Oracle®

Oracle Financial Services Model Management and Governance Installation and Configuration Guide





Oracle Oracle Financial Services Model Management and Governance Installation and Configuration Guide, Release 8.1.3.3.0

G18356-23

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

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

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

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

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

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

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

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

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

Contents

Preface	
Audience	1
Additional Resources	1
Conventions	2
Abbreviations	2
Introduction	
Components of Oracle Financial Services Model Management and Governance Installation Check List	1 2
Hardware and Software Requirements	
License Information	1
Preinstallation	
Oracle Database Instance Settings	1
Create the Database Schema on Oracle Database	1
Creating an Oracle User	1
Creating the Application Configuration Schema	2
Create the MMG Studio Schema	3
Create MMG Graph Schema	3
Create the Installation, Download, and Metadata Repository Directories	4
Configure the OS File System Settings and Environment Settings in the .profile File	5
Configure Operating System and File System Settings	5
Configure the Environment Settings	6
Java Settings	7
Oracle Database Server and Client Settings	7
TNS entries in the tnsnames.ora file for Non-TCPS	7
Time Zone Settings	8
Prerequisite for Data Pipeline	9
Setup Password Stores with Oracle Wallet	10
Setup the Password Stores for Database User Accounts	10

5 Installation

Prerequisites	1
Download the OFS MMG Installer Kit	1
Extract the Software	2
Configure the config.sh file	3
Import Server Certificate to Java Keystore	37
Run the MMG Installer	37
Modify the Logging Level	39
Starting MMG Services	41
Stopping MMG Services	42
Generate GRAPH-KEYSTORE.P12	42
Installing Conda	43
Installing Python Library	43
Prerequisites	43
Procedure	44
Setting up the Environment for Hive Data Sourcing	46
Remote MMG Studio Configuration	48
PGX Installation	49
Configure the config.sh File of PGX	50
Starting PGX Server	52
Stopping PGX Server	52
R Interpreter	52
Installing Oracle R Distribution	53
R 4.1.2 Installation	53
MMG Connection Objects Library Setup	53
Installing ROracle Library	53
Installing RODBC Library	54
Installing the IRkernel	55
Configuring R Interpreter	56
Using MMG Studio to Oracle Connection Objects	56
Workspaces	56
Connections	57
Multi Level Approval	57
Kafka Installation	58
Configuring Interpreters	61
Spark Interpreter	61
Spark Configuration and Execution in Python	61
Set the Hadoop Clusters	61
Configuration on the OFS MMG Instance Server	61

	Set Apache Spark in the OFS MMG Installed Server	62
	Code Snippet to Create the Spark Dataframe and Trigger Executions in to Spark	
	Cluster (with PySpark) Interpreter Configuration	62 65
	interpreter Configuration	03
6	Post-installation Steps	
	Access the Application	1
	Create Application Users	1
	Map Application User(s) to User Group	1
	User Access and Permissioning Management	3
	Mapping User Groups	4
	User Groups	4
	User Group - Role Mapping	5
	Functions and Roles required to perform CRUD operations for Conda	7
	Access MMG using AAI Realm	7
	Prerequisites	8
	Access MMG using SAMLRealm	9
	Access the Application by Using SAML Realm	10
	Access Data Studio Using SAML Realm	11
	Model Techniques/ Model Library	12
	.PEM file creation for Model Service	13
	AAI User Provisioning SQL Scripts Generator Utility	13
	IDCS Server Configuration	13
	Conda Environment Migration and Restoration	17
	Manual Configuration of Email Notification for Scheduler Batch Execution	19
	Fine Tuning Studio Sessions	20
7	Upgrade Installation	
	Upgrading to 8.1.3.3.0	1
8	Update Utility to Reconfigure Installation Parameters	
9	Cloning the MMG Instance	
	Copying the Directories	1
	Copying the Database Schemas	1
	Configuring Password Store with Oracle Wallet	1
	Updating the Host Details	2
	Update LOG_HOME and FTPSHARE	2
	Setting up the SSL Keystore	3

	Updating Wallet Aliases for Oracle Schemas	3
	Updating Context and Ports	3
	Starting MMG Services	4
	Cloning the Environment	4
10	Frequently Asked Questions (FAQs) and Error Dictionary	
	Frequently Asked Questions	1
	Frequently Asked Questions	1

Preface

This section provides information about the Oracle Financial Services Model Management and Governance (OFS MMG) Installation and Configuration Guide.

Topics:

Related Topics

- Audience
- Additional Resources
- Conventions
- Abbreviations

Audience

OFS MMG Installation and Configuration Guide is intended for Administrators and Implementation Consultants who handle installing and maintaining the Application Pack Components.

This document assumes that you have experience in installing Enterprise Components and basic knowledge about the following:

- OFS AAI Components
- OFSAA Architecture
- UNIX Commands
- Database Concepts
- Web Server or Web Application Server

Additional Resources

This section identifies additional resources to the OFS MMG Application. You can access the following documents from the Oracle Help Center:

- OFS Model Management and Governance Release Notes
- OFS Model Management and Governance User Guide

Additional related documents are as follows:

OFS Analytical Applications 8.1.2.0.0 Technology Matrix

Note

In the Tech Matrix, the Java Version column is no longer applicable for release 8.1.3.3.0. Instead, Java "17" 2021-09-14 LTS is applicable.



Conventions

The following text conventions are used in this document:

Table 1-1 Document Conventions

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, file names, text that appears on the screen, or text that you enter.
Hyperlink	Hyperlink type indicates the links to external websites, internal document links to sections.

