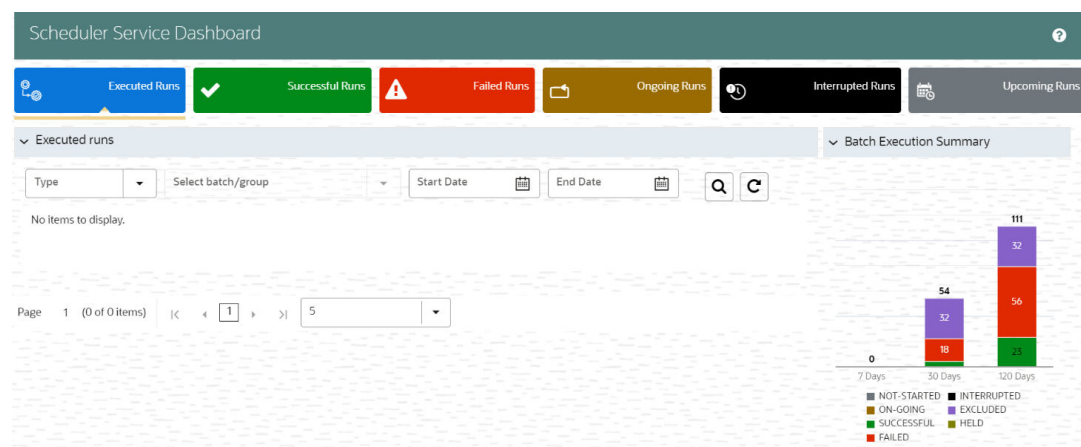
To view the demonstration of the Dashboard window, see the [Scheduler Service Introduction video](#).

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, on-going 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.

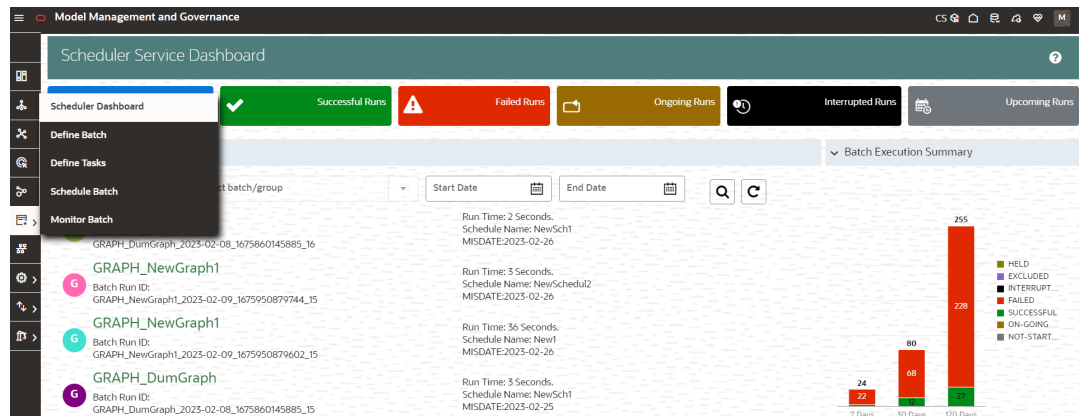
Figure 12-1 Scheduler Service Dashboard



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	The Batch Name is generated based on the values provided by you. NOTE: - 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	Enter a description for the Batch based on the Batch Name. 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.
Service URL Name/ Service URL Notify on Mail	- 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. 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. NOTE: E-mail is a mandatory field in IDCS. The BATCH_NOTIFY_FUNCT function has to 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 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	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 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	Select the task type from the drop-down list: 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. Executable: Used to run shell scripts (.sh) or other system-level executables.
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. 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. 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). 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.

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 Batch 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.

The Scheduler UI displays the configured timezone and its display name across all batch scheduling screens. UTC (Coordinated Universal Time) is the default timezone. For the list of supported timezones, see the [List of Supported Timezones](#) topic.

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.

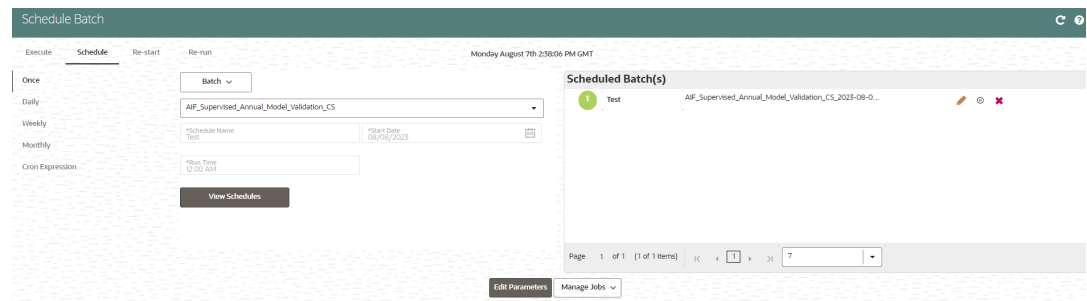
To pause a batch:

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:

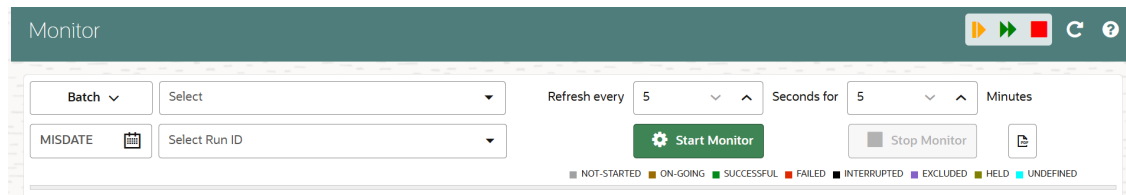
```
./EICSchedulerService.sh -o 1 -u mmgadmin -w CS -c ofsauser -s secret -x 232 -
b BatchModel -t rest -v '{"batchParams":
{"$FICMISDATE$":"2021-11-30", "$BATCHRUNID$":"BATCHRUNID"}, "taskRuntimeParams":
{"Task1":{"optionalparams":"parameterSets=Set1"}, "Task2":
{"optionalparams":"parameterSets=Set2"}}}
```

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.

Figure 12-4 Monitor Batch



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.
6. To view log information, click the log icon in the **List View** tab. The **Log Viewer** window appears displaying the log details.
7. Click the Download icon to download the log, or click the close icon to close the log viewer.

Note

On the **Batch Monitor** screen, click **Download PDF** to generate and download a PDF report for the selected batch run.

Figure 12-5 Generate PDF Icon



The PDF report is also saved in the <ftpsharepath>/printservice/pdfs/scheduler/<DATE>/ path, where <ftpsharepath> is the FTP share path that was configured during the installation.

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:

1. Click **Monitor Batch** from the Header panel.

The **Monitor** window is displayed.

2. Select the **Batch Group from the Select** drop-down and then select the **Batch Run ID from the Run ID** drop-down.

3. 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.
4. 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

Table 12-4 (Cont.) General configuration parameters

Parameter	Description	Default Value
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	Defines the additional execution time (in percentage) allowed beyond the last successful batch completion time before sending a threshold notification email. 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%). 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. For instance, the next notification would be triggered after 20% of 72 minutes (i.e., 86.4 minutes), and so on.	20
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

Note

If general email notification is disabled and only if the threshold email notification is enabled, scheduler sends only the threshold email.

Table 12-4 (Cont.) General configuration parameters

Parameter	Description	Default Value
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 <code>application.properties</code> the parameter <code>token.clientid</code> is set to <code>ofsuser</code>

Table 12-5 (Cont.) List of possible flagged arguments to be passed while executing EICSchedulerService.sh

Flag	Description	Comments
-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	• Execute batch "batch1": – Syntax: . / EICSchedulerService.sh -o 1 -u <ofs_remote_user> -w <Workspace Id> -c <Client Id for authentication>	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.

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	-s < Client Secret for authentication > -x < External Unique Id > -b < Object Name > -t < Object Type> — Example: ./EICSchedulerService.sh -o 1 -u scheduler -w WS1 -c ofsouser -s secret -x batch1_1001 -b batch1 -t rests • Execute batch “batch1” with the Dynamic Parameter list — Syntax: . / EICSchedulerService.sh -o 1 -u < ofs_remote_user > -w < Workspace Id > -c < Client Id for authentication > -s <		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
	Client Secret for authenti cation > -x < External Unique Id > -b < Object Name > - t < Object Type> -v <Dynamic Runtime paramete r list>		-1	Batch or BatchGroup execution is failed.
	— Example: ./EICSched ulerServ icerService.sh - o 1 -u mmgadmin -w CS -c ofsauser -s secret - x 232 -b BatchMod el -t rest -v '{"batch Params": {"\$FICMI SDATE\$": "2021-11 -30", "\$B ATCHRUNI D\$": "BAT CHRUNID" , "param_ 1": "qwer t", "para m_2": "ab cde"}, "t askRunTi meParams ": {"Task1" : {"p1": "1			

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	1102", "p 2": "2220 2", "abc" : "123342 ", "extra P": "wert 2y"}}, "Ta sk2": {"abc": " 1233", "p 3": "3333 ", "extP2 ": "0983" }}}'			
				i N o t e

T

h

e

T

a

s

k

/

B

a

t

c

h

p

a

r

a

m

e

t

e

r

s

s

h

o

u

l

d

b

e

t

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	i h e a c t u a l p a r a m e t e r k e y . • Execute batch “batch1” with “task1” and “task2” included: – Syntax: . / EICSched ulerServ ice.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			

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	Name > - t < Object Type> -I < Included Jobs with comma separate d values> — Example: ./ EICSchedulerService.sh - o 1 -u scheduser -w WS1 -c ofsouser -s secret - x batch1_1 001 -b batch1 - t rest - i task1,task2 • Execute batch “batch1” with “task1” as excluded and “task2” as held: — Syntax: ./ EICSchedulerService.sh - o 1 -u < ofs_remove_user > -w < Workspace Id > - c < Client Id for authentication >			

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	-s < Client Secret for authenti cation > -x < External Unique Id > -b < Object Name > - t < Object Type> -e < Excluded Jobs> -h < Held Jobs> — Example: ./ EICSched ulerServ ice.sh - o 1 -u scheduse r -w WS1 -C ofsauser -s secret - x batch1_1 001 -b batch1 - t rest - e task1 -h task2			
Status	• Get status of execution having runid = MMG_R1_2022-09-07_1662557327886_1: — Syntax: ./EICSchedulerService.sh -o 2-u <ofs_remove_user	To check the status	-3 -4 -5 0	Batch or Batchgroup Run Id does not exist. Invalid arguments passed in request/not enough params in Request body Invalid Request Headers Batch or batchgroup status is success

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	> -w < Workspac e Id > - c < Client Id for authenti cation > -s < Client Secret for authenti cation > -r < Batch Executio n Id>		-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.
	- Example: ./EICSched ulerServ ice.sh - o 2 -u scheduse r -w WS1 -c ofsauser -s secret - r MMG_R1_2 022-09-0 7_166255 7327886_ 1			
Restart	Restart execution for batch “MMG_R1” having runid = MMG_R1_2022-09-07_1662557327886_1: Syntax: ./EICSched ulerService.sh -o 3-u < ofs_remote_user > -w <	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

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	Workspac e Id > - c < Client Id for authenti cation > -s < Client Secret for authenti cation > -b < Object Name> -r < Batch Executio n Id> — Example: ./EICSched ulerServ ice.sh - o 3 -u scheduse r -w WS1 -c ofsauser -s secret - b MMG_R1 -r MMG_R1_2 022-09-0 7_166255 7327886_ 1		-1	Batch or BatchGroup restart is failed.
Rerun	• Rerun execution for batch “batch1” having runid = batch1 -r batch1_2022-09-01_1662011065557_1: — Syntax: ./EICSched ulerServ ice.sh - o 4-u < ofs_remo	To rerun the batch execution	-4 -5 -3 0	Invalid arguments passed in request/not enough params in Request body Invalid Request Headers Batch or BatchgroupId or Run Id does not exist. Batch or BatchGroup rerun is success

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	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> -r < Batch Executio n Id>		-1	Batch or BatchGroup rerun is failed.
	— Example: ./ EICSched ulerServ ice.sh - o 4 -u scheduse r -w WS1 -c ofsauser -s secret - x batch1_1 002 -b batch1 - r batch1_2 022-09-0 1_166201 1065557_ 1		-9	Duplicate External Unique Id
Interrupt	• Interrupt execution for batch “batch1” having runid = batch1_2022-	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

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description
	09-07_166253 0601924_1:		-5	Invalid Request Headers
	– Syntax: ./EICSchedulerService.sh -o 5 -u <ofs_remote_user> -w <Workspace Id> -c <ClientId for authentication> -s <ClientSecret for authentication> -b <ObjectName> -r <BatchExecution Id>		0	Batch or BatchGroup interrupt is success
			-1	Batch or BatchGroup interrupt is failed
	– Example: ./EICSchedulerService.sh -o 5 -u scheduler -w WS1 -c ofsouser -s secret -b batch1 -r batch1_2 022-09-07_166253 0601924_1		-7	Batch or BatchGroup status is not ongoing or already interrupted.

Table 12-6 (Cont.) List of possible exit codes returned by “EICSchedulerService.sh

API	Syntax	Description	Status Code	Code Description

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 DEM07 mmgadmin BatchRest batch`
N CS_batch_BatchRest

MMG Scheduler Shell Utility (mmg_scheduler.sh)

The `mmg_scheduler.sh` utility provides a command-line interface to trigger and monitor OFS MMG batches or batch group executions. Use this utility to:

- Trigger an immediate batch or batch group execution by name.
- Include, exclude, or hold certain selected tasks within the execution.
- Optionally, specify custom batch parameters.
- Query the execution status of any batch or batch group by using its run ID.
- Monitor the progress automatically (auto-follow) or check its status as needed.

The `mmg_scheduler.sh` utility can be found in `OFS_MMG/bin`.

General Syntax

`./mmg_scheduler.sh [core options] [mode options] [execution options] [monitoring options]`

The utility supports two main modes:

- **trigger** — Starts a batch or batch group execution.
- **status** — Queries status of an existing run using its run ID

Core Options (Required for Most Commands)

These options are typically required for both trigger and status calls:

- **-u <USER>**: MMG user identifier
- **-w <WORKSPACE>**: Workspace name
- **-c <CLIENT_ID>**: client ID
- **-s <CLIENT_SECRET>**: client secret
- **-B <BASE_URL>**: MMG base URL (for example:
"https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/<WEB_CONTEXT>/")

Triggering Execution

– Trigger a Batch (Default)

```
./mmg_scheduler.sh -m trigger -b <BATCH_NAME> -u <USER> -w <WORKSPACE> -c
<CLIENT_ID> -s <CLIENT_SECRET> -B "https://
<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/<WEB_CONTEXT>/"
```

Example

```
./mmg_scheduler.sh -m trigger -b Batch_Model1 -u mmgadmin -w CS -c
ofsauser -s secret -B "https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/
<WEB_CONTEXT>/"
```

– Trigger a Batch Group

Use **-t group** to indicate that the name provided with **-b** is a **batch group**:

```
./mmg_scheduler.sh -m trigger -b <BATCH_GROUP_NAME> -t group -u <USER> -w
<WORKSPACE> -c <CLIENT_ID> -s <CLIENT_SECRET> -B "https://
<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/<WEB_CONTEXT>/"
```

Example

```
./mmg_scheduler.sh -m trigger -b Batch_Group1 -t group -u mmgadmin -w CS -
c ofsauser -s secret -B "https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/
<WEB_CONTEXT>/"
```

The following table provides information on the additional options that are supported by this utility:

Table 12-7 Additional Options Supported by the mmg_scheduler.sh Utility

Option	Usage
Customize Task Execution	Include, Exclude, or Hold specific tasks: • -i TaskA, TaskB - Use this to include only specified tasks • -e TaskC - Use this to exclude specified tasks • -H TaskD - Use this to keep specified tasks on hold (not run in this execution) The -i task will always take precedence over the -e and -H options
Pass Custom Parameters	Use the following option to pass custom parameters: Use -v to pass a JSON payload to specify: • batchParams: parameters applied at the batch level • taskRuntimeParams: task-specific parameters keyed by task name -v '{"batchParams": {"param1": "value"}, "taskRuntimeParams": {"Task1": {"param": "value"}}}' Example <pre>./mmg_scheduler.sh -m trigger -b Batch_Model1 -v '{"batchParams": {"param1": "value1"}, "taskRuntimeParams": {"Task1": {"param": "value"}}}' -u mmgadmin -w CS -c ofsauser -s secret -B "https:// <MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/ <WEB_CONTEXT>/'</pre>
Monitoring & Polling Options	• --follow (default) - This option will poll the execution status automatically after being triggered. – Use -T <seconds> to adjust the auto-follow timeout. The default timeout value is 20 seconds. – Use -n <seconds> to change the polling interval. The default interval is 10 seconds. • --no-follow - Only triggers the batch and does not monitor its status. Example: Trigger and follow for up to 10 minutes <pre>./mmg_scheduler.sh -m trigger -b Batch_Group1 -t group -u mmgadmin -w CS -c ofsauser -s secret -B "https:// <your-server>:<mmguiport>/mmg8133/" -- follow -T 600 -n 10</pre>

Table 12-7 (Cont.) Additional Options Supported by the mmg_scheduler.sh Utility

Option	Usage
Filter status by individual tasks	Use this to filter the status for individual selected tasks: -k TaskA, TaskB This filter is only applicable for the status option. Example <pre>./mmg_scheduler.sh -m status -r Batch_RunID_12345 -k TaskA,TaskB -u mmgadmin -w CS -c ofsauser -s secret -B "https:// <MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/ <WEB_CONTEXT>/"</pre>

- **Querying the Status by Run ID**

Use **-m status** and provide the run ID by using **-r**.

- **Syntax**

```
./mmg_scheduler.sh -m status -r <BATCH_RUN_ID> -u <USER> -w <WORKSPACE> -c
<CLIENT_ID> -s <CLIENT_SECRET> -B "https://
<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/<WEB_CONTEXT>/"
```

- **Example**

```
./mmg_scheduler.sh -m status -r Batch_RunID_12345 -u mmgadmin -w CS -c
ofsauser -s secret -B "https://<MMG_SERVICE_HOST>:<MMG_GATEWAY_PORT>/
<WEB_CONTEXT>/"
```

Help/Usage Information

To display the list of supported options and usage examples, run:

```
./mmg_scheduler.sh --help
```

Authentication

The script requires valid OFS MMG credentials (client ID or secret), workspace, and user identity, as it has been configured in your OFS MMG deployment.