Abbreviations

The following table lists the Abbreviations used in this document:

Table 1-2 Abbreviations

Abbreviation	Meaning
BDP	Big Data Processing
DBA	Database Administrator
DDL	Data Definition Language
DEFQ	Data Entry Forms and Queries
DML	Data Manipulation Language
EAR	Enterprise Archive
EJB	Enterprise JavaBean
ERM	Enterprise Resource Management
FTP	File Transfer Protocol
HDFS	Hadoop Distributed File System
HTTPS	Hypertext Transfer Protocol Secure
J2C	J2EE Connector
J2EE	Java 2 Enterprise Edition
JCE	Java Cryptography Extension
JDBC	Java Database Connectivity
JDK	Java Development Kit
JNDI	Java Naming and Directory Interface
JRE	Java Runtime Environment
JVM	Java Virtual Machine
LDAP	Lightweight Directory Access Protocol
LHS	Left Hand Side
MFA	Multi-Factor Authentication



Table 1-2 (Cont.) Abbreviations

Abbreviation	Meaning
MOS	My Oracle Support
OFSAA	Oracle Financial Services Analytical Applications
OFS AAI	Oracle Financial Services Analytical Application Infrastructure
OFS MMG	Oracle Financial Services Model Management and Governance Application
OHC	Oracle Help Center
OLAP	On-Line Analytical Processing
OLH	Oracle Loader for Hadoop
ORAAH	Oracle R Advanced Analytics for Hadoop
OS	Operating System
RAM	Random Access Memory
RDBMS	Relational Database Management System
RHEL	Red Hat Enterprise Linux
SFTP	Secure File Transfer Protocol
SID	System Identifier
SSL	Secure Sockets Layer
TNS	Transparent Network Substrate
URL	Uniform Resource Locator
VM	Virtual Machine
WAR	Web Archive
XML	Extensible Markup Language
PGX	Parallel Graph AnalytiX
FQDN	Fully Qualified Domain Name

Introduction

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.

Topics:

Related Topics

•

Components of Oracle Financial Services Model Management and Governance

Components of Oracle Financial Services Model Management and Governance

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

- Workspace Management
- Model Management
- Dataset
- Model Pipelines
- Model Actions
- Graphs
- Scheduler Service
- Audit Trail
- Data Studio Options
- Object Migration
- Model Training

For more information on how to use the application, see the <u>OFS Model Management and Governance User Guide</u>.



Installation Check List

To complete the installation process, you must perform the following steps listed in the Preinstall Checklist. Use this checklist to verify whether these steps are completed or not.

Table 2-1 Installation Checklist

SI. No.	Activity
Pre-installation Steps	
2	Install all the prerequisite hardware and software as per the OFS Analytical Applications 8.1.2.0.0 Technology Matrix. Configure the Database Instance Settings.
_	Comigare the Database Instance Cettings.
	In the Tech Matrix, the Java Version column is no longer applicable for release 8.1.3.3.0. Instead, Java 17.x is applicable.
3	Create the Installation, Download, and Metadata Repository Directories:
	Installation Directory Tampagary Directory
	Temporary DirectoryStaging Area/Metadata Repository
	Download Directory
4	Configure the following Operating System and File System Settings:
	File Descriptor
	Total number of processesPort(s)
	profile file permissions
	 Add FTP or SFTP Configuration for file transfer (to access Staging Area and Metadata Directory)
5	Update the following Environment Settings as required for the installation in the .profile file:
	 Java Settings Oracle Database Server and Client Settings
	 Add TNS entries in the TNSNAMES.ORA file
	 Time Zone Settings
Installation Steps	
6	Download the Installer Kit.
7 8	Extract the Installer Kit. Configure the config.sh file.
9	Trigger the Application Installation.



Table 2-1 (Cont.) Installation Checklist

SI. No.	Activity
Post-Installation Steps	
10	Access the MMG Application.
11	Create Application Users.
12	Map Application User(s) to User Groups.

(i) Note

MMG v8.1.3.3.0 supports Python 3.8.x, Python 3.9.x, Python 3.10.x, Python 3.11.x, and Python 3.12.x. Apache Flink packages are not supported with Python 3.12.x.

Hardware and Software Requirements

The following are the hardware and software requirements:

Table 3-1 Hardware and Software Requirements

Category	Version
Processor Type	Linux x86-64
OS Version	Oracle Linux Server release 8.4 and later versions
Java Version	OpenJDK 17.0.x
Database Server	Oracle Database 19c (19.3+) Enterprise Ed.
Java	Version 17 and 21
Oracle Database	Version 19 and 23
Python	Version 3.8-12
Python	Version 3.10
OS	OEL9, OEL8 and Solaris

License Information

For details on the third-party software tools used, see the <u>OFSAA Licensing Information User Manual Release 8.1.2.0.0</u>.

Preinstallation

This section lists all the prerequisites to install OFS MMG.

Oracle Database Instance Settings

Ensure that the following database instance settings are configured:

- NLS_CHARACTERSET to AL32UTF8
- NLS LENGTH SEMANTICS to BYTE
- OPEN CURSORS limit to greater than 1000

Create the Database Schema on Oracle Database

Create the following Database Schemas:

- MMG Studio Schema
- MMG Graph Schema

Tablespace

You can either use the existing Tablespace or can create a new Tablespace during schema creation using the following script:

Permanent Tablespace

```
CREATE TABLESPACE <tablespace_name >

DATAFILE '<tablespace_name >.dat'

SIZE 1G

ONLINE;
```

Temporary Tablespace

```
CREATE TEMPORARY TABLESPACE <tablespace_name >
TEMPFILE '<tablespace_name >.dbf'
SIZE 100M;
```

Creating an Oracle User

You can create an Oracle user using the following script:

CREATE USER <oracle_user_name> IDENTIFIED BY <password> DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE TEMP QUOTA <quota_size>|UNLIMITED ON USERS



Creating the Application Configuration Schema

Create the application schema. You must create an Oracle User to create the application schema. For more details, see Creating an Oracle User section. This section discusses the various grants required for the Oracle Database User.