Typical Workflow

The following is the typical work flow of the utility:

1. Trigger a batch or group job. This will return the batch run ID.
2. Auto-follow or manually follow and monitor the batch or batch group status. This will return the task or batch progress and results in real time.
3. Optional, you can re-query the job status any time by using the batch run ID.

Handling Errors

Errors are clearly reported for invalid credentials, unreachable APIs or servers, or invalid batches or run IDs. For failed tasks, status, and errors, they can be viewed in the command output.

Notes and Limitations

- The default polling automatically times out after 20 seconds unless it has been overridden.

- Filtering and custom parameters support for batch groups might be limited.
- Always use the correct client ID or secret. These values are configured in the OFS MMG UI **application.property** file:
 - **aai.client.id**
 - **aai.client.secret**

In the **application.property** file under **mmg-ui** service, the secret may appear masked (because it is sensitive). If the actual value is not visible, contact your MMG administrator (or the team managing your deployment secrets) to obtain the correct client ID/secret.

- For failures or unexpected results, check the command output for error messages.

Scheduler Configuration / Admin User Interface

The Scheduler Configuration / Admin User Interface (UI) is used to manage the scheduler notification settings and to configure email notification recipients for specific batches or batch groups.

Access Control

To access the Scheduler Configuration/Admin UI, you must have the **BATCH_ADMIN** role. By default, the **BATCH_ADMIN** role is mapped to the seeded user group **SCHEDULERADMIN**.

From the Orchestration page, navigate to the Scheduler Configuration/Admin UI.

Scheduler Configuration/Admin UI Features

The Scheduler Configuration/Admin UI provides the following functionalities:

- **General Configuration** (system-level configuration parameters)
- **Batch/Batch Group to User Mapping** (workspace-level email mapping)
- **Batch/Batch Group to User Mapping** (workspace-level)

You can use this page to map specific users to a batch or batch group for email notifications. This configuration is workspace-specific.

Access the Batch/Batch Group to User Mapping

In the Scheduler Configuration/Admin UI, select **Batch to User Configuration**. Click the **Eye** icon adjacent to the Batch to User Configuration tile (if applicable) to open the page.

Search

To search for a batch or batch group, enter the desired keywords in the **Search** field and then click **Search**. This feature enables you to search by the batch name, batch code, and batch type.

Add a Mapping

Perform the following steps to add a mapping:

1. Click **Add**.
2. Select the batch type.
3. Select the required batch or batch group.
4. Select the required user(s).
5. Click **Create**.

A message appears informing you that indicating that the mapping was successfully created.

Manage Existing Mappings (Action menu)

The **Batch to User Mapping** page lists all the batch or batch groups that are mapped to user(s). Use the **Action** menu to:

- **View:** This displays mapping details.
- **Edit:** Updates the mapped user list (add/remove users).
- **Delete:** Removes the batch/batch group to user mapping.
- Email notification behavior (mapping vs default recipients)

The system can send notifications to default recipients based on the configured notification role/function. If a batch/batch group is explicitly mapped to a user(s), notifications for that batch/batch group are sent only to the mapped user(s). If no batch-to-user mapping is configured for a batch/batch group, notifications are sent to the default recipients.

General Configuration (system-level)

Use this page to update scheduler service configuration parameters. General Configuration values are system-level; a change made in one workspace is reflected across all workspaces.

Access General Configuration:

Perform the following steps to access the general configuration page:

1. In the Scheduler Configuration/Admin UI, select **General Configuration**.
2. Click the **Eye** icon on the **General Configuration** tile (if applicable) to open the page.
3. Update configuration values.
4. Click **Edit**.
5. Modify the required configuration values.
6. Click **Save** to apply the changes.

OR

Click **Cancel** to discard changes and revert to the previous values.

Note

- Batch/Batch Group to User Mapping is workspace-specific.
- General Configuration is system-level and is effective across all workspaces.
- Access to this page requires the **BATCH_ADMIN** role (seeded through the **SCHEDULERADMIN** group by default).

Model and AI Risk Management (MARM)

The Oracle Financial Services Model and AI Risk Management (MARM) platform provides a robust, end-to-end framework for governing the lifecycle of models such as Machine Learning (ML) and statistical models.

This guide provides instructions for Modelers, Validators, and Risk Managers to navigate the system, ensure regulatory compliance (SR 11-7, SR 15-18), and mitigate model-related risks.

Key Capabilities

- **Model Dashboard:** A KPI driven central command center for model risk oversight, providing real-time visibility into the organization's model risk.
- **Centralized Model Inventory:** A single source of truth for all enterprise models.
- **Multi-Stage Validation:** Structured workflows for technical and conceptual assessments.
- **Automated Monitoring:** Real-time drift detection and performance back-testing.
- **Findings Management:** End-to-end tracking of model deficiencies and remediation.

Getting Started with Model and AI Risk Management

This section outlines the requirements for using Model and AI Risk Management (MARM).

Model Registration in MARM

To ensure a seamless transition between development and governance, MARM eliminates the need for redundant data entry by leveraging a unified ecosystem.

To initiate any model's validation in MARM, the model must first be available in MMG. MARM supports a pull-based approach, allowing users to onboard models from MMG for validation and monitoring.

There are two ways to register and use models:

- Local Model Registration
- External Model Registration

Local Model Registration

1. If the model is developed locally:
 - a. Create the model within the environment.
 - b. Provide the required Pipeline Metadata, including:
 - i. Dataset
 - ii. Model Type
 - iii. Target Column
 - iv. Prediction Column
 - v. Date-Time Column

- vi. Training Start Date
- vii. Training End Date
- c. Publish the model pipeline.
Once published:
 - i. The model version is created and becomes a read-only model.
 - ii. You can pull it into MARM.
 - iii. The Validator can initiate the validation.

External Model Registration

Externally developed models can be on-boarded into MMG by creating the appropriate dataset and providing the correct training dates through the pipeline. MMG supports multiple source types, including files and tables.

Validation data does not need to be imported separately. MMG computes validation on demand using the actual dataset, based on dataset filters and the provided training or as-of dates.

Note

The current approach does not preserve historical validation results from the original external environment and assumes that the underlying data remains available for recomputation.

If the model is coming from an external source:

- a. Register the model in MMG.
- b. Ensure all required Pipeline Metadata is provided.
- c. Additionally, register the associated datasets, as they are required for metadata selection.
- d. Publish the model pipeline.
Once published:
 - i. The model version is created and becomes a read-only model.
 - ii. It can be pulled into MARM for further validation and usage.

Running Example: Retail PD Model

- **Model Name:** RETAIL_PD_MODEL_V1
- **Business Purpose:** Predicts the likelihood of a customer defaulting on a credit card.
- **Source:** Pulled directly from the MMG Modeler Workspace.
- **Validation due date:** 01-April-2026

Access and Security

The MARM platform is secured through a robust Role-Based Access Control (RBAC) architecture.

Access to specific modules, data, and validation tools is strictly governed by the Role Groups assigned to your user profile.

Authentication and Login

Accessing the system is a straightforward process designed for enterprise security:

- **Simple Credentials:** Users log in using their enterprise-authorized username and password credentials.
- **Role Identification:**
 - Two primary functions are defined: one for viewing validation results and another for executing validations. These functions are mapped to corresponding roles—read-only and execute—to separate viewing and action-based permissions.
 - Additionally, a dedicated user group (**MMG_MDL_VLDN_ADMIN**) is configured, which is assigned both roles, enabling full access to view and run validations. This structure ensures controlled access, clear separation of responsibilities, and governance over validation activities.

Seamless Persona Switching

It is designed for high-efficiency workflows, particularly for users with oversight responsibilities across the model lifecycle:

- **Integrated Ecosystem:** Either directly log-in as a model validator by selecting from initial login options or once logged into the **Modeler Workspace (MMG)**, users can navigate directly to the **Validator Framework via Switch to Validator Mode** option in top right profile icon's drop-down menu.
- **Single Session Access:** If your Role Group has permissions for both environments, you can switch between modeling and validation spaces.
- **No Re-authentication:** There is no requirement for a separate login or session interruption when moving between the Modeler and Validator interfaces, ensuring a continuous and fluid user experience.

MARM Dashboard

The Executive Dashboard serves as the central command center for model risk oversight, providing real-time visibility into the organization's model landscape.

Global Filters and Controls

Users can refine and navigate models using a set of global filters and controls. These include filtering by workspace scope (all, production, or specific workspace), along with additional filters such as tier and status. A search option is available to quickly locate models using attributes like ID, name, version, objective, workspace, or dates. Users can reset all applied filters using the **Reset Filters** option.

Table View and Actions

Real-Time KPIs

Upon logging in, users are presented with a high-level summary of the model inventory's health and governance status. These KPIs provide immediate visibility into the validation lifecycle and the current risk landscape:

- **Total Tracked Models:** Represents the cumulative headcount of all models ever introduced to the validation framework, including those currently undergoing review and those with finalized assessments.
- **In-Progress Validation Cycle:** Tracks the total number of models currently active within the validation workflow, highlighting the current workload of the MRM team.
- **Completed Validation Cycle:** Displays the count of models that have successfully concluded the independent oversight process and have a finalized validation status.

- **Overdue Validations:** Flags models that have surpassed their scheduled review dates, identifying critical gaps in governance that require immediate attention.
- **High Risk Tier:** A focused count of models classified under the highest risk category, ensuring these high-impact assets remain under constant executive visibility.
- **Due in Next 30 Days:** A proactive metric highlighting models approaching their validation or approval expiry within the month, allowing teams to initiate reviews early and prevent governance lapses.

Interactive Visualization

The MARM dashboard utilizes dynamic, color-coded visuals to transform complex inventory data into actionable risk insights.

- **Core Visualizations:** The system employs distribution charts to categorize the model universe across three governance tiers.
 - **Risk Distribution:** High-level views of Risk Tiers (High/Medium/Low), Business Domain concentration, and Model Type (Traditional vs. AI/ML).
 - **Operational Criticality:** Tracking Model Lifecycle Stages and Third-Party/Vendor model concentration.
 - **Governance Depth:** Analysis of Production Model Vintage to identify aging models requiring re-validation.
- **Click-Through Drill Down:** Every chart element is fully interactive and serves as a navigational shortcut.
 - **Filtered Redirection:** Clicking any segment (for example the "High Risk" slice of a pie chart) instantly redirects the user to the Model Inventory, pre-filtered to show only the models driving that specific metric.
 - **Contextual Access:** This eliminates manual searching, allowing users to jump from an executive metric directly to specific model metadata or validation records in a single click.

Calendar and Deadline Integration

The dashboard integrates a centralized Validation Calendar:

- Track upcoming regulatory deadlines and compliance milestones.
- View scheduled model reviews and optimum resource allocations to ensure no statutory deadlines are missed.
- **Operational Example:** For our RETAIL_PD_MODEL_V1, the calendar shows a validation deadline of 01-April-2026 and carries forward the count until validated.

Model Inventory Management

Model list displays key details such as Name, Version, Objective, Workspace, Tier, Last Validation Date, Next Validation Due, and Status. The Name field is clickable, allowing users to navigate to model detail view.

Pagination is provided for easy navigation across records.

Common actions include:

- Filtering models based on required criteria
- Refining results using search
- Accessing model details by selecting the model name for further inspection or validation

Model Details View

The Details View is a central feature of the Model Inventory that allows users to access comprehensive model information from the main list. By clicking on any model in the inventory, it navigates to provide a 360-degree view of the model's structure and governance status.

- **Integrated Model Metadata and Lineage:** The view summarizes all core identifying information, including the model's technical specifications, open findings, versioning, and assigned risk tier.
- **Technical and Development Assets:** To support deep technical reviews, this provides direct access to the model's underlying assets:
 - **Model Pipeline and Code:** View the end-to-end execution pipeline and source code.
 - **Model Notebooks:** Access the original development notebooks for insights into the model's experimental phase.
 - **Model Documentation:** Centralized repository for whitepapers, technical specifications, and user manuals.
- **Governance and Action Center:** This also serves as the primary gateway for validation activities:
 - **Validation History:** A chronological log of all past validation cycles, outcomes, and recorded drift events.
 - **Launch or Resume Validation:** A high-priority action button that allows authorized users to initiate a new or resume draft validation cycle directly from the drawer.

Validation Workflow

The MARM validation process is a systematic, six-stage flow designed to ensure model integrity—from initial data profiling to final report generation.

Each stage must be completed and approved sequentially to progress.

1. **Initiation Methods:** Validation can be launched using two primary entry points:
 - a. **Model Inventory:** Perform the following:
 - i. Search for the model in the model inventory list
 - ii. Click on the model's Name. It will redirect to the **Model Details View** page.
 - iii. Click the **Start New Validation** button for a fresh validation or **Resume Validation and Discard Draft** in case of on-going validation.
2. **Managing Validation States**
 - a. **Start New Validation:** To start a new validation.
 - b. **Resume Validation:** If a validation is already in progress, the system identifies the existing draft, allowing you to pick up exactly where you left off.
 - c. **Discard Draft:** If the current analysis is no longer relevant, you can discard the active draft to enable a fresh validation launch.
3. **Operational Example: RETAIL_PD_MODEL_V1** To begin the workflow for our PD model:
 - a. Navigate to the **Model Inventory** and search for RETAIL_PD_MODEL_V1.
 - b. If multiple versions are available for the same name, validator need to select the appropriate version.

- c. Click on the name RETAIL_PD_MODEL_V1 from the list view. It will redirect to the **Model Details View** page for this model.

Select **Start New Validation** or **Resume Validation**, if validation is already going on. The system will automatically pull the latest artifacts (code/data) to commence validation.

Data Input and Profiling

This stage provides a deep-dive into the data health and stability of the model, ensuring the input features remain consistent with the data used during the initial training phase.

- **Drift and Distribution Analysis:** The system automatically executes Python-based statistical profiling to identify Data Drift over time. It assesses the skewness and normality of the current dataset to determine if the model's original assumptions still hold true.
- **Comparison Configuration:** To perform the analysis, configure the following:
 - a. **Current Dataset:** Define the period by entering a specific Date Range or selecting from presets: Last 7 days, 30 days, or 90 days.
 - b. **Baseline Dataset:** Select the reference point for comparison, either the original Training data or the Last Validation Run (which automatically fetches the relevant historical dates and data).
- **Execution and Visualization:** Upon clicking Compute, the engine compares the datasets, automatically plotting distribution graphs for every feature. The system identifies "Drifted Features" where the statistical variance has breached system-defined thresholds.
- **Operational Insights:** These profiles provide the technical justification for model adjustments, such as tweaking specific parameters or adding new data transformations to handle outliers or noise.

Operational Example: RETAIL_PD_MODEL_V1

For the Retail PD model, the validator follows these steps to ensure data stability:

1. **Selection:** The validator selects the Last 90 days as the Current Dataset and checks the Training dataset option in the Baseline section to compare against the model's inception logic.
2. **Computation:** After clicking Compute, the system generates drift graphs for all input variables.
3. **Observation:** The system flags that the Average Credit Score in the applicant pool has drifted significantly (skewed lower) compared to the baseline. This insight is automatically captured in the drift table.
4. **Action:** The validator notes in the comments that the model may need a recalibration of weightings to remain accurate for the current economic climate and proceeds to the next validation step.

Outcome Analysis & Backtesting

The Backtesting Engine evaluates model performance by comparing predicted outcomes with actual observed values over a selected production window, and benchmarking the results against the training dataset and/or the last validation run. This helps assess predictive accuracy, consistency, and performance stability over time.

- **Core Functionality:** Comparison of model predictions against actual outcomes for a selected production period, with reference to historical benchmarks (training or previous validation results).
- **Workflow Process:**

1. **Current Dataset Selection:** Define the period by inputting a specific Date Range or selecting from presets (Last 7 days, 30 days, or 90 days).
 2. **Baseline Selection:** Choose the reference benchmark—either the Training dataset or the Last Validation Run—to automatically fetch comparison dates & data.
 3. **Dynamic Metric Filtering:** Upon selecting the datasets, the system automatically filters and displays relevant statistical techniques (for example Gini, AUC, F1-Score) based on the Model Type (Regression or Classification).
- **Execution and Validation:** Clicking Compute triggers the engine to calculate and display a side-by-side metric comparison across the current, baseline, and last validation datasets. The Validator reviews these results, adds step-specific comments, and proceeds. Supports a comprehensive set of statistical metrics:
 - **For regression model:** MAE, MSE, RMSE, MAPE, R2 Score, Median AE, Max Error, Percentiles, Mean Error, Std. Error.
 - **For classification model:** True positive, True Negative, False Positive, False Negative, Accuracy, Precision, Recall, F1 Score, AUC ROC, PR AUC, Log Loss, Brier Score.

Operational Example: RETAIL_PD_MODEL_V1

To validate the predictive power of the Retail PD model, the validator executes the following steps:

- **Define Scope:** The validator selects the Last 90 days as the Current Dataset. Under the Baseline section, they select Training and Last Validation Run to enable a three-way performance comparison.
- **Technical Selection:** Because the system recognizes the PD model as a Classification type, it automatically populates metrics such as Gini, AUC, and ROC.
- **Compute:** The validator clicks Compute. The system generates a report comparing the current Gini coefficient (e.g., 0.62) against the Training Gini (e.g., 0.65) and the Last Validation Gini (e.g., 0.64).
- **Results and Documentation:** The system displays the performance graphs and metric tables. The validator observes that while there is a slight dip, the performance remains within acceptable limits.
- **Finalize:** The validator enters a comment noting the stability of the AUC/ROC curves and saves the step to proceed.

Conceptual Soundness

This stage evaluates the underlying logic and theoretical basis of the model through a structured assessment.

- **Questionnaire-Based Assessment:** The validator completes a targeted questionnaire by selecting the appropriate options for each question.
- **Final Review and Commentary:** After answering the questions, the validator provides an overall summary comment for the stage before moving to the next step.
- **Questionnaire Configurability:** To align with specific bank standards, the entire questionnaire is fully configurable. Users can update the questionnaire by modifying the JSON configuration file (questionnaire.json) in the MARM backend located at configuration\configuration\marm\conceptual_soundness without requiring code changes. There is a template file as well named questionnaire.template.json in the same folder that you can use to design a new questionnaire.

Operational Example: For RETAIL_PD_MODEL_V1, the validator will answer questions regarding the as per the slandered and provide a final assessment of the model's theoretical fitness for credit risk.

Model Tiering Review

The Model Tiering Review is a systematic assessment that classifies models into risk categories (e.g., High, Medium, or Low) based on their potential impact and complexity.