Assign the Grants

Assign the following grants to the workspace schema user.

```
grant create SESSION to coracle_database_user>;
grant create PROCEDURE to <oracle_database_user>;
grant create SEQUENCE to cracle_database_user>;
grant create TABLE to coracle_database_user>;
grant create TRIGGER to <oracle_database_user>;
grant create VIEW to coracle_database_user>;
grant create MATERIALIZED VIEW to coracle_database_user>;
grant select on SYS.V_$PARAMETER to <oracle_database_user>;
grant create SYNONYM to <oracle_database_user>;
grant select on sys.v_$parameter to <oracle_database_user>;
grant select on sys.dba_free_space to <oracle_database_user>;
grant select on sys.dba_tables to <oracle_database_user>;
grant select on sys.Dba_tab_columns to <oracle_database_user>;
grant create RULE to coracle_database_user>;
grant create any trigger to <oracle_database_user>;
grant drop any trigger to <oracle_database_user>;
grant select on SYS.DBA RECYCLEBIN to <oracle database user>;
create or replace DIRECTORY fs_list_logs_dir AS '/file_store/fs_list/logs/';
grant read, write on DIRECTORY fs_list_logs_dir TO schema_name;
create or replace DIRECTORY fs_list_script_dir AS '/file_store/fs_list/script/';
grant read, execute on DIRECTORY fs_list_script_dir TO schema_name;
create or replace DIRECTORY fs_list_control_dir AS '/file_store/fs_list/
control/';
grant read on DIRECTORY fs_list_control_dir TO schema_name;
-- Directory creation and access grants
create or replace DIRECTORY fs_list_logs_dir AS '/file_store/fs_list/logs/';
grant read, write on DIRECTORY fs_list_logs_dir to &schemaname;
create or replace DIRECTORY fs_list_script_dir AS '/file_store/fs_list/
```



```
script/';
grant read, execute on DIRECTORY fs_list_script_dir to &schemaname;

create or replace DIRECTORY fs_list_control_dir AS '/file_store/fs_list/
control/';
grant read on DIRECTORY fs_list_control_dir to &schemaname;
```

Create the MMG Studio Schema

You must create an Oracle User to create the MMG Studio Schema. For more details, see Creating an Oracle User section.

Assign the following grants:

```
GRANT CONNECT, CREATE TABLE, CREATE VIEW, CREATE SEQUENCE TO <mmgstudio_schema_name>;
```

A new user group created in MMG which requires permission for Studio operations, is to be added in the following files:

- mmgstudio/conf/application.yml
- mmgstudio/conf/ofsaa-permissions.init.yml



Make sure the groups are in upper case, as AAI groups are always in upper case.

Create MMG Graph Schema

Ensure that create an Oracle User to create the MMG Graph Schema. For more details, see Creating an Oracle User section.

Assign Grants

This section discusses the various grants required for the Graph Schemas.

Assign the following grants for the schema:

1. Pre-installation grants for Graph Schema:

```
GRANT CREATE SESSION TO <GRAPH_SCHEMA>;

GRANT CREATE TABLE TO <GRAPH_SCHEMA>;

GRANT CREATE VIEW TO <GRAPH_SCHEMA>;

GRANT CREATE ANY PROCEDURE TO <GRAPH_SCHEMA>;

GRANT CREATE SEQUENCE TO <GRAPH_SCHEMA>;

GRANT CREATE JOB TO <GRAPH_SCHEMA>;

GRANT CREATE MATERIALIZED VIEW TO <GRAPH_SCHEMA>;

GRANT EXECUTE ON DBMS_SCHEDULER to <GRAPH_SCHEMA>;

GRANT EXECUTE ON DBMS_COMPARISON TO <GRAPH_SCHEMA>;
```



```
GRANT EXECUTE ON DBMS_RLS TO <GRAPH_SCHEMA>;

GRANT EXECUTE ON SYS.DBMS_SESSION TO <GRAPH_SCHEMA>;

GRANT EXECUTE ON DBMS_REDEFINITION TO <GRAPH_SCHEMA>;

GRANT REDEFINE ANY TABLE TO <GRAPH_SCHEMA>;

GRANT SELECT ON SYS.V_$PARAMETER TO <GRAPH_SCHEMA>;

GRANT SELECT ON <DATA_SOURCE_SCHEMA>.<TABLE_NAME> TO <GRAPH_SCHEMA>;
```

Example:

Change the <DATA_SOURCE_SCHEMA> to the schema used in the Graph pipeline.



If a user has to execute the custom graph, the same permissions have to be provided for the input tables referred in Custom Graph Pipeline.

Create the Installation, Download, and Metadata Repository Directories

To install the application, create the following directories:

- OFS MMG Download Directory (Optional): This is the directory where the downloaded installer or patches can be copied. Create a download directory and copy the OFS MMG Application Pack Installer File (archive). Assign 755 permission to this directory.
- **Temporary Directory:** This is the default temporary directory where the installation files are stored for a short time to support faster installation. Configure adequate space on the /tmp directory. It is recommended to allocate more than 10 GB of space. Assign 755 permission to this directory and disable the NOEXEC option.

(i) Note

If the NOEXEC option is enabled, the extraction of files by the installer into the / tmp directory is prevented and the binaries will not execute in the directory, hence resulting in failure of the installation.

- OFS MMG Installation Directory (Mandatory): Create an installation directory where the product binaries are installed. Assign 755-user permission to the Installation Directory.
- **OFS MMG Staging/Metadata Directory (Mandatory):** This is a directory to hold the application metadata artifacts and additionally, act as the staging area for the flat files. This directory is also referred to as FTPSHARE. Create a Staging or Metadata Repository Directory to copy data files, save data extracts, and so on. You can configure this directory on a different mount or under a different user profile.





(i) Note

Ensure that the OFS MMG Staging Directory is not set to the same path as the OFS MMG Installation Directory and is not a sub-directory inside the OFS MMG Installation Directory.

Configure the OS File System Settings and Environment Settings in the .profile File

A .profile file is a start-up file of a UNIX User. Create the .profile file at the home directory of the logged-in user if it is not already available. The user must have 755 permission on the file to execute it. This file consists of various parameters for Environment Settings, OS, and File System Settings.

To set the parameters for the .profile file, login as a non-root user, and configure the environment settings.



🚹 Warning

Do not modify any other parameters other than the parameters mentioned in the following subsections.

Configure Operating System and File System Settings

To install the application, configure the operating system and file system settings refer the parameters and configuration actions.



Table 4-1 Configure operating system and file system settings

ge the number of is the root user:
etc/sysctl.con
000
the following
,

(i) Note

The value specified here is the minimum value to be set for the installation process to go forward. For other modules, this value may depend on the available resources and the number of processes executed in parallel.

Total Number of Process Settings

In the sysctl.conf file, set the value to greater than 4096.



The value specified here is the minimum value to be set for the installation process to go forward. For other modules, this value may depend on the available resources and the number of processes executed in parallel.

Configure the Environment Settings

Environment Settings refers to values related to the current environment, like the Operating System or user sessions. To configure the environment settings refer the following topics.



Java Settings

To configure the Java Settings, refer the following table:



(i) Note

In the application, Java 21 is supported.

Table 4-2 Java Settings

Description	Example Value
In the .profile file, set JAVA_BIN to include the JDK absolute path.	For example: JAVA 17

Table 4-3 Java Settings

Description	Example Value
In the .profile file, set the Java tool options for all versionsJDK-17 and above updates. Ensure that SYMBOLIC links to JAVA installation are not set in the PATH variable.	JAVA_TOOL_OPTIONS=" - Djdk.util.zip.disableZip64ExtraFieldVal idation=true" export JAVA_TOOL_OPTIONS
In the .profile file, set JAVA_BIN to include the JDK absolute path.	For example: JAVA_BIN =/usr/java/ jdk-17/bin export JAVA_BIN

Oracle Database Server and Client Settings

To configure the Oracle Database Server and Client Settings, refer to the following table:

Table 4-4 Oracle Database Server and Client Settings

Description	Example Value
In the .profile file, set TNS_ADMIN pointing to the appropriate tnsnames.ora file.	TNS_ADMIN=\$HOME/tns
In the .profile file, set ORACLE_HOME pointing to the appropriate Oracle Client installation.	ORACLE_HOME=/scratch/oraofss/ app_client19c/product/ 19.0.0/client_1
In the .profile file, set PATH to include the appropriate \$ORACLE_HOME/bin path.	PATH=\$JAVA_HOME/bin:\$ORACLE_HOME/bin

TNS entries in the tnsnames.ora file for Non-TCPS

You must configure the TNS entries in the tnsnames.ora file for Non-TCPS.

Non-TCPS

To configure the TNS entries in the tnsnames.ora file for Non-TCPS, refer to the following table:



Table 4-5 TNS entries in the TNSNAMES.ORA file for Non-TCPS


```
<SID NAME> =

(DESCRIPTION =

(ADDRESS_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = <HOST NAME>)(PORT = <PORT NUMBER>))
) (CONNECT_DATA =

(SERVICE_NAME = <SID NAME>)
)

<ATOMICSCHEMANAME> =

(DESCRIPTION =

(ADDRESS_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = <HOST NAME>)(PORT = <PORT NUMBER>))
)

(CONNECT_DATA =

(SERVICE_NAME = <SID NAME>)
)
```

Time Zone Settings

In the <code>.profile</code> file, set the Time Zone Parameter to indicate the time zone of your region or location.

For more information, see Oracle Financial Service Stress Testing and Scenario Analytics User's Guide.



Table 4-6 Time Zone Settings

Description	Example Value
Time Zone	TZ=Asia/Calcutta

Prerequisite for Data Pipeline

The below are prerequisites for Data Pipeline. The steps need to be executed before triggering MMG Installer.

- Log in as DB user(ex: fccmdb) in your DB server (For example: 1759).
- Create one directory with preferred reference name (For example: datapipline) at the preferred location.
- 3. Now execute below commands in putty session of DB server:
 - mkdir -p ##location of dir created in step 2##/file_store/ fs_list/logs # A directory by the external table to write the logs.
 - mkdir -p ##location of dir created in step 2##/file_store/ fs_list/script # A directory to hold a pre-processor script used to list the files in a directory. This needs Read-Execute permissions.
 - mkdir -p ##location of dir created in step 2##/file_store/ fs_list/control # A directory to hold files to control which directories can be listed. This needs Read permissions.
 - mkdir -p ##location of dir created in step 2##/file_store/ fs list/fccm-data # A directory to store the csv files.
- 4. After that login to DB server(For example: 1759) as sysadmin(For example: sys) in sqldeveloper or any preferred application execute below:
 - CREATE OR REPLACE DIRECTORY fs_list_logs_dir AS '##location of dir created in step 2##/file_store/fs_list/logs';
 - GRANT READ, WRITE ON DIRECTORY fs_list_logs_dir TO ##MMG Application Schema##;
 - CREATE OR REPLACE DIRECTORY fs_list_script_dir AS '##location of dir created in step 2##/file store/fs list/script';
 - GRANT READ, EXECUTE ON DIRECTORY fs_list_script_dir TO ## MMG Application Schema##;
 - CREATE OR REPLACE DIRECTORY fs_list_control_dir AS '##location of dir created in step 2##/file store/fs list/control';
 - GRANT READ ON DIRECTORY fs_list_control_dir TO ##MMG Application Schema##;
 - CREATE OR REPLACE DIRECTORY external_tables_dir AS '##location of dir created in step 2##/file_store/fs_list/fccm-data';
 - GRANT READ ON DIRECTORY external_tables_dir TO ## MMG Application Schema##;
 - GRANT CREATE TABLE TO ##MMG Application Schema##;

Once the above steps are done, trigger the installation of Compliance Studio build.



(i) Note

- If the new schema is residing on any-other DB server other than the one on which above steps were performed. Please perform all the above steps on that respective server.
- Once configured the DB server, if new schema is created on it. Need to perform the step for the respective schema.

Setup Password Stores with Oracle Wallet

This section describes the steps to create a wallet and the aliases for the database user accounts. For more information on configuring authentication and password stores, see theOracle Database Security Guide.

As part of an Application Installation, Administrators must set up password stores for Database User Accounts using Oracle Wallet. These password stores must be installed on the Application Database side. The Installer handles much of this process. The Administrators must perform some additional steps.