- **Automated Computation:** The system computes the final Risk Tier based on standard business logic. The validator selects scores for specific business parameters, such as Financial Impact and Regulatory Scope.
- **Tiering Criteria Configurability:** All tiering parameters and scoring weights are fully configurable via the back-end to align with specific internal organizational policies. Administrator can simply update the criteria by modifying the JSON configuration file i.e. criteria.json in the MARM back-end located at “\configuration\configuration\marm\tiering” without requiring code changes. There is a template file as well named criteria.template.json in the same folder that can be used to design the new criteria.

Operational Example: For RETAIL_PD_MODEL_V1, the validator checks all the criteria, read the description of criteria and each class in it & assigns the score. For this model, validator selects “High” class for “Regulatory Scope” and other parameters. The system then automatically calculates the model as final Tier on the basis of configuration.

Final Review

This stage serves as the final validation closure checkpoint, consolidating key findings, schedule decisions, tier outcomes, and reviewer conclusions into a single sign-off step.

- **Findings Consolidation:** The system presents section-wise findings from prior validation stages (Data Profiling and Data Analysis, Outcome Analysis and Backtesting, Conceptual Soundness, and Model Tiering Review), including comment counts and full comment history (author, timestamp, and message), so validators can complete closure with full context.
- **Validation Schedule Decision:** Validators can explicitly choose whether to set the next validation date. If enabled, a date is selected in-line; if not select Do Not Set Now, no date is recorded for this cycle.
- **Tier Outcome Summary:** The page displays the computed model tier result from Model Tiering Review, including tier, risk level, total score, and recommendation text for final reference.
- **Finding Summary By Section:** A consolidated log that maps the validator's specific comments and qualitative insights to their corresponding workflow stages, providing a centralized audit trail of all identified issues and comments. Final Decision and Reviewer Notes: Validators select a final decision (Approve / Require changes / Reject) and can add final reviewer comments to document closure rationale, residual risk, and sign-off notes.
- **Completion and Submission:** The Finish action is enabled after final decision selection, and submission completes the run and returns the user to Model Details with confirmation.

14

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 14-1 Object Export Summary page

Name	Status	Last Modified By	Action
EXP5	Success	mmgtest	[Menu]
EXP4	Success	mmgtest	[Menu]
EXP3	Success	mmgtest	[Menu]
EXP2	Success	mmgtest	[Menu]
EXP1	Success	mmgtest	[Menu]

Page 1 of 1 (1 - 5 of 5 items)

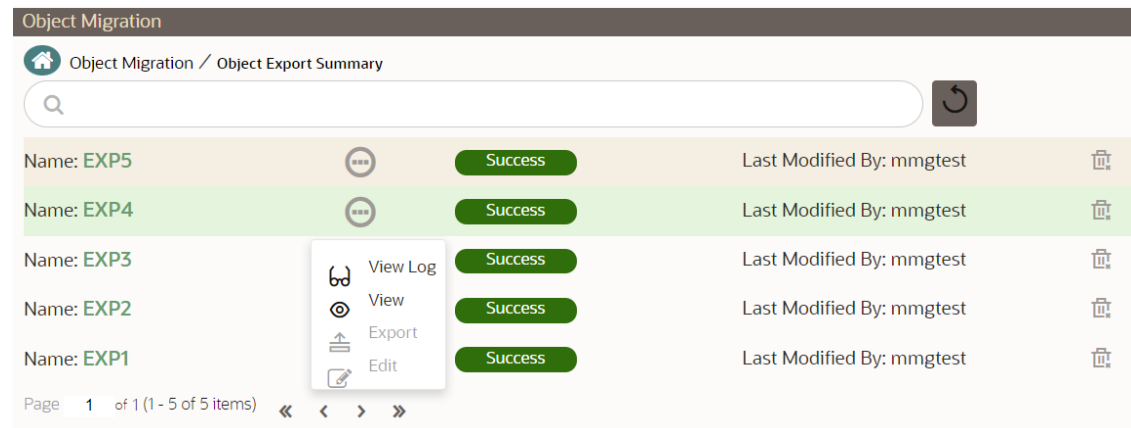
- **Name:** The unique migration name assigned to the collection when the migration definition was created.
- **Object Migration Status:** The migration status of the record corresponding to the specified Definition Name. The three migration status values are as follows:
 - **Success** - Set to Success, 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 14-2 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 14-3 Migration Definition window

Migration Definition

Migration Name Required

File Name

Apply Close

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 14-1 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 14-1 (Cont.) View Migration Export Objects

Field	Description
Export	Click Export to initiate Object Migration (Export) for a specific Migration Definition. When the migration is completed, the status will change from Saved to Success. For more information, refer to Exporting Migration Objects .
Edit	Click Edit to view and edit the objects linked to a Migration Definition. For more information, refer to Editing Migration Export Definitions .

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 14-2 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	The migration status of the export object. • 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 14-4 Object Import Summary page

Name	Status	Last Modified By
IMP4	Success	mmgtest
IMP3	Success	mmgtest
IMP2	Success	mmgtest
IMP1	Success	mmgtest

Page 1 of 1 (1 - 4 of 4 items)

- **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 14-5 Object Migration

Object Migration			
Object Migration / Object Import Summary			
Name: IMP4	Success	Last Modified By: mmgtest	
Name: IMP3	Success	Last Modified By: mmgtest	
Name: IMP2	Success	Last Modified By: mmgtest	
Name: IMP1	Success	Last Modified By: mmgtest	

Page 1 of 1 (1 - 4 of 4 items)

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

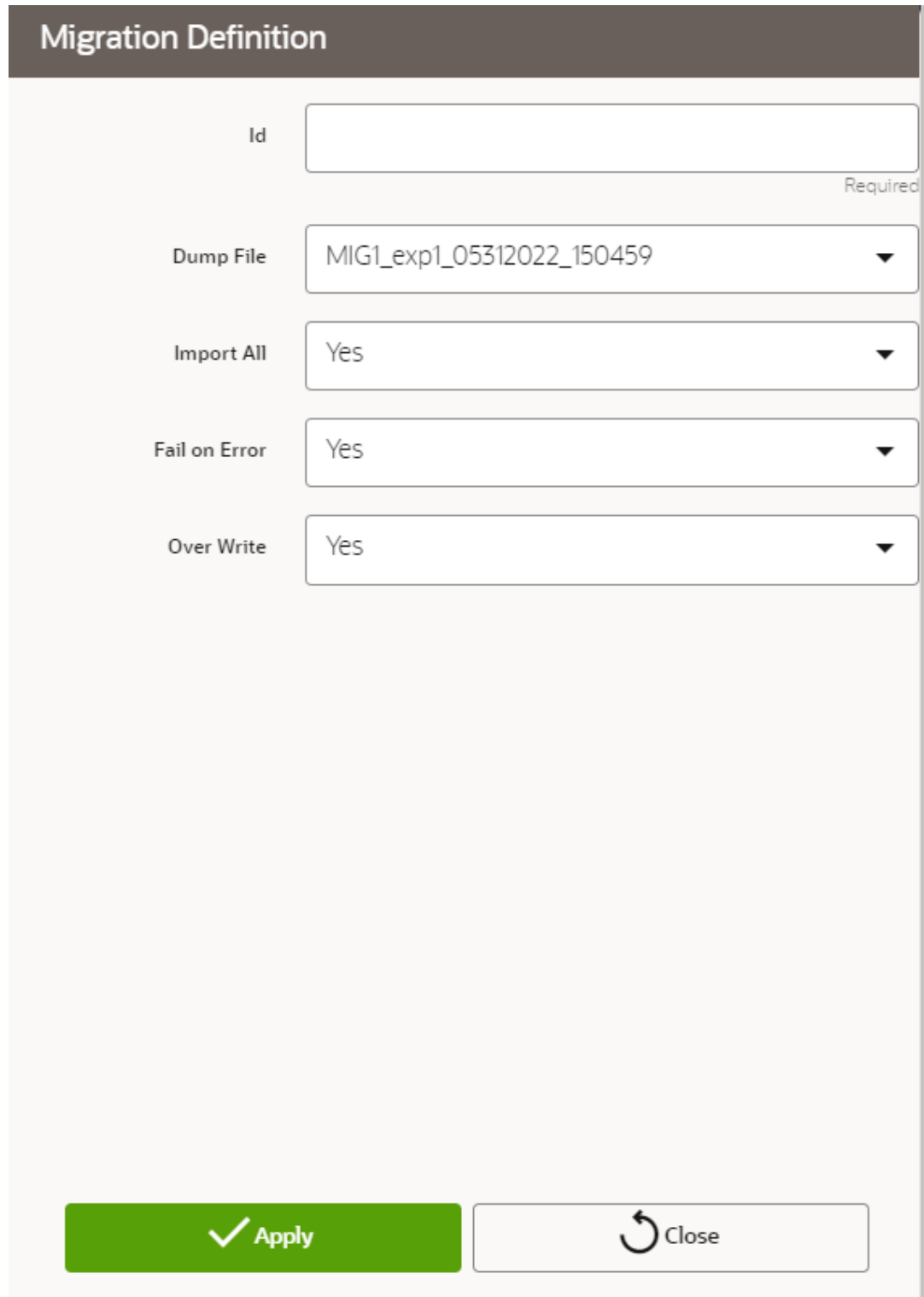
You can create Import Object definitions for migration for the following object types.

- 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 14-6 Migration Definition window

The screenshot shows the 'Migration Definition' window. It has a dark header bar with the title 'Migration Definition'. Below the header, there are five rows of form fields. The first row is 'Id' with an empty text box and a 'Required' label to its right. The second row is 'Dump File' with a dropdown menu showing 'MIG1_exp1_05312022_150459'. The third row is 'Import All' with a dropdown menu showing 'Yes'. The fourth row is 'Fail on Error' with a dropdown menu showing 'Yes'. The fifth row is 'Over Write' with a dropdown menu showing '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.

Migration Definition

Id Required

Dump File

Import All

Fail on Error

Over Write

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 14-7 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 14-3 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 14-4 View Log of Migration Import Objects

Field	Description
Object Migration ID	The migration ID associated with the export object.

Table 14-4 (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. • 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.

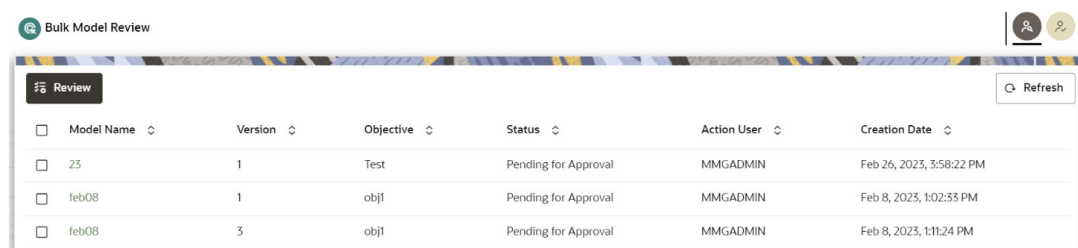
Model Actions

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 a 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 14-8 Review Models screen



The screenshot shows the 'Bulk Model Review' interface. At the top, there's a header with a 'Review' button and a 'Refresh' button. Below the header is a table with 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'.

Model Name	Version	Objective	Status	Action User	Creation Date
Z5	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 14-9 Review Models window

Review Models ×

Are you sure you want to review the below models for their respective requests?

☒	Model Name ▾	Version ▾	Status ▾
☒	23	1	Pending for Approval
☒	feb08	1	Pending for Approval

Comments

Cancel Review

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 14-10 Approve Models screen

Approve Models ×

Are you sure you want to approve the below models for their respective requests?

☒	Model Name ▾	Version ▾	Status ▾
☒	23	1	Pending for Approval
☒	feb08	1	Pending for Approval

Comments

Cancel Approve

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.

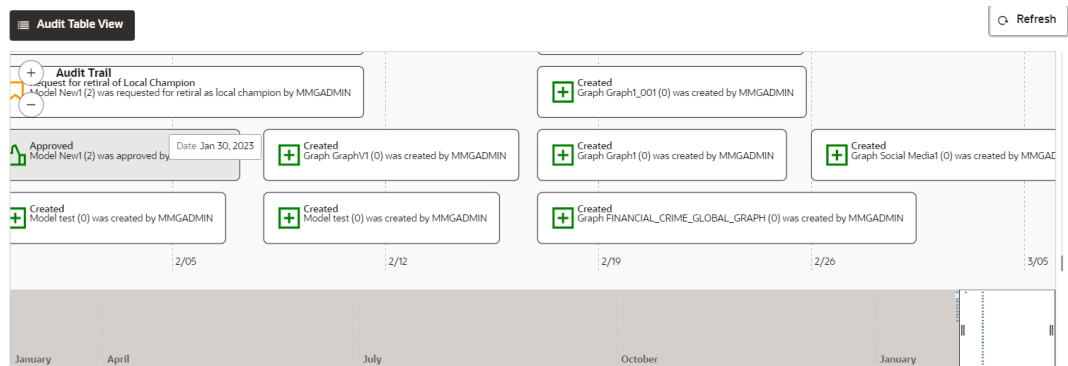
Figure 14-11 Audit Trail window

Audit Timeline View						Search	Refresh
Status	Type	Name	User	Comments	Time		
Created	Dataset notebook	XCASCKAS	MMGADMIN	Dataset notebook XCASCKAS (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, 08:56:33		
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, 18:00:32		
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	dft1	MMGADMIN	Model dft1 (0) was created by MMGADMIN	28 Feb 2024, 15:31:42		
Created	Model	dft9	MMGADMIN	Model dft9 (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 14-12 Audit Trail Window Timeline View with Horizontal Time Axis

About Data Transformation Platform

Data Transformation Pipelines enable organizations to quickly respond to changes in regulations, adapt to new product introduction, counter more sophisticated money laundering techniques in a quick and cost-effective manner by removing dependency on Oracle to develop the scenarios and extensions most appropriate to their business requirements.

Purpose

The purpose of the Data Transformation Platform is to provide Financial Institutions with full scenario configurability and extensibility, allowing them to quickly create scenario pipelines using complex logic, advanced configuration, leveraging extended data sources.

User Personas

This guide is intended for users who are responsible for implementing the scenarios in line with the Risk analysis formulated by the compliance team and by Compliance Officers who need to understand the additional possibilities offered by scenarios based on data they are capturing, in order to make their AML program more effective and efficient. Oracle expects these users to be senior compliance officers, who have significant understanding of their data, the specific behaviors of interest, and the logic required to generate meaningful alerts.

Getting Started with Data Transformation Pipelines

This section explains how to create, copy, and configure data transformation pipelines, and how to use the widgets available in the DTP Pipeline Designer.

To access the DTP Pipeline Designer, follow these steps:

1. In the Applications landing page, click the **Navigation Menu**  to access the Navigation List. The Navigation List displays the list of modules.
2. Expand **Business Processes** and then **Data Transformation Pipeline** in the Navigation List to view DTP menu options.
 - [DTP Pipeline Designer](#)
 - [DTP Threshold Manager](#)
 - [DTP Jobs](#)
 - [DTP Audit History](#)

To access Administrator functionalities, expand **Application Administration** and then [Data Transformation Pipeline Admin](#).

These features are described in this guide.

Downloading DTP Pipelines

The Download Pipelines feature enables you to download a version of an existing pipeline. The pipelines are saved in JSON format which can be imported into the application.

This is frequently used to download a pipeline from a non-production environment and then import the pipeline into a production environment.

1. Navigate to the DTP Pipeline Designer page.
2. Click the **Action** button  corresponding to the pipeline that you want to download.
3. Click **Download** .
4. Select **JSON**. The Download Pipeline pane displays.
5. Enter the version of the pipeline to download in the **Pipeline Version** field. You can only enter numerals here.

Note

This should be the same as the current version of the pipeline you are downloading.

6. Click **Download**.
7. Select a folder to save the pipeline.
8. Enter a file name, using the following naming convention: fileName . filenameformat : pipelineName_versionNumber.json
9. Click **Save**. The pipeline JSON is downloaded.

Importing DTP Pipelines

You can import a new JSON version of an existing Data Transformation pipeline. This is frequently done to import a pipeline from a non-production environment into a production environment.

Before importing a pipeline, you must first download the pipeline using the steps in [Downloading DTP Pipelines](#).

Note

You cannot import a pipeline that already exists in the environment. Before importing a pipeline, verify that this pipeline does not already exist in the environment. If so, [delete the pipeline](#) before importing the new version.

To import a pipeline, follow these steps:

1. Navigate to the DTP Pipeline Designer page.
2. Click **Import**. The Import pane opens.
3. Enter the required version of the pipeline to import in the **Version** field. You can also use the arrows to select the version number.

The Version number provided should be the same as that in the downloaded .JSON file.

4. Drag and Drop the .JSON file of the pipeline to import into the box. You can also click the **Drag and Drop** box to browse to the location of the .JSON file. The pipeline should have the following naming convention: fileName . filenameformat :
pipelineName_versionNumber.json

Note

You can only import the .JSON file of a DTP pipeline.

5. Click **Import**. The pipeline is imported and added to the My Pipelines list.

Copying DTP Pipelines

You can copy an existing pre-configured or user-configured pipeline and then configure it to meet your specific business requirements. This reduces the effort of creating and configuring a new pipeline when you only need to modify a few components of the pipeline.

Pre-configured pipelines that are included out-of the box cannot be modified. In order to customize these pipelines, you must first create a copy of those pipelines and save it as a new pipeline. You can then configure the newly copied pipeline according to your requirements. To copy a pipeline, follow these steps.

1. Navigate to the DTP Pipeline Designer page.
2. Click the **Action** button  corresponding to the pipeline that you want to copy.



3. Click **Copy**. The Copy Pipeline pane displays.
4. Provide the details as described in the following table.

Table 14-5 Fields for Copying Pipelines and their Descriptions

Field	Description
Name	Displays the name of the pipeline that you want to copy. You cannot modify this field.
Copy As	Enter the name for the new pipeline that you want to create by copying the existing pipeline. This field allows only the following characters: A-Z a-z 0-9 '-' For example, Abc-123 or Abc123.
Pipeline Focus	Displays the focus for the pipeline that you want to copy. This will be the focus for the new pipeline. You cannot modify this field.
Search Tags	Enter the keywords for the pipeline. These keywords can be used as search tags while searching for a pipeline. Search tags are also used to group pipelines of the same type. These search tags appear as filters in the dashboard.
Description	Enter the description for the pipeline.

5. Click **Save**. The pipeline is copied with all widgets and values provided in the original pipeline.

You can now modify these components, if desired. See [Editing DTP Pipelines](#) for more information.