Setup the Password Stores for Database User Accounts

After the database is installed and the default Database User Accounts are set up, Administrators must set up a password store using the Oracle Wallet. This involves assigning an alias for the username and associated password for each Database User Account. The alias is used later during the application installation. This password store must be created on the system where the Application Server and database client are installed.

(i) Note

In this section, <wallet location> is a placeholder text for illustration purposes. Before running the command, ensure that you have already created the <wallet location> directory where you want to create and store the wallet.

The wallet is created in the <wallet location> directory with the auto-login feature enabled. This feature enables the database client to access the wallet contents without using the password. For more information, see Oracle Database Security Guide.

To create a wallet, follow these steps:

- Login to the server as a Linux user.
- Create a wallet in the <wallet_location> using the following command: mkstore -wrl <wallet_location> -create

(i) Note

The mkstore utility is included in the Oracle Database Client Installation.

After you run the command, a prompt appears. Enter a password for the Oracle Wallet in the prompt.

A prompt appears to re-enter the password. Re-enter the password.



Figure 4-1 Wallet Creation

```
S mkstore -wrl /scratch/ofsasftp/pgx server/wallet -create
Oracle Secret Store Tool Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
Copyright (c) 2004, 2019, Oracle and/or its affiliates. All rights reserved.

Enter password:
Enter password again:
S mkstore -wrl /scratch/ofsasftp/pgx server/wallet -createCredential MMGConfigSchema_Alias MMG_Config_Schema
Oracle Secret Store Tool Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
Copyright (c) 2004, 2019, Oracle and/or its affiliates. All rights reserved.

Your secret/Password is missing in the command line
Enter your secret/Password:
Re-enter your secret/Password:
Re-enter your secret/Password:
Enter your secret/Password:
Enter wallet password:
S mkstore -wrl /scratch/ofsasftp/pgx_server/wallet -createCredential MMGStudio_Schema_Alias MMG_Studio_Schema
Oracle Secret Store Tool Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
Copyright (c) 2004, 2019, Oracle and/or its affiliates. All rights reserved.

Your secret/Password is missing in the command line
Enter your secret/Password:
Enter your secret/Password:
Enter your secret/Password:
Enter wallet password:
S mkstore -wrl /scratch/ofsasftp/pgx_server/wallet -createCredential MMGGraphSchema_Alias MMG_Graph_Schema
Oracle Secret Store Tool Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
Copyright (c) 2004, 2019, Oracle and/or its affiliates. All rights reserved.

Your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password:
Enter your secret/Password is missing in the command
```

4. Create the database connection credentials for the MMG Schema using the following command:

mkstore -wrl <wallet_location> -createCredential <alias-name> <mmgschema-name>

Here, MMG Schema is the same as explained in Create the MMG Schema section.

After you run the command, a prompt appears. Enter the password associated with the Database User Account in the prompt. You are prompted to re-enter the password. You are prompted for the wallet password used in Step 1.

Note

In this manner, create a wallet and associated database connection credentials for all the Database User Accounts including Graph and Studio Schema.

After the wallet is created, go to the <wallet_location> directory and click Refresh to view the created wallet folder.

The wallet folder contains two files: **ewallet.p12** and **cwallet.sso**.

After you run the command, a prompt appears. Enter the password associated with the Database User Account in the prompt. You are prompted to re-enter the password. You are prompted for the wallet password used in Step 1.

(i) Note

In this manner, create a wallet and associated database connection credentials for all the Database User Accounts including Graph and Studio Schema.

After the wallet is created, go to the <wallet_location> directory and click Refresh to view the created wallet folder.





(i) Note

In addition to creating wallet aliases for MMG Studio and Graph, you must also create wallet aliases for the production schemas; specifically, the config and atomic schemas. Ensure these aliases are added to the tnsnames.ora file.

The wallet folder contains two files: **ewallet.p12** and **cwallet.sso**.

In the <wallet_location> directory, configure the tnsnames.ora file to include the entry for each alias name to be set up. Alias name can be renamed as wallet db alias name.

(i) Note

- You can either update the existing thsnames.ora file with the above details or create new trsnames.ora file and add the required entries.
- <alias-name> is a user-defined value.
- 6. Create a sqlnet.ora file in the wallet directory using the following content:

```
WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA =
(DIRECTORY = <Wallet Location>)) ) SQLNET.WALLET OVERRIDE=TRUE
SSL_CLIENT_AUTHENTICATION=FALSE
```

Here, <Wallet_Location> should be in below format:

```
WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA =
(DIRECTORY = <WALLET PATH>) ) )
```

Verify the Connectivity of the Wallet

To verify the connectivity of the wallet, follow these steps:

Test the connectivity using the following command:



(i) Note

The ORACLE HOME used with the wallet must be the same version or higher than the wallet created.

```
$ export WALLET_LOCATION=<wallet_location>
$ export TNS ADMIN=<tnsnames.ora location>. If you have created a new
tnsnames.ora file, provide the location of the new file.
$ sqlplus /@<alias_name>
```

The output is similar to:

```
SQL*Plus: Release 11
Connected to:
Oracle Database 12c
```

To verify if you are connected to the correct user:



SQL> show user

The output is similar to:

USER is "<database-user-name>"

Installation

This section provides detailed steps to install the application.

Prerequisites

The list of prerequisites are as follows:

Port Details

MMG Studio uses the following ports by default, so make sure these are free:

Graph Service: 7059

Server: 7008

Markdown-Interpreter: 7009, 7029 Python-Interpreter: 7012, 7032, 6012

Shell-Interpreter: 7013, 7033

Plainr-Interpreter 7019, 7039, 6311 (Rserve port- configurable in plainr.Json)

PGX Server: 7007

PGX-Interpreter: 7022, 7042 JDBC-Interpreter: 7011, 7031 Coherence Cluster: 7574

(i) Note

These are default ports provided in the installation kit. You can modify them in the configuration files if required to avoid conflicts with existing services.

Enhancements for Preinstall Check

Warning messages for below validations:

- To validate whether the given hostname is valid or not.
- To validate whether given hostname and hostname in keystore are matching.
- Error messages and failing the preinstall check for below validations.
- To check sso_token is present or not.
- To check for public and private key are present in the mmghome/conf location.

Download the OFS MMG Installer Kit

To download the software as a .zip folder, download the mandatory minor release patch **38481933** from My Oracle Support (MOS).



Download the installer archive and copy (in Binary Mode) to the download directory that exists in the OFS MMG Installation Setup.

Extract the Software

You must be logged in to the UNIX Operating System as a Non-Root User to perform the following steps. To extract the software, follow these steps:

1. Download the unzip (OS-specific) unzip_<os>.zip and copy it in Binary Mode to the directory that is included in your PATH variable.

If you already have an unzip utility to extract the contents of the downloaded archive, skip this step. Uncompress the unzip installer file with the command:

uncompress unzip_<os>.Z

(i) Note

If an error message "uncompress: not found [No such file or directory]" is displayed, contact your UNIX Administrator.

2. Assign execute (751) to the file with the following command:

chmod 751 unzip_<OS>

For example: chmod 751 unzip sparc

3. Extract the contents of the OFS MMG Application Pack Release 8.1.3.3.0 installer archive file in the download directory with the following command:

```
unzip OFS_MMG_8.1.3.3.0_<OS>.zip
```

After unzipping the OFS_MMG_8.1.3.3.0 folder, following zip folders are displayed under OFS MMG folder.

- mmg-installer.zip
- mmg-metadata-manager.zip
- mmg-pgx.zip
- OFSMMG_8.1.3.3.0_Readme.html

Unzip mmg-installer.zip and the following components are available under / OFS_MMG/mmg-installer.zip:

- mmg-ui
- mmg-studio
- mmg-service
- mmg-schema-creator
- mmg-pipeline
- · mmg-load-to-graph
- lib
- bin
- conf
- mmg-gateway



- OFSMMG 8.1.3.3.0 Readme.html
- **4.** Navigate to the download directory and assign execute permission to the installer directory with the following command:

```
chmod -R 750 OFS_MMG
```

Configure the config.sh file

To configure the config.sh file for installing the application, perform the following steps:

- 1. Log in to the server as a non-root user.
- 2. Navigate to the <installation directory>/OFS_MMG/MMG/OFS_MMG/bin directory.
- 3. Configure the following applicable config.sh attributes:

```
Sample config.sh files:
```

```
#!/bin/sh ##
## Common properties
export APPLICATION_NAME=##APPLICATION_NAME##
export WALLET_LOCATION=##WALLET_LOCATION##
export TNS ADMIN PATH=##TNS ADMIN PATH##
export WALLET_ALIAS=##WALLET_ALIAS## export LOG_HOME=##LOG_HOME##
export LOG HOME=##LOG HOME##
export FTPSHARE=##FTPSHARE##
export LOG_TIMEZONE=##LOG_TIMEZONE##
##By default, Data Studio is assumed to run on the same server. If it is
running on a different host, uncomment the line below and provide the
appropriate URL. #export DATASTUDIO_URL=##DATASTUDIO_URL##
export BE_HOSTNAME=##
BE HOSTNAME##
export BE PORT=##BE PORT##
# Add SMTP host and port parameters for email notifications
export SMTP_HOST=##SMTP_HOST##
export SMTP_PORT==##SMTP_PORT##
export UI_PORT=##UI_PORT##
export SCHEMA_PORT=##SCHEMA_PORT##
export CONTEXT_PATH=##CONTEXT_PATH##
export STUDIO_AUTH_TYPE=##STUDIO_AUTH_TYPE##
export
RESTRICT UNMAPPED DATASTORES ACCESS=##RESTRICT UNMAPPED DATASTORES ACCESS##
export SSL_ENABLED=##SSL_ENABLED##
export SSL_KEYSTORE=##SSL_KEYSTORE##
export SSL_KS_SECRET=##SSL_KS_SECRET##
export SSL_KS_TYPE=##SSL_KS_TYPE##
```



- export SSL_KS_ALIAS=##SSL_KS_ALIAS##
- export SESSION_TOKEN_CREDENTIALS=##SESSION_TOKEN_CREDENTIALS##
- export FCC_API_USER=##FCC_API_USER##
- export MMG_DATASOURCE_MAX_POOL_SIZE=##MMG_DATASOURCE_MAX_POOL_SIZE##
- export MMG_DATASOURCE_MINIMUM_IDLE=##MMG_DATASOURCE_MINIMUM_IDLE##
- export MMG_DATASOURCE_IDLE_TIMEOUT=##MMG_DATASOURCE_IDLE_TIMEOUT##
- export MMG_DATASOURCE_CONN_TIMEOUT=##MMG_DATASOURCE_CONN_TIMEOUT##
- export EXT_DATASOURCE_MAX_POOL_SIZE=##EXT_DATASOURCE_MAX_POOL_SIZE##
- export EXT_DATASOURCE_MINIMUM_IDLE=##EXT_DATASOURCE_MINIMUM_IDLE##
- export EXT DATASOURCE IDLE TIMEOUT=##EXT DATASOURCE IDLE TIMEOUT##
- export EXT_DATASOURCE_CONN_TIMEOUT=##EXT_DATASOURCE_CONN_TIMEOUT##
- export MMG HTTP MAX CONN=##MMG HTTP MAX CONN##
- export MMG_HTTP_MAX_CONN_PER_ROUTE=##MMG_HTTP_MAX_CONN_PER_ROUTE##
- export MMG_HTTP_CONNECT_TIMEOUT=##MMG_HTTP_CONNECT_TIMEOUT##
- export MMG_HTTP_READ_TIMEOUT=##MMG_HTTP_READ_TIMEOUT##
- export APPLICATION ID=##APPLICATION ID##