Note

When editing a pipeline, consider the impact your changes may have on subsequent widgets in the pipeline.
For example, removing a mapped table or column in the Data Forge may impact how an existing Aggregator functions.

Add DTP Pipelines (MMG)

You can add a new pipeline.

To add a new pipeline, follow these steps:

1. Navigate to the DTP Pipeline Designer page.
2. Click **Add Pipeline**. The Add Pipeline pane opens.
3. Provide the details as described in the following table.

Table 14-6 Fields for Creating Pipelines and their Descriptions

Field	Description
Name	Enter the name for the pipeline. This field allows only the following characters: A-Z a-z 0-9 '-' For example, Abc-123 or Abc123.
Pipeline Type	Select Data Transformation from the Pipeline Type drop-down list.
Focus	Select the Focus for your pipeline from the drop-down list. For more information about Focuses, see Creating Focus Types .
Add Search Tags	Enter the keywords for the pipeline. These keywords can be used as search tags while searching for a pipeline. Search tags are also used to group pipelines of the same type. These search tags appear as filters in the Pipeline page.
Description	Enter the description for the pipeline.

- Click **Create**. A new pipeline is created and displayed in the Pipeline page. You can perform the required configurations in the newly created pipeline using the Hamburger menu or by right-clicking on the canvas to view available widgets.

For more information about widgets available in DTP pipelines, see [Widgets in Data Transformation Pipelines](#).

- Click **Save** to save the pipeline.

Editing DTP Pipelines

You can configure user-defined pipelines to better address your business needs as requirements change.

Note

Pre-configured pipelines cannot be edited. You can only edit user-defined pipelines.

To edit pipelines, follow these steps:

- Navigate to the DTP Pipeline Designer page.
- Click the pipeline name of the pipeline you want to edit. The DTP Pipeline Designer displays the widgets used in this pipeline.
- Perform the required configurations using the Hamburger menu or by right-clicking on the canvas to view available widgets.

For more information about widgets available in DTP pipelines, see [Widgets in Data Transformation Pipelines](#).

Note

When editing a pipeline, consider the impact your changes may have on subsequent widgets in the pipeline. For example, removing a mapped table or column in the Data Forge may impact how an existing Aggregator functions.

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

Editing Pipeline Descriptions

You can edit the description and search tags for user-created pipelines.

Note

Pre-configured pipelines cannot be edited. You can only edit user-configure pipelines

To edit the description and search tags, follow these steps:

1. Navigate to the DTP Pipeline Designer page.
2. Click the Action button  corresponding to the pipeline that you want to copy.
3. Click **Edit** . The Edit Pipeline pane displays.
4. Modify the required details.

Table 14-7 Fields for Editing Pipelines and their Descriptions

Field	Description
Name	Displays the name of the pipeline that you want to edit. You cannot modify this field.
Pipeline Focus	Displays the focus for the pipeline that you want to edit. This will be the focus for the new pipeline. You cannot modify this field.
Search Tags	Add or remove keywords for the pipeline. These keywords can be used as search tags while searching for a pipeline. Search tags are also used to group pipelines of the same type. These search tags appear as filters in the dashboard.
Description	Edit the description for the pipeline.

5. Click **Save**.

Deleting DTP Pipelines

You can delete a pipeline, if required for your implementation.

To delete a pipeline, follow these steps:

1. Navigate to the DTP Pipeline Designer page.
2. Click the **Action** button  corresponding to the pipeline that you want to delete.
3. Click **Delete** corresponding to the pipeline that you want to delete. A dialog box opens.
4. Enter a comment explaining why you are deleting the pipeline.
5. Click **Delete**. The selected pipeline is deleted.

Widgets in Data Transformation Pipelines

Depending on the pipeline type, specific widgets are available in the widgets pane of the pipeline. This topic describes the widgets in Data Transformation pipelines.

The following table describes the widgets available in Data Transformation pipelines.

Table 14-8 Data Transformation Pipeline – Widgets and Descriptions

Widget	Name	Description
	Calendar	Use this widget to add a Calendar, which allows you to designate and configure the schedule the scenario pipeline should be run according to.
	Data Forge	Use this widget to select the source(s) (Tables) which form the dataset to be further used in the pipeline.
	Business Filter	Use this widget to filter the data in the pipeline to create a subset of the data records which are available by applying the business filters. This allows you to search and analyze behaviors of interest.
	Data Transformation	Use this widget to write complex expressions and transformation in Python.
	Aggregator	Use this widget to define aggregators using available aggregations in the widget.
	Evaluation	Use this widget to configure an evaluation. Evaluations are used to define conditions for the aggregations that are defined in the aggregator. Relevant data (aggregators) is compared against these conditions using logical comparisons to identify the early signals of increasing risk exposures in various areas of an enterprise, resulting in event generation.
	Create Event	Use this widget to produce an event. This widget accepts inputs from evaluation and generates events based on evaluated output data from the evaluation widget.
	Using the Additional Information Widget	Use this widget to provide specific types of information related to entities involved in an event, which can help investigators by displaying relevant information about the entities involved, offering a more comprehensive view of the event.

Using the Calendar Widget

The Calendar widget allows you to designate and configure the calendar for the scenario's look back and frequency periods.

Before adding or configuring a calendar, you must first have created the pipeline.

1. In the Pipeline Designer page, select the pipeline you want to add or modify the calendar for. The Pipeline Canvas displays.

2. Right-click on the Canvas to display the list of widgets. Select **Calendar** . The Calendar widget is added to the pipeline Canvas. You can drag and drop the widget within the Canvas as needed.
3. Click **Options**  then click **Edit** . The Calendar pane displays.
4. Provide the details as described in the following table.

Table 14-9 Fields for Creating Calendars and their Descriptions

Field	Description
Name	Enter the name for the calendar. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123
Lookback Type	Select the Lookback Type used in this pipeline using the drop-down list. This is the range of days of data monitored in each run of the detection process. Options include Days or Months. This controls the unit in which both the lookback period and frequency period are expressed.
Days	Select either Calendar Days or Business Days from the drop-down list.
Weekdays	Select all the days which should be considered Week Days from the list.
Holiday Master	Select the applicable holidays which should be considered in this calendar. This section is populated by the data provided in the FCC_AM_HOLIDAY_MASTER table. For more about how this table is populated, see AMLHolidayMasterDataLoad Batch Details.

5. Click **Save**. A confirmation message displays.

Using the Data Forge Widget

The Data Forge widget allows you to select the source(s) (Tables) which form the dataset to be further used in the pipeline.

Before adding or configuring a Data Forge, you must first have created the pipeline.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Data Forge. The Pipeline Canvas displays.
2. To add a Data Forge to a Calendar, click **Add To** , and then select **Data Forge** .
 - Alternately, you can right-click on the Canvas to display the list of widgets, select **Data Forge** and associate the Data Forge with the Calendar.
 - You can also use the hamburger menu on the left side to display the Node Picker, which lists any externalized data forges already created. Drag and drop the data forge to the pipeline as desired.
3. Click **Options**  then click **Edit** . The Data Forge pane displays.
4. Under **Basic Configuration**, provide the following details:

- a. Enter the **Name** for the Data Forge. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123
- b. Select the **Type** from the drop-down list.
- c. The **Externalize** selector allows you to externalize this data forge so that it can be used in additional pipelines. To access all externalized data forges, click the hamburger menu on the left side to display the Node Picker, which lists any externalized data forges already created.
- d. Select one or more **Source Tables** using the drop-down list. Use the search bar in the list to find specific tables easily.

To remove a source table from the Data Forge, click **Remove**  beside the table name in the Source Tables field.

- e. The **Connected Data Forges** lists any data forges which are associated with this data forge.
- f. You can add a Runtime Parameter to the Data Forge by clicking **Add Run Time**

 **Run Time Parameter**

Parameter . For more information about how to create run time parameters, see [Creating Runtime Parameters](#).

5. The **Advanced Configuration** tab displays any Joins and Filters associated with this Data Forge.

Note

At least two sources (tables/dataforges) must be selected or connected for the join section to be enabled in the **Advance Configuration** tab.

- Expand **Joins** to view or create Joins associated with this Data Forge. The Output box provides a summary of all current Joins, with the Join configurations listed in detail below. You can add new Joins using the steps in [Creating Joins](#).
 - Expand **Filters** to view or create Filters associated with this Data Forge. The Output box provides a summary of all current Filters, with the Filter configurations listed in detail below. You can add new Filters using the steps in [Creating Filters](#).
6. Use the **Mapped Columns** tab to view and map columns within the Data Forge. To map additional columns, follow these steps:
 - a. In the Mapped Columns tab, select a Source Table/Dataforge from the drop-down list. The available columns in this table display.
 - b. Select a Column from the list. Use the search bar in the list to find specific tables easily.



- c. Click **Map**  to map the column. The column displays in the Mapped Columns list. You can edit the Mapped Column Name, if desired.



- d. Alternately, click **Add Expression**  to open the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To use the Expression Builder, follow these steps:

- a. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- b. Click **Save** to save the changes.

7. When you have finished configuring your Data Forge, click **Save**.

Creating Joins

Joins allow you to combine or group multiple tables using various join operators.



1. In the Advanced Configuration tab, click **Add Join**. A new Join row is added to the list.
2. Use the **Include** selector to indicate whether or not this Join should be included.
3. Select a **Join Operator** from the drop-down list.

The following types of join operators are available:

- **Inner Join:** The Inner Join selects all rows from both participating tables as long as there is a match between the columns.
 - **Left Join:** The Left Join returns all rows from the left table, with the matching rows in the right table.
 - **Right Join:** The Right Join returns all rows from the right table, with the matching rows in the left table.
 - **Full Join:** The Full Join combines the results of both the left and right outer joins and returns all rows from the tables on both sides
4. Add a Join condition to the Join table by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Group** and then **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- iii. Click **Save** to save the changes.

- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.

- ii. Select a Column for the left-hand side from the drop-down list.
- iii. Select an Operator from the drop-down list.
- iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
- v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Filters

You can configure a filter by defining various filter conditions.

1. In the Advanced Configuration tab, click **Add Group** . A new Filter Group row is added to the list.
2. Use the **Include** selector to indicate whether or not this Filter should be included.
3. Add a Filter condition by following these steps. You can add multiple groups and multiple conditions under each group.

- a. Click **Add Condition** .
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.
 - iii. Click **Save**  to save the changes.
 - **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.

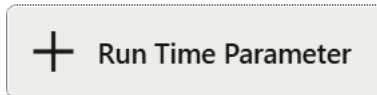
- iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
- v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:



1. Click **Add Runtime Parameter** . The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-10 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Using the Business Filter Widget

The Business Filter widget to filter the data in the pipeline to create a subset of the data records which are available by applying the business filters. This allows you to search and analyze behaviors of interest

Before adding or configuring a Business Filter, you must first have created the pipeline and added either a Data Forge or Data Transformation.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Business Filter. The Pipeline Canvas displays.

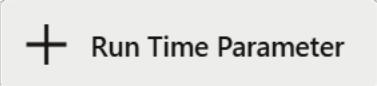
- To add a Business Filter to a Data Forge, click **Add To** , and then select **Business Filter** . Alternately, you can right-click on the Canvas to display the list of widgets, select **Business Filter** and associate the Business Filter with the Data Forge.

 Note

Business Filters must be associated with a Data Forge.

- Click **Options**  then click **Edit** . The Business Filter pane displays.
- Under **Basic Configuration**, provide the following details:
 - Enter the **Name** for the business filter. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123
 - Connected Sources** lists the name of the connected Data Forge or Data Transformation.
- Click **List of Filters** to add a Business Filter which has already been created for columns in your data set. Use the selectors to select one or more filters as needed.
 You can delete Business Filters from the List of Filters tab. Click **Delete** corresponding to the business filter that you want to delete. A dialog box opens. Click **Delete**.
 The system verifies that this business filter is not in use by any pipeline. If this filter is enabled in any pipeline, an error message displays and the business filter can not be deleted. Otherwise, the selected filter is deleted.
- The **Questionnaire** tab lists the mapped filters, which are the filters you selected in the previous step. Based on the Filter Type, you can further refine the filter.
 - For Include-Exclude types, use **Include** and **Exclude** to include or exclude one or more specific fields. For example, Include Retail Banking.
 - For Yes-No types, use **Yes** and **No** to choose whether to enable this filter.
 - For Filter types, use the drop-down lists to select the operators and code values. Select **Parameter** if you want to filter based on one or more specific fields.
 - Use **Add Filter** to add a new business filter to the List of Filters.. For more information about how to add a filter, see [Adding New Business Filters](#)
 - You can add additional conditions.
 - You can add a Runtime Parameter to the Business Filter by clicking **Add Run Time**

 **Run Time Parameter**

Parameter . For more information about how to create run time parameters, see [Creating Runtime Parameters](#).

- Click **Save**. A confirmation message displays.

Adding New Business Filters

You can create new business filters as needed to meet your business requirements.

- In the Business Filter pane, click **Add Filter**.
- Provide the details as described in the following table.

Table 14-11 Fields for Creating Business Filters and their Descriptions

Field	Description
Name	Enter the Name for the business filter. This field allows only the following characters: A-Z a-z 0-9 '_' For example, Abc_123 or Abc 123
Entities	Select which data set to create the filter from using the Entities drop-down list.
Attribute	Select the column on which you will be filtering the data from the Attribute drop-down list. All columns that were mapped in the Data Forge or Columns resulting from the connected Data Transformation widget will be displayed.
Filter Type	Select from the following options: • Include-Exclude: Use the Input Codes section to select the input which should be included or excluded. Use the Input Codes section to feed the inputs in the form of CODES and logical Names. (for example, IND-Individual). • Yes-No • Filter

3. Add the filter, using one of the following options. A confirmation message displays.
 - Click **Add** to add this Business Filter to the List of Filters.
 - Click **Add & Attach** to add this Business Filter to the List of Filters and attach it to the current data set.
4. When you have finished creating filter, click **Save**.

 Note

Oracle recommends creating the following commonly used Business Filters in order to simplify future pipeline configuration.

Table 14-12 Common Business Filters

Business Filter	Purpose
Account Business Type	Indicates which types of functional areas in which the account does business should be included when detecting behaviors of interest.
Account Holder Type	Indicates which types of internal or external customers should be included when detecting behaviors of interest.
Transaction Type	Indicates which transaction types are included when detecting behaviors of interest.
Transaction Purpose	Indicates which transaction purposes should be included when detecting behaviors of interest.
Party Role	Indicates which party roles are included when detecting behaviors of interest.

Creating Custom Filters

The Custom Filter allows you to create business filters using the Expression Builder.

You can form filter conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:

1. In the Business Filter pane, click **Custom Filter**.
2. Select the required Dataset, Attribute, and operators to create the filter. The resulting condition is displayed in the **Condition Value** box.
3. Click **Save**  to create this filter
4. Create additional filters, as desired. When you have finished creating all the filters needed for this pipeline, click **Save** to save the changes.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:

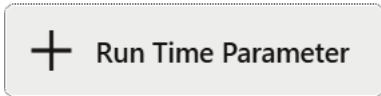
1. Click **Add Runtime Parameter** . The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-13 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Using the Data Transformation Widget

The Data Transformation widget allows you to write and import complex expression and transformation in Python and Java.

To use this widget effectively, you must have a working knowledge of scripting and Java or Python.

1. Create a pipeline, as shown in [Adding DTP Pipelines](#).
2. Add a Data Transformation widget to the pipeline by right-clicking the Canvas to display

the list of widgets and selecting **Data Transformation** .

The Data Transformation widget can be stand-alone or associated with another widget, such as Business Filter, Evaluation, or Aggregator.

3. Click **Options**  then click **Edit** . The Data Transformation pane displays.
4. In Basic Configuration, provide the following details:
 - a. Enter a **Name** for this Data Transformation.
 - b. Select a **Runtime Parameter** from the drop-down list. To add a new Runtime Parameter, see [Creating Runtime Parameters](#).
 - c. For Data Transformations associated with another widget, map the columns in **Column Definition** using **Add Columns**. The **View Input Table Columns** shows which columns were mapped in the connected Data Forge.
5. The **Define Transformation** tab displays the script options.
 - **Standard**
 - a. Select the **Type** from the drop-down list.
 - b. Select the **Transform** from the drop-down list.
 - c. Select **Interpreters** from the drop-down list. Python allows you call a custom python script.
 - i. In the **Script** section, provide a description.
 - ii. Enter or paste the python script you want to execute.

Note

If this is a script you intend to reuse, you can create a template by clicking **Templates**, then **Add**. Enter the pertinent details about the script, enter or paste the script into the box, and click **Save**. Existing templates can be modified by clicking **Edit**. Editing a template updates the script in every pipeline which uses this template.

To use an existing template, select **Use Template** and select the desired template from the **Script** drop-down list.

- iii. Click **Save**.
- Selecting **Custom** allows you to create a notebook.

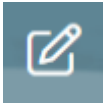
Notebooks allow you to add and run paragraphs, invalidate session, edit, add, export the notebook, and so on. To understand the icons which display in the Notebook, see [Common Screen Elements](#).

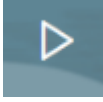
- a. Click **Add Python Paragraph** to add a paragraph in the notebook and fetch the runtime parameters. This is available for all the interpreters.
 - b. Add paragraphs to the Notebook as desired.
 - c. Click **Play** to run the script and display the label.
6. You can import an existing notebook by selecting **Import Notebook**.
- Click **Import Archive File** to select the .dsnb file from the file selector dialog and select a file. Click **Open**.
 - OR
 - Drag and drop the file into the **Import Archive File** field.

Common Screen Elements in the Notebook

This topic describes common screen elements in the notebook that can be used to perform quick actions when preparing and executing the notebooks.

Elements in the Notebook

Icon Name	Action/Description
Modify Notebook 	Modifies the details of a notebook, such as a name, description, and (or) tags.
Hide Code 	Hides or shows the Code Section in all the paragraphs in a notebook.
Hide Result 	Hides or shows the Results Section in all the paragraphs in a notebook.
Re-initialize session 	Re-initialize the paragraph in a notebook. Note: By default, the Start Widget gets executed automatically on clicking the Re-initialize icon. All other paragraphs need to be executed manually.

Icon Name	Action/Description
Read-Only 	Sets the notebook to read-only mode. Note: The notebook is protected from edit, clear result, delete, share, reset session, and run paragraphs in Read-only mode.
Write 	Set the notebook to write mode.
Run Paragraphs 	Executes all the paragraphs in a notebook in sequential order.
Invalidate Session 	Resets any connection or code executed in a notebook.
Delete Notebook 	To delete the selected notebook.
Clear Result 	Clears results for all the paragraphs in a notebook. Note: This action clears all the results. You must rerun the paragraphs to view the results.
Clear Paragraph Dependencies 	Remove all defined paragraph dependencies.
Open as an iframe 	Opens a notebook in an Iframe. This allows a notebook to be embedded inside another webpage.

Icon Name	Action/Description
Share Notebook 	Share the notebook with another user, user group, or role.
Clone Notebook 	Creates a copy of a notebook. All paragraphs in the current notebook are replicated in the new notebook. The cloned notebook is created with the default name, Copy of <Current Notebook Name>.
Export Notebook 	Export the notebook to your computer as a DNSB file.
Print Notebook 	To print a notebook in the PDF format and save it in your local machine.
Default Template 	To apply the overall look and feel of the notebook using the default template.
Layout 	Sets the preferred layout. The available layouts are Zeppelin and Jupyter.
Default Template 	Applies the overall look and feel of the notebook using the default template.
Show Panel 	Shows or hides the Paragraph Settings Bar Commands, Results Toolbar, and Settings Dialog for a selected paragraph in a panel to the notebook's right.

Icon Name	Action/Description
Attach Credentials 	You can use this option to attach credentials (wallet and password) to the notebook to enable secure data access.
Versioning 	You can use this option to create versions for your notebook, which helps you analyze the changes based on the version control.
Run Paragraph 	To execute the code or query in a paragraph.
Enter Dependency Mode 	To add or remove dependent paragraphs. Paragraphs with dependent paragraphs are executed in the dependency order.
Comments 	To add comments to a paragraph.
Expand 	To expand a paragraph and view the paragraph in full-screen mode.
Show/Hide Line Numbers 	To show or hide line numbers in the code in a paragraph. Note: This button is applicable only to the code section.
Visibility 	To manage the visibility settings in a paragraph. It controls how a paragraph may be viewed by the author and other users who have access to the notebook.

Icon Name	Action/Description
Settings 	Users can perform the following actions: • Resize the width of a paragraph • Change the order of placement of the paragraphs by moving them up or down • Run the all above or all below paragraph from the selected paragraph. • Clear the paragraph result • Open the notebook in Embedded window • Disable the run action • Clone the paragraph • Delete a paragraph

Using the Aggregator Widget

Relevant data is compared against aggregators to identify the early signals of increasing risk exposures in various areas of an enterprise.

Before adding or configuring an Aggregator, you must first have created the pipeline, data forge, and business filter.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Aggregator. The Pipeline Canvas displays.

2. To add an Aggregator to a Business Filter, click **Add To** , and then select **Aggregator**.

 Alternately, you can right-click on the Canvas to display the list of widgets, select **Aggregator** and associate the Aggregator with the Business Filter.

An aggregator can be associated to another aggregator, if desired, to further refine the aggregated data. The fields will be populated only by data provided in the parent aggregator. When two aggregators are linked to another aggregator a new tab, **Aggregator Joins**, displays. For information on how to create aggregator joins, see [Creating Aggregator Joins](#).

3. Click **Options**  then click **Edit** . The Aggregator pane displays.
4. Under **Basic Configuration**, enter the **Name** for the aggregator. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123s:
5. Click **List of Aggregators** to add an Aggregator which has already been created in your data set. Use the selectors to select one or more filters as needed. Use **View** to view the Aggregator Details.
6. The **Aggregators** tab lists the mapped aggregators, which are the aggregators you selected in the previous step. Click **Edit** to further define the aggregator.
 - Clicking **Edit** on a standard aggregator displays the add/edit aggregator tab, and the aggregator can be edited.
 - Clicking **Edit** on a custom aggregator opens the **Custom Aggregator**, which allows you to form aggregator conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text conditions. To define a condition using the Expression Builder, see [Creating Custom Aggregators](#).
7. Click **Save**. A confirmation message displays.

Adding and Editing Aggregators

You can create new aggregators or edit existing aggregators as needed to meet your business requirements.

1. Select the **Aggregator** from the drop-down list. This list contains the types of aggregation which are available for this Business Filter, such as Sum, Count or Average Monthly.
2. Enter a **Group By** clause. This allows you to group aggregators by the specified attribute across the pipeline. Grouping by a certain attribute will be applied to all aggregators used in the pipeline.
3. Select the **Attribute** from the drop-down list. This list contains the data forge tables and mapped columns.
4. Select the required **Look Back** option and provide the required Value details. You can further configure the Look Back period for the aggregator when building the expression.
5. Select the **Filter Attribute**.
6. Select the **Filter Operator**.
7. Select the required **Filter Value** option and provide the required details.
8. Add the aggregator using one of the following options. A confirmation message displays.
 - Click **Add** to add this aggregator to the List of Aggregators.
 - Click **Add & Attach** to add this aggregator to the List of Aggregators and attach it to the current data set.
9. When you have finished creating aggregators, click **Save**.

Creating Custom Aggregators

The Custom Aggregator option allows you to create aggregators using the Expression Builder.

You can form aggregator conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text aggregator conditions. To define an aggregator condition using the Expression Builder, follow these steps:

1. In the Aggregator pane, click **Custom Aggregator**.
2. Enter the **Name** for the Aggregator. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123.
3. Select the required Attributes and Operators to create the Aggregator. The resulting condition is displayed in the **Condition Value** box.
4. Select the required **Look Back** option and provide the required Value details. You can further configure the Look Back period for the aggregator when building the expression.
5. Add the aggregator using one of the following options. A confirmation message displays.
 - Click **Add** to add this aggregator to the List of Aggregators.
 - Click **Add & Attach** to add this aggregator to the List of Aggregators and attach it to the current data set.
6. When you have finished creating aggregators, click **Save**.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:



1. Click **Add Runtime Parameter**. The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-14 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Creating Aggregator Joins

When two aggregators are linked to another aggregator you should create Aggregator Joins.

This tab will only display when two parent aggregators are associated with a child aggregator.



1. In the Aggregator Joins tab of the Aggregator pane, click **Add Join**. A new Join row is added to the list.
2. Use the **Include** selector to indicate whether or not this Join should be included in this aggregator.
3. Select a **Join Operator** from the drop-down list.

The following types of join operators are available:

- **Inner Join:** The Inner Join selects all rows from both participating tables as long as there is a match between the columns.
- **Left Join:** The Left Join returns all rows from the left table, with the matching rows in the right table.
- **Right Join:** The Right Join returns all rows from the right table, with the matching rows in the left table.

- **Full Join:** The Full Join combines the results of both the left and right outer joins and returns all rows from the tables on both sides
4. Add a Join condition to the Join table by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Group** and then **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.
 - iii. Click **Save**  to save the changes.
 - **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.
 - iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
 - **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Using the Evaluation Widget

Evaluations are used to define conditions for the measures that are defined in the aggregator. Evaluations perform logical comparisons against these conditions to generate events.

Before adding or configuring an Evaluation, you must first have created the pipeline, data forge, business filter, and aggregator.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Evaluation. The Pipeline Canvas displays.

2. To add an Evaluation to an Aggregator, click **Add To** , and then select **Evaluation** . Alternately, you can right-click on the Canvas to display the list of widgets, select **Evaluation** and associate the Evaluation with the Aggregator.
3. Click **Options**  then click **Edit** . The Evaluation pane displays.
4. Under **Basic Configuration**, enter the **Name** for the Evaluation. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123s:
5. The **Advanced Configuration** tab displays the detection logic and event generation criteria. Event generation criteria are shown in formula-style expressions in the Output box.
6. You can add multiple groups and multiple conditions under each group. Add conditions to the Group by clicking **Add Group** and then **Add Condition** .
7. Select the required Aggregator, operators, and conditions to build the evaluation. You can define the evaluation conditions using one of the following:
 - Expression Builder: You can form filter conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text conditions. To define a condition using the Expression Builder, follow these steps:
 - a. Click **Exp**. The Expression Builder dialog box is displayed.
 - b. Select the required Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition Value field.
 - c. Click **Save** to save the changes.
 - Text: You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - a. Click **Text**.
 - b. Select an aggregator and operator, and then enter the text in the field on the right-hand side.
 - c. Click **Save** to save the changes.
8. Use the Include selector to choose whether to include the condition group in this Evaluation.
9. Click **Save**. A confirmation message displays.

Using the Create Event Widget

This widget accepts inputs from evaluation and generates events based on evaluated output data from the evaluation widget

In Scenario Pipelines, the Create Event widget is the final part of the pipeline and is used to produce an event. An event is a record of one or more pattern matches in a detection run, which is a signal for further investigation.

The Create Event widget can only be attached to an Evaluation widget. Multiple Evaluation widgets can be connected to one Create Event widget.

To create an event, follow these steps:

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Create Event widget. The Pipeline Canvas displays.
2. To add Create Event to an Evaluation, click **Add To** , and then select **Create Event** . Alternately, you can right-click on the Canvas to display the list of widgets, select **Create Event** and associate the Create Event widget with the Evaluation.
3. Click **Options**  then click **Edit** . The Create Event pane displays.
4. Under **Basic Configuration**, select the **Event Type** from the drop-down list.
You can add additional Event Types, using the steps in [Creating Event Types](#).
You must define the Focus associated with events generated from this widget.
 - Select the required **Data Forge** from the drop-down list. The values which display are based on the tables you mapped in the Data Forge widget.
 - Select the required **Focus Column** from the drop-down list. The values which display are based on the columns you mapped in the Data Forge widget.

Note

You can only select one value in each drop-down list. The column selected should be a column which is unique to the table.

- Use the **Bind IDs Configuration** tab to choose which types of information you want to display in the generated events. For more information about this tab, see [Configuring Bind IDs](#).
5. Click **List of Highlights** to add Highlights which have already been created in your data set. This tab displays all relevant highlights from the aggregator. Use the selectors to select one or more highlights as needed.
 6. The **Highlights** tab displays any Highlights associated with this pipeline.
Highlights include the most salient facts associated with a match or alert, and are intended to aid with the understanding and possible disposition of an event. This tab includes the following details:
 - Highlight Name: This name will remain constant across the application.
 - New Highlight Name: A unique name for the highlight which will be used in the UI. This name can be changed as needed to meet your business requirements.
 - Column : The name of the column used to aggregate the highlight.
 You can add a new highlight by clicking **Add**, and following these steps:
 - a. Select the **Attribute** from the drop-down list.
 - b. Enter a **Name** for this highlight.
 - c. Click **Save**.
 7. The **Bind IDs Configuration** tab displays the tables and columns which you have mapped in the Data Forge widget. Use the selector to choose the Bind IDs you are associating with the Focus of the events generated by this widget.
 8. When you have finished configuring the Create Event widget, click **Save**.

Configuring Bind IDs

Bind IDs are used to define the focal entity by selecting the data on which the analysis is centered.

These bindings determine the types of information which display in the generated events. To configure Bind IDs, follow these steps:

1. Click to open the **Bind IDs Configuration** tab in the Create Event pane of the Create Event widget. The Bind IDs Configuration tab displays all tables and columns which were mapped in the data forge for this pipeline.
2. Use the Bind ID selector to choose which types of information you want to display in the generated events. For example, to display Customer information in your events, select the **Customer Internal ID** column from the Customer Dimension table as your Bind ID.

Note

Bind IDs selected should be a column which is unique to the table.

The following table provides examples of some commonly selected Bind IDs.

Table 14-15 Bind ID Examples

Focal Entity	Table Name	Column Name
Account	Account Dimension	Account Internal Identifier
Customer	Customer Dimension	Customer Internal Identifier
Transaction	Transactions	Transaction Internal ID
Account Profile	Account Summary	

3. Click **Save**.

Using the Additional Information Widget

The Additional Information widget allows you to provide additional context not considered by the scenario that will aid in investigating events generated by the Create Event widget in DTP scenario pipelines.

Oracle FCCM clients provide many types of data in Business tables which may not be required for a particular scenario pipeline to successfully generate an event. This information can provide essential context which will assist investigators as they disposition the events and cases. By adding one or more Additional Information widget, you can include this information in the event, so that it is available for compliance officers and investigators to consider.

The Additional Information widget can only be attached to a Create Event widget. Multiple Additional Information widgets can be connected to one Create Event widget.

To add an Additional Information widget, follow these steps:

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Additional Information widget. The Pipeline Canvas displays.
2. To add Additional Information to a Create Event widget, click **Add To**, and then select **Additional Information**.

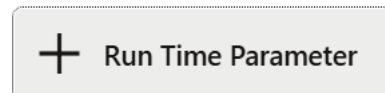
Alternately, you can right-click on the Canvas to display the list of widgets, select **Additional Information** and associate the widget with the Create Event widget.

3. Click **Options** then click **Edit**. The Additional Information pane displays.
4. Under **Basic Configuration**, provide the following details:
 - a. Enter the **Name** for the Additional Information widget. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123
 - b. Select the **Source Entity Type** from the drop-down list.
 - c. Select the **Target Entity Type** such as Customer, External Entity, Account Summary, and so on, from the drop-down list.
 - d. Select one or more **Source Tables** using the drop-down list. Use the search bar in the list to find specific tables easily.

The values which display in the Source Tables box are based on the tables you mapped in the Data Forge widget.

To remove a source table, click **Remove**  beside the table name in the Source Tables field.

- e. You can add a Runtime Parameter by clicking **Add Run Time Parameter**



. For more information about how to create run time parameters, see [Creating Runtime Parameters](#).

5. The **Advanced Configuration** tab displays any Joins and Filters associated with this widget

Note

At least two sources (tables/data forges) must be selected or connected for the join section to be enabled in the **Advance Configuration** tab.

- Expand **Joins** to view or create Joins associated with this widget. The Output box provides a summary of all current Joins, with the Join configurations listed in detail below. You can add new Joins using the steps in [Creating Joins](#).
 - Expand **Filters** to view or create Filters associated with this widget. The Output box provides a summary of all current Filters, with the Filter configurations listed in detail below. You can add new Filters using the steps in [Creating Filters](#).
6. Use the **Mapped Columns** tab to view and map columns within the widget. To map additional columns, follow these steps:

Note

Mapped columns must be relevant from both the Source and Target tables.

- a. In the Mapped Columns tab, select a Source Table/Data forge from the drop-down list. The available columns in this table display.
- b. Select a Column from the list. Use the search bar in the list to find specific tables easily.



- c. Click **Map** to map the column. The column displays in the Mapped Columns list. You can edit the Mapped Column Name, if desired.



- d. Alternately, click **Add Expression** to open the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To use the Expression Builder, follow these steps:

- a. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.

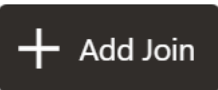


- b. Click **Save** to save the changes.

7. When you have finished configuring your widget, click **Save**

Creating Joins

Joins allow you to combine or group multiple tables using various join operators.



1. In the Advanced Configuration tab, click **Add Join**. A new Join row is added to the list.
2. Use the **Include** selector to indicate whether or not this Join should be included.
3. Select a **Join Operator** from the drop-down list.

The following types of join operators are available:

- **Inner Join:** The Inner Join selects all rows from both participating tables as long as there is a match between the columns.
 - **Left Join:** The Left Join returns all rows from the left table, with the matching rows in the right table.
 - **Right Join:** The Right Join returns all rows from the right table, with the matching rows in the left table.
 - **Full Join:** The Full Join combines the results of both the left and right outer joins and returns all rows from the tables on both sides
4. Add a Join condition to the Join table by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Group** and then **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.

- ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- iii. Click **Save** to save the changes.

- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.
 - iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Filters

You can configure a filter by defining various filter conditions.



1. In the Advanced Configuration tab, click **Add Group**. A new Filter Group row is added to the list.
2. Use the **Include** selector to indicate whether or not this Filter should be included.
3. Add a Filter condition by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



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

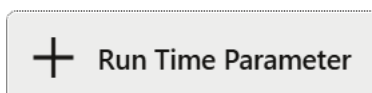
- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.
 - iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:



1. Click **Add Runtime Parameter** . The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-16 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Managing Data Pipelines

Data pipelines 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 tables of data, adding derived attributes to data, running derivations on the data to determine the risk associated with the entity, and so on.

Widgets in Data Pipelines

The following table describes the widgets available in Data pipelines.

Table 14-17 Data Pipeline – Widgets and Descriptions

Widget	Name	Description
Figure 14-13 Persist Widget 	Using the Persist Widget	Use this widget to write data to database tables so that it can be used in other pipelines.
	Using the Data Forge Widget	Use this widget to select the source(s) (Tables) which form the dataset to be further used in the pipeline.
	Using the Data Transformation Widget	Use this widget to write complex expressions and transformation in Python.

Using the Persist Widget

Use this widget to write data to database tables so that it can be used in other pipelines.

Before adding or configuring Persist, ensure that the Persist widget is properly connected to a Data Forge on the Pipeline Canvas; otherwise, it will not function as expected.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Persist. The Pipeline Canvas displays.

2. To add a Persist to a Data Forge, click **Add To** , and then select **Persist**. 
 - Alternately, you can right-click on the Canvas to display the list of widgets, select **Persist**, and associate the Persist with the Data Forge.
 - You can also use the hamburger menu on the left side to display the Node Picker, which lists any externalized data forges already created. Drag and drop the data forge to the pipeline as desired.

3. Click **Options**  then click **Edit** . The Data Forge pane displays.

4. Under **Basic Configuration**, provide the following details:

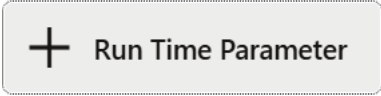
Ensure that the Persist widget is connected to a Data Forge on the Pipeline Canvas. The Persist widget will not function unless it is associated with a Data Forge.

- a. In the **Name** field, enter a user-defined name for the Persist widget. (This field is not auto-populated.)
- b. Select the **Target Table** from the drop-down list.
- c. The **Source Data Forges** field displays the Data Forge selected in the Data Forge widget. This field is system-generated and disabled.

Note

This value is inherited from the associated Data Forge and cannot be modified in the Persist widget.

- d. You can add a Runtime Parameter to the Data Forge by clicking **Add Run Time**

 **Run Time Parameter**

Parameter. For more information about how to create run time parameters, see [Creating Runtime Parameters](#).

5. The **Advanced Configuration** tab displays any Joins and Filters associated with this Data Forge.

Note

At least two sources (dataforges) must be selected or connected for the join section to be enabled in the **Advance Configuration** tab.