Properties for MMG-ui

export APPLICATION_FAVICON_PATH=##APPLICATION_FAVICON_PATH##

#export UI AUTH TYPE=aai

- export UI_AUTH_TYPE=##AUTH_TYPE##
- export AAI_AUTH_URL=##AAI_AUTH_URL##
- export SAML_IDP_URL=##SAML_IDP_URL##
- export SAML_SP_ENTITY=##SAML_SP_ENTITY##
- export SAML_SRV_URL=##SAML_SRV_URL##
- export SAML_LOGOUT_URL=##SAML_LOGOUT_URL##
- export SAML SIGN AUTHN REQ=##SAML SIGN AUTHN REQ##
- export SAML_PRIVATE_KEY_PATH=##SAML_PRIVATE_KEY_PATH##
- export SAML_SP_X509_CERT_PATH=##SAML_SP_X509_CERT_PATH##
- export SAML_SIGN_ALGORITHM=##SAML_SIGN_ALGORITHM##
- export AUTH_SAML_REQUEST_TYPE=##AUTH_SAML_REQUEST_TYPE##
- export AUTH_SAML_INCLUDE_SP_CERT=##AUTH_SAML_INCLUDE_SP_CERT##
- export LDAP URL=##LDAP URL##
- export LDAP_SEARCH_BASE=##LDAP_SEARCH_BASE##
- export LDAP_USER_FILTER=##LDAP_USER_FILTER##
- export LDAP USER SEARCH FILTER=##LDAP USER SEARCH FILTER##
- export LDAP_GROUP_SEARCH_FILTER=##LDAP_GROUP_SEARCH_FILTER##
- export LDAP_GROUP_SEARCH_BASE=##LDAP_GROUP_SEARCH_BASE##
- export LDAP_GROUP_MEMBER=##LDAP_GROUP_MEMBER##



① Note

Accepted Values for SAML SIGN ALGORITHM Property:

- http://www.w3.org/2001/04/xmldsig-more#rsa-sha512
- http://www.w3.org/2001/04/xmldsig-more#rsa-sha384
- http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
- http://www.w3.org/2000/09/xmldsig#rsa-sha1

Properties for enabling AAI Authorization. Applicable for SAML/LDAP Profiles

```
## AAI_AUTH_URL must be set.
```

```
export AAI AUTHZ ENABLED=##export AAI AUTHZ ENABLED##
```

export AAI_CLIENT_ID=##export AAI_CLIENT_ID##

export AAI_CLIENT_SECRET=<Instance Access Token>

export SERVER_COOKIE_DOMAIN=##SERVER_COOKIE_DOMAIN##

export SERVER COOKIE NAME=##SERVER COOKIE NAME##

export SERVER_COOKIE_TIMEOUT=##SERVER_COOKIE_TIMEOUT##

export SERVER COOKIE IS SECURE=##SERVER COOKIE IS SECURE##

##Properties for MMG-Service

export BE_AUTH_TYPE=public

export MMG_PYTHON_INTERPRETER=##MMG_PYTHON_INTERPRETER##

#Properties for enabling git based import and export of model pipelines

#set username and PAT(personal access token) to access private remote repositories

#set proxy url and port if you are behind a firewall

export GIT_ENV_ID=##GIT_ENV_ID##

export GIT_USERNAME=##GIT_USERNAME##

export GIT_PAT_SECRET=##GIT_PAT_SECRET##

export GIT_PROXY_URL=##GIT_PROXY_URL##

export GIT_PROXY_PORT=##GIT_PROXY_PORT##

(i) Note

Only tokens (classic) is supported now. Fine grained is not yet supported.

##Properties for MMG-Studio

export DATASTUDIO_SCHEMA_WALLET_ALIAS=##DATASTUDIO_SCHEMA_WALLET_ALIAS##

export LOGIN_SHOW=##LOGIN_SHOW##

export SESSION_MODE=##SESSION_MODE##

export STUDIO_REALM=##STUDIO_REALM##



```
export OFSAA_URL=##OFSAA_URL##
```

export API_USERS=##API_USERS##

export VALID_ROLES=##VALID_ROLES##

export STUDIO LOG LEVEL=##STUDIO LOG LEVEL##

export PYTHON_HOME=##PYTHON_HOME##

#Following are spark specific configurations, Leave as it is if not applicable

export MMG_SPARK_ENABLED=##MMG_SPARK_ENABLED## #will be false by default

export SPARK_HOME=##SPARK_HOME## #required if MMG_SPARK_ENABLED is true

export HADOOP_HOME=##HADOOP_HOME## #only needed if running spark with hadoop cluster

export SPARK MASTER=##SPARK MASTER## #required if MMG SPARK ENABLED is true

export SPARK_DEPLOY_MODE=##SPARK_DEPLOY_MODE## #required if MMG_SPARK_ENABLED
is true

export DATASTUDIO_SPARK_INTERPRETER_PORT=##DATASTUDIO_SPARK_INTERPRETER_PORT##
#required if MMG_SPARK_ENABLED is true

export R_ENABLED=##R_ENABLED## export R_PYTHON_HOME=##R_PYTHON_HOME##

export R_PYTHON_HOME=##R_PYTHON_HOME##

Following are fcc services specific configurations, Leave as it is if not applicable

export TEMPLATE_CONFIG_PATH=##TEMPLATE_CONFIG_PATH##

export TEMPLATE_DEFAULT_LINK=##TEMPLATE_DEFAULT_LINK##

export AUTH_SERVICE_URL=##AUTH_SERVICE_URL##

export META_SERVICE_URL=##META_SERVICE_URL##

export ER_SERVICE_URL=##ER_SERVICE_URL##

export BATCH_SERVICE_URL=##BATCH_SERVICE_URL##

export SAML_ISSUER=##SAML_ISSUER##

export SAML_DESTINATION=##SAML_DESTINATION##

export SAML_ASSERTION=##SAML_ASSERTION##

export SAML_ROLE_ATTRIBUTE=##SAML_ROLE_ATTRIBUTE##

export SAML_STUDIO_LOGOUT_URL=##SAML_STUDIO_LOGOUT_URL##

export SAML_COOKIE_DOMAIN=##SAML_COOKIE_DOMAIN##

Following are pipeline services specific configurations, Leave as it is if not applicable