- Expand **Joins** to view or create Joins associated with this Data Forge. The Output box provides a summary of all current Joins, with the Join configurations listed in detail below. You can add new Joins using the steps in [.Creating Joins](#)
 - Expand **Filters** to view or create Filters associated with this Data Forge. The Output box provides a summary of all current Filters, with the Filter configurations listed in detail below. You can add new Filters using the steps in [Creating Filters](#).
6. Use the **Map** tab to view and map columns within the Persist widget. To map columns, follow these steps:
 - a. Click the **Map** tab.
 - b. Under **Type**, select one of the following options:
 - **Incremental:** Maps data from the source dataset to the target table incrementally. Only new records are inserted during each run. For example, if 100 records are loaded on Day 1 and 110 records exist on Day 2, only the additional 10 new records are inserted into the target table.
 - **Full Load:** Inserts all filtered data into the target table along with Processing Date and Data Origin values during each run.
 - c. The **Select a Data Forge** field displays a drop-down list of the Data Forges connected to the Persist widget. Select the required Data Forge from the list to define the source dataset for column mapping.
 - d. The **Target Table** field displays the target table selected under Basic Configuration.

e. Do either of the following:

- If you select **Incremental** as Type, do the following:
 - i. From the **Source Column** list, select the required column. Use the search bar to locate specific columns.
 - ii. Select the corresponding Target Column.
 - iii. In the **Selected Type** dialog box, select either Unique or Direct.



- iv. Click **Map** to map the column. The column displays in the Mapped Columns list. You can edit the Mapped Column Name, if desired.
 - v. The mapped column is displayed in the mapped columns list.
 - vi. Repeat the steps to map additional columns, if required.
- If you select **Full Load** as Type, do the following:
 - i. From the Source Column list, select the required column.
 - ii. Select the corresponding Target Column.



- iii. Click **Map** to map the column. The column displays in the Mapped Columns list. You can edit the Mapped Column Name, if desired.

Note

In Full Load, all filtered data is inserted into the target table during each run along with Processing Date and Data Origin.



- f. Alternately, click **Add Expression** to open the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To use the Expression Builder, follow these steps:
 - a. Select the required Data Forge, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- b. Click **Save** to save the changes.

7. When you have finished configuring your Persist, click **Save**.

Note

If a batch fails or gets stuck, run the PurgeAMIngestion batch. While running the batch, edit the BatchRunId and pass the corresponding DTPIngestion batch run ID.

Creating Joins

Joins allow you to combine or group multiple tables using various join operators.



1. In the Advanced Configuration tab, click **Add Join**. A new Join row is added to the list.
2. Use the **Include** selector to indicate whether or not this Join should be included.
3. Select a **Join Operator** from the drop-down list.

The following types of join operators are available:

- **Inner Join:** The Inner Join selects all rows from both participating tables as long as there is a match between the columns.
 - **Left Join:** The Left Join returns all rows from the left table, with the matching rows in the right table.
 - **Right Join:** The Right Join returns all rows from the right table, with the matching rows in the left table.
 - **Full Join:** The Full Join combines the results of both the left and right outer joins and returns all rows from the tables on both sides
4. Add a Join condition to the Join table by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Group** and then **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- iii. Click **Save** to save the changes.
- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.
 - iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.

- ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
- iii. Select a Column for the left-hand side from the drop-down list.
- iv. Select an Operator from the drop-down list.
- v. Enter the text in the field on the right-hand side.

Creating Filters

You can configure a filter by defining various filter conditions.

1. In the Advanced Configuration tab, click **Add Group** . A new Filter Group row is added to the list.
2. Use the **Include** selector to indicate whether or not this Filter should be included.
3. Add a Filter condition by following these steps. You can add multiple groups and multiple conditions under each group.

- a. Click **Add Condition** .

- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp.** The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.

- iii. Click **Save**  to save the changes.

- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.
 - iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.

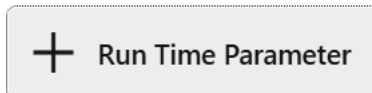
- iv. Select an Operator from the drop-down list.
- v. Enter the text in the field on the right-hand side.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:



1. Click **Add Runtime Parameter**  . The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-18 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Using the Data Forge Widget

The Data Forge widget allows you to select the source(s) (Tables) which form the dataset to be further used in the pipeline.

Before adding or configuring a Data Forge, you must first have created the pipeline.

1. In the Pipeline Designer page, select the pipeline where you want to add or modify the Data Forge. The Pipeline Canvas displays.
2. Click **Add To** and then select **Data Forge**  , and then select **Data Forge**  .
 - Alternately, you can right-click on the Canvas to display the list of widgets, and then select Data Forge.
 - You can also use the hamburger menu on the left side to display the Node Picker, which lists any externalized data forges already created. Drag and drop the data forge to the pipeline as desired.
3. Click **Options**  then click **Edit**  . The Data Forge pane displays.
4. Under **Basic Configuration**, provide the following details:

- a. Enter the **Name** for the Data Forge. This field allows only the following characters: A-Z a-z 0-9 ' _ ' For example, Abc_123 or Abc 123
- b. Select the **Type** from the drop-down list.
- c. The **Externalize** selector allows you to externalize this data forge so that it can be used in additional pipelines. To access all externalized data forges, click the hamburger menu on the left side to display the Node Picker, which lists any externalized data forges already created.
- d. Select one or more **Source Tables** using the drop-down list. Use the search bar in the list to find specific tables easily.

To remove a source table from the Data Forge, click **Remove**  beside the table name in the Source Tables field.

- e. The **Connected Data Forges** lists any data forges which are associated with this data forge.
- f. You can add a Runtime Parameter to the Data Forge by clicking **Add Run Time**

 **Run Time Parameter**

Parameter . For more information about how to create run time parameters, see [Creating Runtime Parameters](#).

5. The **Advanced Configuration** tab displays any Joins and Filters associated with this Data Forge.

Note

At least two sources (tables/dataforges) must be selected or connected for the join section to be enabled in the **Advance Configuration** tab.

- Expand **Joins** to view or create Joins associated with this Data Forge. The Output box provides a summary of all current Joins, with the Join configurations listed in detail below. You can add new Joins using the steps in [Creating Joins](#).
 - Expand **Filters** to view or create Filters associated with this Data Forge. The Output box provides a summary of all current Filters, with the Filter configurations listed in detail below. You can add new Filters using the steps in [Creating Filters](#).
6. Use the **Mapped Columns** tab to view and map columns within the Data Forge. To map additional columns, follow these steps:
 - a. In the Mapped Columns tab, select a Source Table/Dataforge from the drop-down list. The available columns in this table display.
 - b. Select a Column from the list. Use the search bar in the list to find specific tables easily.



- c. Click **Map**  to map the column. The column displays in the Mapped Columns list. You can edit the Mapped Column Name, if desired.



- d. Alternately, click **Add Expression**  to open the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To use the Expression Builder, follow these steps:

- a. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- b. Click **Save** to save the changes.
7. When you have finished configuring your Data Forge, click **Save**.

Creating Joins

Joins allow you to combine or group multiple tables using various join operators.



1. In the Advanced Configuration tab, click **Add Join**. A new Join row is added to the list.
2. Use the **Include** selector to indicate whether or not this Join should be included.
3. Select a **Join Operator** from the drop-down list.

The following types of join operators are available:

- **Inner Join:** The Inner Join selects all rows from both participating tables as long as there is a match between the columns.
 - **Left Join:** The Left Join returns all rows from the left table, with the matching rows in the right table.
 - **Right Join:** The Right Join returns all rows from the right table, with the matching rows in the left table.
 - **Full Join:** The Full Join combines the results of both the left and right outer joins and returns all rows from the tables on both sides
4. Add a Join condition to the Join table by following these steps. You can add multiple groups and multiple conditions under each group.



- a. Click **Add Group** and then **Add Condition**.
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.



- iii. Click **Save** to save the changes.
- **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.

- ii. Select a Column for the left-hand side from the drop-down list.
- iii. Select an Operator from the drop-down list.
- iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
- v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Filters

You can configure a filter by defining various filter conditions.

1. In the Advanced Configuration tab, click **Add Group** . A new Filter Group row is added to the list.
2. Use the **Include** selector to indicate whether or not this Filter should be included.
3. Add a Filter condition by following these steps. You can add multiple groups and multiple conditions under each group.

- a. Click **Add Condition** .
- b. Specify rules for the condition. You can define the conditions using one of the following:
 - **Expression Builder:** You can form conditions using all the operators given in the Expression Builder. The Expression Builder is used to define free flow text filter conditions. To define a filter condition using the Expression Builder, follow these steps:
 - i. Click **Exp**. The Expression Builder dialog box is displayed.
 - ii. Select the required Dataset, Attribute and Runtime Parameters and operators. The resulting condition is displayed in the Condition field.
 - iii. Click **Save**  to save the changes.
 - **Tables:** You can define conditions using the various columns of tables. The columns of the two tables are compared with each other using the required operators. To define conditions using tables, follow these steps:
 - i. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - ii. Select a Column for the left-hand side from the drop-down list.
 - iii. Select an Operator from the drop-down list.

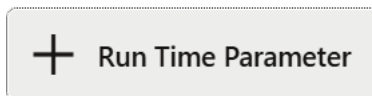
- iv. Select a Table or Data Forge for the right-hand side from the drop-down list.
 - v. Select a Column for the right-hand side from the drop-down list.
- **Text:** You can define filter conditions using text. A particular column in a table is compared with the input text using the required operators. To define filter conditions using text, follow these steps:
 - i. Click **Text**.
 - ii. Select a Table or Data Forge for the left-hand side from the drop-down list.
 - iii. Select a Column for the left-hand side from the drop-down list.
 - iv. Select an Operator from the drop-down list.
 - v. Enter the text in the field on the right-hand side.

Creating Runtime Parameters

A Runtime parameter is a variable whose value can be defined and then called from within that same pipeline.

When you define a runtime parameter, you enter the default value to use. When you create or edit a job that includes a pipeline with runtime parameters, you can specify another value to override the default.

To create a runtime parameter, follow these steps:



1. Click **Add Runtime Parameter** . The New Runtime Parameter dialog box displays.
2. Provide the details as described in the following table:

Table 14-19 Fields in New Runtime Parameter and their Descriptions

Field	Description
Name	Enter the name for the runtime parameter. Any spaces () given in the Name field will be replaced with underscore (_) while saving the runtime parameter.
Description	Enter the description for the runtime parameter.
Datatype	Select the datatype for the runtime parameter from the drop-down list.
Default Value	Provide the default values for the runtime parameter.

3. Click **OK**. The runtime parameter is created.

Using the Data Transformation Widget

The Data Transformation widget allows you to write and import complex expression and transformation in Python and Java.

To use this widget effectively, you must have a working knowledge of scripting and Java or Python.

1. Create a pipeline, as shown in [Adding DTP Pipelines](#).

2. Add a Data Transformation widget to the pipeline by right-clicking the Canvas to display the list of widgets and selecting **Data Transformation** .

The Data Transformation widget can be stand-alone or associated with another widget, such as Business Filter, Evaluation, or Aggregator.

3. Click **Options**  then click **Edit** . The Data Transformation pane displays.
4. In Basic Configuration, provide the following details:
 - a. Enter a **Name** for this Data Transformation.
 - b. Select a **Runtime Parameter** from the drop-down list. To add a new Runtime Parameter, see [Creating Runtime Parameters](#).
 - c. For Data Transformations associated with another widget, map the columns in **Column Definition** using **Add Columns**. The **View Input Table Columns** shows which columns were mapped in the connected Data Forge.
5. The **Define Transformation** tab displays the script options.
 - **Standard**
 - a. Select the **Type** from the drop-down list.
 - b. Select the **Transform** from the drop-down list.
 - c. Select **Interpreters** from the drop-down list. Python allows you call a custom python script.
 - i. In the **Script** section, provide a description.
 - ii. Enter or paste the python script you want to execute.

Note

If this is a script you intend to reuse, you can create a template by clicking **Templates**, then **Add**. Enter the pertinent details about the script, enter or paste the script into the box, and click **Save**. Existing templates can be modified by clicking **Edit**. Editing a template updates the script in every pipeline which uses this template.

To use an existing template, select **Use Template** and select the desired template from the **Script** drop-down list.

- iii. Click **Save**.
- Selecting **Custom** allows you to create a notebook. Notebooks allow you to add and run paragraphs, invalidate session, edit, add, export the notebook, and so on. To understand the icons which display in the Notebook, see [Common Screen Elements](#).
 - a. Click **Add Python Paragraph** to add a paragraph in the notebook and fetch the runtime parameters. This is available for all the interpreters.
 - b. Add paragraphs to the Notebook as desired.
 - c. Click **Play** to run the script and display the label.

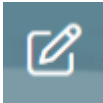
 6. You can import an existing notebook by selecting **Import Notebook**.

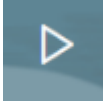
- Click **Import Archive File** to select the .dsnb file from the file selector dialog and select a file. Click **Open**.
- OR
- Drag and drop the file into the **Import Archive File** field.

Common Screen Elements in the Notebook

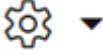
This topic describes common screen elements in the notebook that can be used to perform quick actions when preparing and executing the notebooks.

Elements in the Notebook

Icon Name	Action/Description
Modify Notebook 	Modifies the details of a notebook, such as a name, description, and (or) tags.
Hide Code 	Hides or shows the Code Section in all the paragraphs in a notebook.
Hide Result 	Hides or shows the Results Section in all the paragraphs in a notebook.
Re-initialize session 	Re-initialize the paragraph in a notebook. Note: By default, the Start Widget gets executed automatically on clicking the Re-initialize icon. All other paragraphs need to be executed manually.
Read-Only 	Sets the notebook to read-only mode. Note: The notebook is protected from edit, clear result, delete, share, reset session, and run paragraphs in Read-only mode.
Write 	Set the notebook to write mode.

Icon Name	Action/Description
Run Paragraphs 	Executes all the paragraphs in a notebook in sequential order.
Invalidate Session 	Resets any connection or code executed in a notebook.
Delete Notebook 	To delete the selected notebook.
Clear Result 	Clears results for all the paragraphs in a notebook. Note: This action clears all the results. You must rerun the paragraphs to view the results.
Clear Paragraph Dependencies 	Remove all defined paragraph dependencies.
Open as an iframe 	Opens a notebook in an Iframe. This allows a notebook to be embedded inside another webpage.
Share Notebook 	Share the notebook with another user, user group, or role.
Clone Notebook 	Creates a copy of a notebook. All paragraphs in the current notebook are replicated in the new notebook. The cloned notebook is created with the default name, Copy of <Current Notebook Name>.

Icon Name	Action/Description
Export Notebook 	Export the notebook to your computer as a DNSB file.
Print Notebook 	To print a notebook in the PDF format and save it in your local machine.
Default Template 	To apply the overall look and feel of the notebook using the default template.
Layout 	Sets the preferred layout. The available layouts are Zeppelin and Jupyter.
Default Template 	Applies the overall look and feel of the notebook using the default template.
Show Panel 	Shows or hides the Paragraph Settings Bar Commands, Results Toolbar, and Settings Dialog for a selected paragraph in a panel to the notebook's right.
Attach Credentials 	You can use this option to attach credentials (wallet and password) to the notebook to enable secure data access.
Versioning 	You can use this option to create versions for your notebook, which helps you analyze the changes based on the version control.

Icon Name	Action/Description
Run Paragraph	To execute the code or query in a paragraph.
	
Enter Dependency Mode	To add or remove dependent paragraphs. Paragraphs with dependent paragraphs are executed in the dependency order.
	
Comments	To add comments to a paragraph.
	
Expand	To expand a paragraph and view the paragraph in full-screen mode.
	
Show/Hide Line Numbers	To show or hide line numbers in the code in a paragraph. Note: This button is applicable only to the code section.
	
Visibility	To manage the visibility settings in a paragraph. It controls how a paragraph may be viewed by the author and other users who have access to the notebook.
	
Settings	Users can perform the following actions: • Resize the width of a paragraph • Change the order of placement of the paragraphs by moving them up or down • Run the all above or all below paragraph from the selected paragraph. • Clear the paragraph result • Open the notebook in Embedded window • Disable the run action • Clone the paragraph • Delete a paragraph
	

About Scenario Pipelines

Scenario Pipelines enable you to create scenarios by defining behavior that consists of events in a predetermined order.

You can use these events to thread multiple data streams together. Scenarios are used to identify behaviors of interest, potentially problematic behaviors with respect to securities, regulations, and possible money-laundering activities.

These scenarios consider whether the geographical location or entities involved warrant enhanced scrutiny; monitor activity between accounts, customers, correspondents, and other entities to reveal relationships that could indicate efforts to launder funds; address sudden, significant changes in transaction activity that could indicate money laundering or fraud; and detect other types of activities that are considered potentially suspicious or indicative of money laundering.

As part of configuring a scenario pipeline, parameters are defined, which are then tuned in the Threshold Manager.

Pre-Configured DTP Scenario Pipelines

The application comes with the following ready-to-use, pre-configured scenario pipeline:

- Networks of Accounts and Entities and Customers- Account Focus
- CIB High Risk Geography Activity - Account Focus DTP
- Focal High Risk Entity - Customer Focus DTP
- Large Reportable Transaction - Customer Focus DTP
- Deviation from Peer Group - Total Activity - Account Focus

These templates are copies of existing Transaction Monitoring scenarios, included here as implementation samples demonstrating the technical capabilities of DTP pipeline widgets.

Note

Do not execute the same pipelines from **Data Transformation Pipeline** and **Scenario Pipeline** for the same MIS Date.

Managing Threshold Sets (MMG)

FCCM Cloud Service uses tunable Thresholds to change variable values for scenarios.

When scenarios are created, thresholds are established. You can use the Threshold Editor to modify threshold values of scenarios, and create and edit threshold sets to fine-tune how that scenario finds matches, without changing the values defined at the data set or pattern level. These thresholds are applied to scenarios to find matches. Using this tool, you can enter a new value for a threshold (within a defined range) or reset the thresholds to their sample values.

Threshold sets allow you to run the same scenario multiple times against a variety of sources (for example, currencies, or jurisdictions) with separate threshold values for each source.

Note

Changing threshold values can generate significantly more or fewer alerts, depending upon the modifications made.

Accessing Threshold Sets

You can access the threshold sets through the Threshold Manager.

To access the All Threshold Sets page and view threshold sets, follow these steps:

1. In the Navigation List menu, click **Business Processes**, then click **Data Transformation Pipeline**.
2. Select **DTP Threshold Manager**. The All Threshold Sets page displays the complete list of threshold sets available in your implementation.
3. Select the check box for the threshold set you want to view, or click the threshold set name to view the threshold set details.

Creating a Threshold Set

You can create new threshold sets in the Threshold Editor.

To create a new threshold set, follow these steps:

1. Navigate to the **Threshold Editor** page.
2. Click **Add**  in the top right corner. The New Threshold Set page displays.
3. Enter the following values:
 - a. Enter a **Name** for this threshold set.
 - b. The scenario associated with this threshold set displays additional configurable parameters. These parameters are specific to the selected scenario.
 - c. Select one or more **Jurisdictions** for this threshold set using the drop-down list. Jurisdiction refers to the division of data in the database based on criteria such as geographical boundaries, legal entities, and so on.
 - d. Select the **Period Type** used in this threshold set using the drop-down list. Options include Days or Months. By default, the period type is **Days**.
 - e. Enter the **Lookback Period** for this threshold set. Lookback period refers to the number of days or months to lookback from the current date or time to create a time window which is used to consider cases for correlation.
 - f. Enter the **Frequency Period** for this threshold set. Frequency period refers to how frequently the scenario should be run, by number of days or months.
 - g. Enter any **Comments** you have for this threshold set.
 - h. Enter the threshold values you want this threshold set to be updated with in the **New Value** field.
4.
 - Click **Save**  to save the values. A new threshold set is created and a message displays: *New Threshold Set created successfully*.

- Click **Save & Simulate**. A new threshold set is created, and the Create Simulator Conditions page displays. For more information about the Threshold Simulator, see [Threshold Simulator](#).

Copying a Threshold Set

Copy an existing threshold set before modifying the values.

To copy and modify an existing threshold set, follow these steps:

1. Navigate to the Threshold Manager page.
2. Select the check box for the threshold set you want to copy.
3. Click **Copy** . The Copy Threshold Set window displays.
4. Enter the following values:
 - a. Enter a **Name** for this threshold set.
 - b. The scenario associated with this threshold set displays additional configurable parameters. These parameters are specific to the selected scenario.
 - c. Select one or more **Jurisdictions** for this threshold set using the drop-down list. Jurisdiction refers to the division of data in the database based on criteria such as geographical boundaries, legal entities, and so on.
 - d. Select the **Period Type** used in this threshold set using the drop-down list. Options include Days or Months. By default, the period type is **Days**.
 - e. Enter the **Lookback Period** for this threshold set. Lookback period refers to the number of days or months to lookback from the current date or time to create a time window which is used to consider cases for correlation.
 - f. Enter the **Frequency Period** for this threshold set. Frequency period refers to how frequently the scenario should be run, by number of days or months.
 - g. Enter any **Comments** you have for this threshold set.
 - h. Enter the threshold values you want this threshold set to be updated with in the **New Value** field.
5. Click **Save**  to save the values.

Editing Threshold Sets

You can modify user-defined threshold sets in the Threshold Editor.

To edit a threshold set, follow these steps:

1. Navigate to the Threshold Editor page. The existing threshold sets are displayed.
2. Select the check box corresponding to the threshold set you want to edit.

3. Click **Edit** . The Edit Threshold page is displayed.

Alternatively, you can click **Edit** when viewing a threshold set in the View Threshold Set window.

4. Modify the required details.

- a. Select one or more **Jurisdictions** for this threshold set using the drop-down list. Jurisdiction refers to the division of data in the database based on criteria such as geographical boundaries, legal entities, and so on.
- b. Select the **Period Type** used in this threshold set using the drop-down list. Options include Days or Months. By default, the period type is **Days**.
- c. Enter the **Lookback Period** for this threshold set. Lookback period refers to the number of days or months to lookback from the current date or time to create a time window which is used to consider cases for correlation.
- d. Enter the **Frequency Period** for this threshold set. Frequency period refers to how frequently the scenario should be run, by number of days or months.
- e. Enter any **Comments** you have for this threshold set.
- f. Enter the threshold values you want this threshold set to be updated with in the **New Value** field.

Note

You can modify the values of the existing parameters only, you cannot add new parameters.

5.
 - Click **Save**  to save the values. A new threshold set is created and a message displays: *New Threshold Set created successfully*.
 - Click **Save as New** to save the edited threshold set as a new threshold set. The New Threshold window opens. Enter a name for the threshold set and click **Save**. A new threshold set is created and a message displays: *New Threshold Set created successfully*.
 - Click **Save & Simulate**. A new threshold set is created, and the Create Simulator Conditions page displays. For more information about the Threshold Simulator, see [Threshold Simulator](#).

Deleting Threshold Sets

You can delete user-defined threshold sets in the Threshold Editor.

To delete a threshold set, follow these steps:

1. Navigate to the Threshold Editor page. The existing threshold sets are displayed.
2. Select the check box corresponding to the threshold set you want to delete.
3. Click **Delete** . A message displays: *Are you sure you want to delete <threshold set>?*
4. Enter the reason for deletion in the **Your Comment** box. This is mandatory.
5. Click **Delete**. The threshold set is deleted.

About Threshold Simulator

The Threshold Simulator is used to run selected scenario pipelines against the selected threshold sets to find the matches obtained from these combinations.

These matches enable you to identify the events generated for the combination on a specified date. This can be helpful when you want to see which events would be generated with different

threshold settings and be able to explain why your scenario is configured as it is, such as during audits.

View Simulator Conditions

You can view Simulator Conditions using the Threshold Manager.

To view simulator conditions for a threshold set, follow these steps:

1. In the Navigation List menu  click **Business Processes**, then **Data Transformation Pipeline**.
2. Select **DTP Threshold Manager**. The All Threshold Sets page displays.
3. In the All Thresholds page, select the check box corresponding to the threshold set you want to view simulation details for.
4. Click **Simulation History** . The Simulation Details for this threshold set opens as a new tab.
5. Click the **Simulation Run ID** for the simulation you want to view conditions for. The View Threshold Set pop-up window displays the details of this threshold set.

If you want to make modifications to these conditions, click **Edit & Simulate**. The Edit Threshold Set pop-up window displays. Follow the steps in [Edit Simulator Conditions](#) to make these changes.

Edit Simulator Conditions

You can edit Simulator Conditions using the Threshold Manager.

To edit existing simulator conditions for a threshold set, follow these steps:

1. In the All Thresholds page, select the check box corresponding to the threshold set you want to edit.
2. Click **Simulation History** . The Simulation Details for this threshold set opens as a new tab.
3. Select the check box corresponding to the **Simulation Run ID** for the simulation you want to edit.

4. Click **Edit** . The Edit Simulator Conditions pop-up window displays.

Alternatively, you can access the Edit Simulator Conditions window by clicking **Edit & Simulate** in the View Simulator Conditions window.

5. To make modifications to these conditions, update the following values:
 - a. Select one or more **Jurisdictions** for this threshold set using the drop-down list. Jurisdiction refers to the division of data in the database based on criteria such as geographical boundaries, legal entities, and so on.
 - b. Enter the **Lookback Period** for this threshold set. Lookback period refers to the number of seconds, minutes, hours, or days to lookback from the current date or time to create a time window which is used to consider cases for correlation.
 - c. Enter the **Frequency Period** for this threshold set. Frequency period refers to how frequently the scenario should be run.

- d. Enter the threshold values you want this threshold set to be updated with in the **New Value** field. Select the Batch Date as the date for the data you want to test the thresholds against. This can be the current date or a past date.
 - e. Enter the Data Origin.
 6. Click **Save & Simulate** to run the modified simulation. After running a simulation, the Simulator History window displays the result with the number of event matches and details of each event. You can view the following details:
 - Simulation ID: ID for this simulation run.
 - Data Origin: Data Origin of the data set the simulated scenario conditions are run against.
 - Jurisdiction: Jurisdiction the simulated scenario conditions are run for.
 - Batch Date: Date the simulated scenario conditions are run for.
 - Results: Number of event matches generated by the simulated scenario conditions during the run.
 - Event Details: Details for each event generated by the simulated scenario conditions during the run.
 - Event ID
 - Focus Name
 - Focus Type
 - Highlights

Creating New Simulator Conditions

You can create new Simulator Conditions using the Threshold Manager.

To create new simulator conditions for a threshold set, follow these steps:

1. In the **All Thresholds** page, select the check box corresponding to the threshold set you want to create new simulator conditions for.
2. Click **Simulation History** . The Simulation Details for this threshold set opens as a new tab.
3. Click **Open Simulator**. The Create Simulator Condition pop-up window displays.

Alternatively, you can access the Create Simulator window by clicking **Open Simulator** in the All Thresholds page. The Create Simulator Condition pop-up window displays.
4. Create the simulator conditions by providing the following details:
 - a. Select the required scenario from the **Select Scenario** drop-down list.
 - b. Select the required threshold set from the **Select Threshold** drop-down list.
 - c. Select the **Batch Date** as the date for the data you want to test the thresholds against. This can be the current date or a past date.
 - d. Enter the **Data Origin**.
5. Click **Save**.

Running Existing Threshold Sets

You can run a simulation for existing scenario threshold sets.

To run the threshold simulator for an existing scenario threshold set, follow these steps:

1. In the All Thresholds page, select the check box corresponding to the threshold set you want to run the simulator for.
2. Click **Open Simulator**. The Simulator Conditions pop-up window displays.
Alternatively, you can also run existing threshold set from the Simulation Details tab by clicking Open Simulator. The Simulator Conditions pop-up window displays.
3. Provide the following details:
 - a. Select the **Batch Date** as the date for the data you want to test the thresholds against. This can be the current date or a past date.
 - b. Enter the **Data Origin**.
 - c.
 - d. Select one or more **Jurisdictions** for this threshold set using the drop-down list. Jurisdiction refers to the division of data in the database based on criteria such as geographical boundaries, legal entities, and so on.
 - e. Enter the **Lookback Period** for this threshold set. Lookback period refers to the number of seconds, minutes, hours, or days to lookback from the current date or time to create a time window which is used to consider cases for correlation.
 - f. Enter the **Frequency Period** for this threshold set. Frequency period refers to how frequently the scenario should be run.
 - g. Enter any comments you have for this threshold set.
 - h. Enter the threshold values you want for this threshold set in the **New Value** field.
4. Click **Save & Simulate** to run the modified simulation. After running a simulation, the Simulator History window displays the result with the number of event matches and details of each event. You can view the following details:
 - Simulation ID: ID for this simulation run.
 - Data Origin: Data Origin of the data set the simulated scenario conditions are run against.
 - Jurisdiction: Jurisdiction the simulated scenario conditions are run for.
 - Batch Date: Date the simulated scenario conditions are run for.
 - Results: Number of event matches generated by the simulated scenario conditions during the run.
 - Event Details: Details for each event generated by the simulated scenario conditions during the run.
 - Event ID
 - Focus Name
 - Focus Type
 - Highlights

Viewing Simulation Details

When a threshold set simulation has completed, you can view the details of all thresholds in this simulation and their results in the Simulation Details.

This allows you to determine whether your threshold set is generating the expected detection results or if further modification is required. You can also view the Simulation Details for all runs of a threshold set which has been previously run. To open the Simulation Details, follow these steps:

1. In the All Thresholds page, select the check box corresponding to the threshold set you want to view simulation details for.
2. Click **Simulation History** . The Simulation Details for this threshold set opens as a new tab. You can view the following details:
 - Simulation Run: ID for this simulation run with the scenario name.
 - Jurisdiction: Jurisdiction the simulated scenario conditions are run for.
 - Run Date and Time: Date and time the simulation was run.
 - User ID: User who ran the simulation.

You can export the results of the threshold set simulation runs in .xlsx format by selecting the check boxes for one or more simulation runs and clicking **Export Results**.

You can delete a simulation run by selecting the check boxes for one or more simulation runs and clicking **Delete**. The Delete pop-up window displays: *Are you sure you want to delete <Simulation ID>?* Enter a comment and click **Delete**. A confirmation message displays.

To return to the All Threshold Sets tab, click the **All Thresholds List** tab.

Using Jobs (MMG)

The application uses jobs to define the instructions for executing the data pipelines or scenario pipelines against threshold sets, for example, running a scenario or loading data.

These jobs can be included in batches (groups of jobs) which run at configured intervals against the selected threshold to detect and generate events. This allows the jobs to run automatically, without requiring your involvement. Jobs can also be used to monitor the execution of jobs.

To access the DTP Jobs page, follow these steps:

1. In the Applications landing page, click the **Navigation Menu**  to access the Navigation List.
2. Click **Business Processes**, then click **Data Transformation Pipeline** in the Navigation List to view all DTP menu options.
3. Select **DTP Jobs**. The DTP Jobs page opens.

Creating Jobs

You can create new jobs to run in batches.

To create a new job, follow these steps:

1. Navigate to the **DTP Jobs** page.
2. Click **Create Job** . The Create Job pane is displayed.
3. Provide the details as described in the following table:

Table 14-20 Fields to Create Jobs

Field	Description
Job Name	Enter the name for the job.
Pipeline Type	Select the pipeline type for which you want to create the job. The available options are Data and Scenario.
Pipeline	Select the pipeline from the drop-down list for which you want to create a job.
Threshold	Select the threshold set from the drop-down list. The drop-down list displays the list of thresholds that are created for the selected scenario pipeline. The job is run against the selected threshold to detect and generate events.

4. Click **Save**  to save the changes. A new job is created and displayed in the Jobs page.

Editing Jobs

You can edit or delete user-defined jobs

To edit a job, follow these steps:

Note

Pre-configured jobs cannot be edited or deleted. You can only edit or delete user-defined jobs.

1. Navigate to the **DTP Jobs** page.
2. Click on the Job that you want to modify and click **Options**  then click **Edit**  . The Edit Job pane displays.
3. Modify the required details in the Edit Job pane on the right-hand side.
4. Click **Save** to save the changes. The job is modified.

Deleting Jobs

You can edit or delete user-defined jobs.

To delete a job, follow these steps:

Note

Pre-configured jobs cannot be edited or deleted. You can only edit or delete user-defined jobs.

1. Navigate to the **DTP Jobs** page.
2. Click on the Job that you want to modify and click **Options** *** then click **Delete**. The job is deleted.

Viewing Execution Monitor

The Execution Monitor enables you to view the complete history of job execution, including the start and end time, status, log messages generated during job execution, and so on. This will help you see how your jobs are progressing and detect any recurring issues.

To view the execution monitor, follow these steps:

1. Navigate to the Applications landing page.
2. Click the **Navigation Menu** to access the **Navigation List**. The Navigation List displays the list of modules.
3. Expand **Data Transformation Pipeline** in the Navigation List to view all DTP menu options.
4. Click **DTP Jobs** in the Navigation List. The Jobs page opens in a new window.
5. On the Jobs page, click the job for which you want to view the execution monitor.
6. The Execution History pane displays the historical information of the selected job. The details include the Batch Run ID, the date and time during which a job is executed, status of the job execution, and so on.
Use the **Task** and **Sort By** drop-down lists to manage how these details display.
7. Click **Monitor Execution** corresponding to the Batch Run ID for which you want to view more information.
8. The Execution Monitor page is displayed in a new window. The page contains details of the pipeline for which the job is created. The widgets in the pipeline are represented in different colors. Widgets are color coded to indicate the widget type and status of the job:
 - Completed
 - Failed
 - Pending
 - Not Started
9. Clicking **Log Messages**



displays the Log Messages pane which lists a complete log of messages generated during the execution of the pipeline.

10. Click each widget to view the Log Messages dialog box with detailed information about the widget.

The following information is included:

- **Log Messages:** Start and End Date
- **Query/REST Parameters:** The query that is being executed by this widget.
- **Parameters:** The relevant parameters being used in this widget.
- **Errors:** Details for any errors that occurred in this widget during Job execution.

User Group and Roles Mapping

This section provides the User Group, User Role mapping, and activities for Oracle FCCM Data Transformation Pipeline Cloud Service.

This table shows the User Groups and Roles required for activation of Oracle FCCM Data Transformation Pipeline Cloud Service.

Table 14-21 User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPCONFIGGRP	DTPCONFIGROLE	DTPPIPELINEACC

Table 14-21 (Cont.) User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPCONFIGGRP	DTPCONFIGROLE	DTPTHRESHOLDACC

Table 14-21 (Cont.) User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPCONFIGGRP	DTPCONFIGROLE	DTPJOBSACC

Table 14-21 (Cont.) User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPCONFIGGRP	DTPCONFIGROLE	DTPAUDITACC

Table 14-21 (Cont.) User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPADMNGRP	DTPADMNROLE	DTPFOCUSTYPEACC

Table 14-21 (Cont.) User Group and Roles Mapping in Oracle FCCM Data Transformation Pipeline Cloud Service

Group	User Role	Functions
DTPADMNGRP	DTPADMNROLE	DTPEVENTTYPEACC

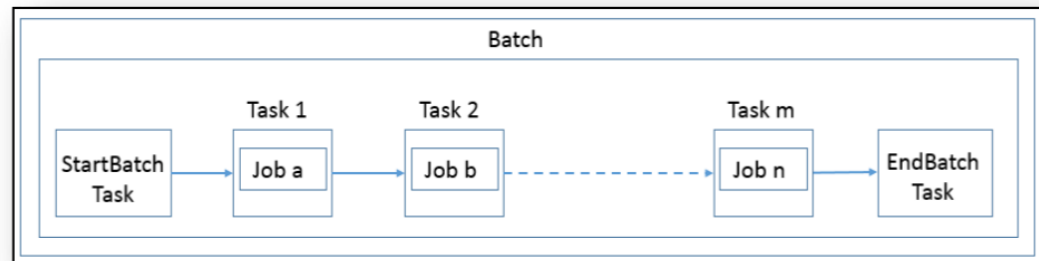
Managing Batches

A batch is a group of jobs that are scheduled to run at a defined interval of time, in sequence, automatically, without user involvement.

Oracle FCCM Cloud Service uses the Scheduler Service to create, schedule, execute and manage batches. A batch is a group of jobs that are scheduled to run at a defined interval of time, in sequence, automatically, without user involvement. Each batch begins with a StartBatch, includes any additional jobs that should be run in this batch, and then completes with the Endbatch.

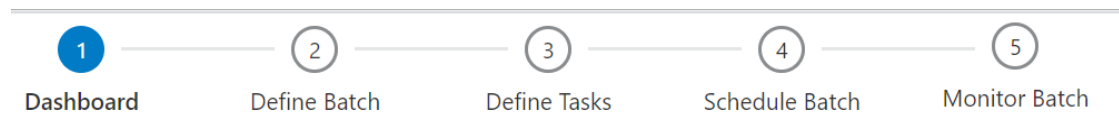
Note

Oracle recommends that you first copy, customize, and then use the pre-configured batches as required.

Figure 14-14 Flow of Batch

To execute the batches, use the Schedule Batch feature in the Scheduler Service. For more information, see [Scheduler Service](#). You can use the Scheduler Service to first define the batch, then define which tasks should be included in this batch.

Next, you must schedule the batch. When the batch runs, you can monitor the batch to verify it is executing as intended. Click the Batch Scheduling Flow to navigate through the Scheduler Service.

Figure 14-15 Scheduler Service -- Batch Scheduling Flow

Pre-Configured Batches

The application contains certain pre-configured batches that can be used to run the default data. You must create new batches to run customer-specific data.

DTPTMScenario Batch Details

The following table provides the tasks that are configured for the DTPTMScenario batch. These tasks must be executed in the following order:

Table 14-22 DTPTMScenario Batch Details

Sequence	Tasks for DTPTMScenario Batch	Jobs for DTPTMScenario Batch	Pipelines for DTPTMScenario Batch
1	StartBatch	Not Applicable	Not Applicable

Table 14-22 (Cont.) DTPTMScenario Batch Details

Sequence	Tasks for DTPTMScenario Batch	Jobs for DTPTMScenario Batch	Pipelines for DTPTMScenario Batch
2	CIBHRGAC	CIB High Risk Geography Activity - Account Focus - DTP	CIB High Risk Geography Activity - Account Focus - DTP
3	FHRECU	Focal High Risk Entity - Customer Focus DTP	Focal High Risk Entity - Customer Focus DTP
4	LRTCUC	Large Reportable Transaction - Customer Focus DTP	Large Reportable Transaction - Customer Focus DTP
5	NOA	Networks of Accounts and Entities and Customers - Account Focus	Networks of Accounts and Entities and Customers - Account Focus
6	DPGTAAC	Deviation from Peer Group - Total Activity - Account Focus	Deviation from Peer Group - Total Activity - Account Focus
7	EndBatch	Not Applicable	Not Applicable

Purge Batch Details

Purge batches are used when your batch has not executed successfully to purge the data and execute the batch again.

If your batch has not executed successfully, has been explicitly interrupted or canceled, or was put on hold during the execution process, then you can purge the data and execute the batch again.

Note

Purge batches are not a regular required activity. They should only be used when other methods to re-run the batches are not successful.

To purge the data which was partially processed during the interrupted or canceled batch execution, follow the steps described in the following table.

Note

Purge Batches should be run in the reverse order of batch execution.

1. Ingestion
2. DTPTMScenario
3. CMIngestion
4. AMLtoCaseManagement

Then the purge batch order should be:

1. AMLtoCaseManagement
2. CMIngestion
3. DTPTMScenario
4. Ingestion

Table 14-23 Executing Purge Batches

Failed Batch Name	Purge Batch	Steps to Purge	Additional Considerations
DTPTMScenario	PurgeDTPTMTTables	1. Run the PurgeDTPTMTTables batches by providing the failed batchrunID as input parameter in the field \$FCCBATCHRUNID\$. 2. Execute the Ingestion batch for the batch date.	—

PurgeDTPTMTTables Batch Details

The following table provides the tasks that are configured for the PurgeDTPTMTTables batch. These tasks must be executed in the following order:

Table 14-24 PurgeDTPTMTTables Batch Details

Sequence	Tasks for PurgeDTPTMTTables Batch	Jobs for PurgeDTPTMTTables Batch	Pipelines for PurgeDTPTMTTables Batch
1	StartBatch	Not Applicable	Not Applicable
2	DeleteTMAAlertsTable	Purge TM Alerts Tables	Purge TM Alerts Tables
3	EndBatch	Not Applicable	Not Applicable

PurgeAMIngestionTables Batch Details

The following table provides the tasks that are configured for the PurgeAMIngestionTables batch. These tasks must be executed in the following order:

Table 14-25 PurgeAMIngestionTables Batch Details

Sequence	Tasks for PurgeAMIngestionTables Batch	Jobs for PurgeAMIngestionTables Batch	Pipelines for PurgeAMIngestionTables Batch
1	StartBatch	Not Applicable	Not Applicable
2	DeleteAMTable	Delete Business Data	Delete Business Data
3	EndBatch	Not Applicable	Not Applicable

Note

There are no Out of the Box data pipelines associated with DTP.

Administrative Tasks

Certain tasks are frequently performed by Supervisor or Administrator users.

This section tells how to perform the following administrative tasks:

- [Viewing Audit History](#)
- [Creating Focus Types](#)
- [Creating Event Types](#)

Viewing Audit History

The Audit History provides a record of changes made to pipelines or threshold configuration.

For example, changing a pipeline name, adding or modifying widgets in the pipeline, creating or deleting a threshold set, changing threshold set configurations, and so on.

This allows you to view the changes made to a scenario before executing the scenario, and to detect and mitigate the risk of internal employee manipulation, as required by auditors and regulators.

You can also filter the results to show updates made to a specific pipeline since the last tuning cycle or last regulatory exam. If needed, you can export the Audit History data in .csv or .xls format to analyze further.

Accessing Audit History

You can access the Audit History from the Navigation List.

1. Navigate to the Applications landing page.
2. Click the **Navigation Menu**  to access the **Navigation List**. The Navigation List displays the list of modules.
3. Click **Business Processes** and then click **Data Transformation Pipeline** in the Navigation List to view all DTP menu options.
4. Select **DTP Audit Screen**. The Audit History page displays.

The following table describes the columns which display in the Audit History.

Table 14-26 Columns in the Audit History

Column	Description
Component	Type of component the action was taken on. For example, Data Transformation Pipeline or Threshold.
Component Name	Name of the threshold or pipeline the action was taken on. For example, if you are viewing the Audit History for a scenario pipeline, the scenario name will display.
Sub Component	Type of sub component the action was taken on. For example, Business Filter, Aggregation, Evaluation, and so on. If there is no sub component, this column will appear blank.
Sub Component Name	Name of the sub component the action was taken on. For example, if a scenario Risk Indicator was updated, the name of the Risk Indicator will display, such as <i>Total of Very High Risk Amount Percentage</i> . If there is no sub component, this column will appear blank.
Action	Type of action that was taken. The DTP Audit History tracks the following types of actions: • Copy: An existing component has been copied with a new name. For example, an existing scenario pipeline is copied and then configured to meet your specific business needs. • Bulk Delete: A widget or pipeline is deleted at the parent level without deleting its underlying parameters. • Delete: A parameter or threshold is deleted. • Edit: A change has been made to an existing component or subcomponent. • Export: A pipeline or threshold set has been downloaded. • Import: A pipeline or threshold set has been imported. • Insert: A new threshold has been added to an existing pipeline.
Current State	The current state of the component which was acted upon. Compare the Current State with the Previous State to see the change which was made.
Previous State	The previous state of the component which was acted upon. Compare the Current State with the Previous State to see the change which was made.
Updated By	User who took the action.
Date & Time	Date and Time the action was taken.

Filtering Audit History

The Filter option allows you to search and narrow down the results of the Audit History.

You can use a combination of these search criteria to quickly find the components you are interested in. If you don't enter any value in any search field, it is equivalent to selecting all the criteria.

To filter the Audit History, follow these steps:

1. In the Audit History page, click **Filter** . The Filter criteria appear on the left-hand-side pane.
2. Select one or more criteria using the drop-down list selectors. You can filter by the following criteria:
 - Updated by
 - Action
 - Component
 - Component Name
 - From Date
 - To Date
3. Click **Apply** to filter the Audit History by the selected criteria.
 - Click **Reset** to clear the filter criteria.
 - Click **Close** to remove all filters.

Exporting Audit History

If needed, you can export the Audit History data in .csv or .xls format to analyze further.

Filter the Audit History data as desired, using the steps in [Filtering Audit History](#).

To export the Audit History, follow these steps:

1. In the Audit History page, click **Export** to expand the output options.
2. Select from the following output options:
 - CSV
 - XLS
3. Select a folder to save the exported data.
4. Enter a file name, if desired.
5. Click **Save**. The exported Audit History is saved.

Creating Focus Types (MMG)

Organizations can create and configure Focus Types according to their business needs.

A Focus Type describes the main entity on which the analysis, report, or calculation is centered. For example, a focal entity might be a customer whose transactions are being analyzed for suspicious activity. Focusing on the entity of interest ensures the granularity required for regulatory, risk, and business reporting or monitoring is appropriate. Relevant data is aggregated, filtered, or analyzed specifically for that entity.

The Account and Customer focus types are provided out-of-the-box, but you can create additional focus types to meet your organization's business needs.

To create a focus type, follow these steps:

1. Navigate to the Applications landing page.
2. Click the **Navigation Menu** to access the **Navigation List**. The Navigation List displays the list of modules.

3. Click **Application Administration**, then **Data Transformation Pipeline Admin**.
4. Select **Focus Type**. The Focus Type admin page displays with a sortable list of current focus types.



5. Click **Add**. The Add Focus Type pane opens.
6. Enter a **Name** for the Focus Type, such as External Entity. Valid characters include: A-Z a-z '-', with a maximum length of 100 Characters. For example, ABCDef or Abc-Def.
7. Click **Save**. A Focus Type ID is generated for this type and the new focus type is added to the list.

Creating Event Types (MMG)

Organizations can create and configure Event Types according to their business needs.

Event Types, such as AML, show the designated classification for the behavior being detected when an event is generated.

To create an event type, follow these steps:

1. Navigate to the Applications landing page.
2. Click the **Navigation Menu** to access the **Navigation List**. The Navigation List displays the list of modules.
3. Click **Application Administration**, then **Data Transformation Pipeline Admin**.
4. Select **Event Type**. The Event Type admin page displays with a sortable list of current event types.



5. Click **Add**. The Add Event Type pane opens.
6. Enter a **Name** for the event type, such as Anti Money Laundering. Valid characters include: A-Z a-z '-', with a maximum length of 100 Characters. For example, ABCDef or Abc Def.
7. Enter a **Code** for the event type, such as AML.
8. Click **Save**. An Event Type ID is generated for this type and the new event type is added to the list.

If you want to delete an Event Type, click **Delete** . A verification message displays. Click **Delete** to remove the event type from the list.

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	-	%python from mmg.constants import * user = "SAUSER" set_user(user)	user - username to be set	8.1.1
Get User	Get the user that has been set	Set User	%python from mmg.constants import * get_user()	-	8.1.1
Attach Workspace	Attach the workspace for a session	-	%python from mmg.workspac e import attach_workspa ce workspace = "CS" attach_workspa ce(workspace)	workspace- workspace to be attached	8.1.1
Get Workspace	Get the workspace that has been attached	Attach Workspace	%python from mmg.workspac e import get_workspace get_workspace()		8.1.1
List Workspaces	Lists all workspaces available to a user	Set User	%python from mmg.workspac e import list_workspaces list_workspace s()		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	%python from mmg.workspac e import check_aif check_aif()		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	%python from mmg.workspac e import list_datasources list_datasource s(order=None)	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	%python from mmg.workspac e import list_datasources workspace="HE LLO NEW" list_datasource s(workspace,or der=None)	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	%python from mmg.workspac e import list_datasources list_datasource s()	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	%python from mmg.workspac e import list_datasources workspace="CS " list_datasource s(workspace)	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.workspac e 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.workspac e 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

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Get Connection Object	Returns an Oracle DB / SQLAlchemy connection object for the data source of order 1 in the attached workspace.	Set User, Attach Workspace	Python: from mmg.workspac e import get_conn conn = get_conn() # conn = get_conn(con n_type=None) import oracledb if not isinstance(c onn, oracledb.Con nection): print("Not Conn Object") print(conn)		8.1.2.0

Note
If conn_type is None or

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				a c l e d b (o r n o t s p e c i f i e d) , t h e A P I r e t u r n s a n o r a c l e d b . C o n n	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				ecti on b y de fau lt . if c onn _t ype ="s qla lch emy ", the API creat	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				<pre>test_connection() obj = connect_by_user_using_create_table_engine() from sqlalchemy</pre>	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				h e m y .	

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 an Oracle DB / SQLAlchemy connection object to the specified data source (by name).	Set User	<pre> Python: from mmg.workspace import get_conn datasource = "DS" conn = get_conn(datasource, conn_type=None) import oracledb if not isinstance(conn, oracledb.Connection): print("Not Conn Object") print(conn) datasource = Name of the datasource for which the connection object must be returned.</pre>	datasource - Name of datasource for which the connection object has to be returned	8.1.2.0

Note
If conn_type is not specified, the default connection type is 'OCI'.

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				S M O n e o r o r a c l e d b (o r n o t s p e c i f i e d) , t h e A P I r e t u r n s a n O r a c l e	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				l e d b . C o n n e c t i o n b y d e f a u l t . l f c o n n _ t y p e = " s q l a l c h e m y " ,	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				the API create a test scenario object by using create_test_engine()	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				from sqlalchemy import *	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
Get Connection Object To Data Source Using Order	Returns an Oracle DB / SQLAlchemy connection object to the data source of the specified order in the attached workspace.	Set User, Attach Workspace	<pre> Python: from mmg.workspace import get_conn datasource = 2 conn = get_conn(datasource, conn_type=None) import oracledb if not isinstance(conn, oracledb.Connection): print("Not Conn Object") print(conn) datasource = Order number of the datasource in the attached workspace for which the connection object must be returned.</pre>	datasource - Order number of datasource in attached workspace for which the connection object has to be returned	8.1.2.0

Note
If connection type

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				p e i s N O n e o r o r a c l e d b (o r n o t s p e c i f i e d) , t h e A P I r e t u r n s a n O	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				r a c l e d b . C o n n e c t i o n b y d e f a u l t . i f c o n n _ t y p e = " s q l a l c h e m	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				y " , t h e A P I c r e a t e s a c o n n e c t i o n o b j e c t b y u s i n g c r e a t e _ e n g i	

Table A-1 (Cont.) Python Paragraph Scripting

API Type	Functionality	Prerequisite	Notebook Execution Script	Notebook Script Input	Minimum Supported Version
				ne() f r o m S Q L A l c h e m y .	
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.
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.

Table A-2 (Cont.) PYTHON LINEPARSER APIs (for MMG version 8.1.0)

API Type	Functionality	Notebook Execution Script	Note
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		%python from mmg.constants import * user = "SAUSER" set_user(user)
2	Get User	Get the user that has been set	Set User	%python from mmg.constants import * get_user()
3	Attach Workspace	Attach the workspace for a session		%python from mmg.workspace import attach_workspace workspace = "CS" attach_workspace(workspace)
4	Get Workspace	Get the workspace that has been attached	Attach Workspace	%python from mmg.workspace import get_workspace get_workspace()
5	Get Base URL	Get MMG Studio Service Base URL		%python from mmg.constants import * get_mmg_studio_service_url()
6	List Workspaces	Lists all workspaces available to a user	Set User	%python from mmg.workspace import list_workspaces list_workspaces()

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
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_datasources list_datasources(order=None)</pre>
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="HELL O 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>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
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>
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_connection() if not isinstance(conn,cx_Oracle.connect): 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_connection(datasource, conn_type=None) if not isinstance(conn,cx_Oracle.connect): print("Not Conn Object") print(conn)</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
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>
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.Conne ction): 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.Conne ction): print("Not Conn Object") print(conn)</pre>

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
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	%python import oracledb from mmg.workspace import get_conn datasource=2 conn=get_conn(datasource, conn_type=None) if not isinstance(conn, oracledb.Connection): print("Not Conn Object") print(conn)
20	Get Objective Hierarchy	Returns the complete objective hierarchy including pipeline name		%python print(objective Id)
21	Fetch FTPSHARE Path	Get complete ftpshare path on the server		%python from mmg.datasets import fetch_ftpshare fetch_ftpshare()
22	Get Pipeline Name	Get the Pipeline Name to which the current notebook is attached	Attach Workspace	%python from mmg.constants import get_pipeline_name get_pipeline_name()
23	Get Pipeline ID	Get ID of Pipeline to which the current notebook is attached	Attach Workspace	%python from mmg.constants import get_pipeline_id get_pipeline_id()
24	Get Pipeline Version	Get version of Pipeline to which the current notebook is attached	Attach Workspace	%python from mmg.constants import get_pipeline_version get_pipeline_version()

Table A-3 (Cont.) MMG Library Python APIs

Index	API Type	Functionality	Prerequisite	Notebook Execution Script
25	Get MISDATE	Return MISDATE from pipeline execution if set otherwise takes default value provided	Attach Workspace	%python misdate=ds.date_picker(name='\$FICMISDATES', format='yyyy-MM-dd', default_value='2000-01-01', label='FICMISDATE') print(misdate)
26	Get BATCHRUNID	Returns BATCHRUNID if set from pipeline, otherwise takes default value		%python batchrunid=ds.textbox(name='\$BATCHRUNID\$', default_value='Batch_auto', label='BATCHRUNID') print(batchrunid)
27	Get TASKID	Returns TASKID if set from pipeline, otherwise takes default value		%python taskid=ds.textbox(name='\$TASKID\$', default_value='task1', label='TASKID') print(taskid)

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.		%python wallet_alias = "WALLET_ALIAS_NAME" import oracledb oracledb.init_oracle_client() conn = oracledb.connect(dsn=wallet_alias)

Table A-4 (Cont.) Miscellaneous Helper Python Scripts

Index	Name	Description	Prerequisite	Script
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	%python query="SELECT table_name FROM user_tables" cur = conn.cursor() cur.execute(que ry) print(cur.fetch all()) if cur: cur.close()
3	Close Connection Object		Connection Object should be created using any of the above methods	%python query="SELECT table_name FROM user_tables" cur = conn.cursor() cur.execute(que ry) print(cur.fetch all()) if cur: cur.close()
4	Close Cursor		Cursor Object should be created using any of the above methods	%python if cur: cur.close()
5	Check environment variables	For example: Check wallet alias value of MMG_LIB_WALLE T_ALIAS		%python import os print(os.enviro n['MMG_LIB_WALL ET_ALIAS'])

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='\$ FICMISDATES', 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/" + objectId + "/" + 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["me ssages"] 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	%python mmg.attachWorks pace(workspace) #eg- mmg.attachWorks pace("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 AIFenabled and attached
2	List Workspaces	Lists all workspaces available to a user	%python mmg.listWorkspa ces()	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	%python print(aif__isAt tached)	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	%python mmg.getConnection(schema_name) #eg- mmg.getConnection("OFSAA")	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
5	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
6	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
7	Load Flat File	%python mmg.loadFlatFile() #eg- mmg.loadFlatFile() Incomplete		
8	Profile Data		%python mmg.profileData(code, title) #eg- mmg.profileData(ABC, XYZ)	code = title =

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>	• 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>	"tasks" is options key. If not passed, then response will contain status all the tasks inside a batch.
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