export DATAPIPELINE_SERVICE_PORT1=##DATAPIPELINE_SERVICE_PORT1##

export DATAPIPELINE_SERVICE_PORT2=##DATAPIPELINE_SERVICE_PORT2##

export

DATAPIPELINE_METADATA_ARCHIVE_PATH=##DATAPIPELINE_METADATA_ARCHIVE_PATH##

export

DATAPIPELINE_METADATA_IMPORT_SERVICE_PORT=##DATAPIPELINE_METADATA_IMPORT_SERVICE_PORT##



```
export DATAPIPELINE_ERXMLPATH=##DATAPIPELINE_ERXMLPATH##
```

export DATAPIPELINE_GATEWAY_SERVICE_PORT=##DATAPIPELINE_GATEWAY_SERVICE_PORT##

export PIPELINE_UI_SERVICE_PORT=##PIPELINE_UI_SERVICE_PORT##

export DATA_PIPELINE_UI_SERVICE_PORT=##DATA_PIPELINE_UI_SERVICE_PORT##

export DATAPIPELINE_HAZELCAST_PORT=##DATAPIPELINE_HAZELCAST_PORT##

#URLS for pipeline, ER and matching service. Leave as it is if not applicable. Will impact the pipeline that could be added to a graph

export MATCHRULE BASE URL=##MATCHRULE BASE URL##

export LOADGRAPH_BASE_URL=##LOADGRAPH_BASE_URL##

export MATCHSRVC_UI_URL=##MATCHSRVC_UI_URL##

#URLS for index service. Leave as it is if not applicable. Will impact the pipeline that could be added to a graph

export GRAPH_INDEX_BASE_URL=##GRAPH_INDEX_BASE_URL##

export LOADINDEX_UI_URL=##LOADINDEX_UI_URL##

export MATCHING_MECHANISM=##MATCHING_MECHANISM##

export CANDIDATE_SELECTION_SERVICE_URL=##CANDIDATE_SELECTION_SERVICE_URL##

export LOAD_TO_OS_URL=##LOAD_TO_OS_URL##

#Changes for auth services + mmg keys

export AAI_COOKIE_DOMAIN=##AAI_COOKIE_DOMAIN##

export MMG_KEYS_LOC=##MMG_KEYS_LOC##

#Properties to package Load to Graph (L2G) service inside MMG ## Start of L2G Properties

export GRAPH_INSTALLATION_PATH=##GRAPH_INSTALLATION_PATH##

export GRAPH_KEYSTORE_PASSWORD=##GRAPH_KEYSTORE_PASSWORD##

export GRAPH_SERVICE_PORT=##GRAPH_SERVICE_PORT##

#Graph HikariCP Configuration Placeholders

export GRAPH_HIKARI_CP_MIN_IDLE=##GRAPH_HIKARI_CP_MIN_IDLE##

export GRAPH_HIKARI_CP_MAX_POOL_SIZE=##GRAPH_HIKARI_CP_MAX_POOL_SIZE##

export GRAPH_HIKARI_CP_IDLE_TIMEOUT=##GRAPH_HIKARI_CP_IDLE_TIMEOUT##

export GRAPH_HIKARI_CP_MAX_LIFETIME=##GRAPH_HIKARI_CP_MAX_LIFETIME##

export GRAPH_HIKARI_CP_TIMEOUT=##GRAPH_HIKARI_CP_TIMEOUT##

Graph Schema Configurations

export MMG_DB_SERVER_NAME=##MMG_DB_SERVER_NAME##

export MMG_DB_PORT=##MMG_DB_PORT##

export MMG_DB_SERVICE_NAME=##MMG_DB_SERVICE_NAME##

export PGX_SERVER_URLS=##PGX_SERVER_URLS##

PGX data memory limits configurations ## Overall Configuration

export MAX_TOTAL_SHARED_DATA_MEMORY_SIZE=##MAX_TOTAL_SHARED_DATA_MEMORY_SIZE##



```
export
```

MAX_TOTAL_PRIVATE_DATA_MEMORY_SIZE=##MAX_TOTAL_PRIVATE_DATA_MEMORY_SIZE##

export MAX_PER_SESSION_DATA_MEMORY_SIZE=##MAX_PER_SESSION_DATA_MEMORY_SIZE##
Role wise data memory limits

export MAX_DATA_MEMORY_SIZE_DSUSRGRP=##MAX_DATA_MEMORY_SIZE_DSUSRGRP##

export MAX DATA MEMORY SIZE DSBATCH=##MAX DATA MEMORY SIZE DSBATCH##

export MAX_DATA_MEMORY_SIZE_DSINTER=##MAX_DATA_MEMORY_SIZE_DSINTER##

export MAX DATA MEMORY SIZE DSAPPROVER=##MAX DATA MEMORY SIZE DSAPPROVER##

export MAX_DATA_MEMORY_SIZE_DSUSER=##MAX_DATA_MEMORY_SIZE_DSUSER##

export MAX_DATA_MEMORY_SIZE_IHUSRGRP=##MAX_DATA_MEMORY_SIZE_IHUSRGRP##

export GRAPH_SERVICE_CACHE_SERVER_PORT=##GRAPH_SERVICE_CACHE_SERVER_PORT##

#end of Properties configurations for L2G

##Schema details for graph service. This is configured as a temporary/target space for DP to create target tables which will act as input to L2G

export GRAPH SCHEMA WALLET ALIAS=##GRAPH SCHEMA WALLET ALIAS##

export GRAPH_SCHEMA_DB_SCHEMA_NAME=##GRAPH_SCHEMA_DB_SCHEMA_NAME##

#Additional MMG Features

export

MMG_MODEL_PIPELINE_SANDBOX_DEFAULT_VIEW=##MMG_MODEL_PIPELINE_SANDBOX_DEFAULT_VIEW##

The following properties are optional and enabled by default. If needed, you can uncomment them and set them to false.

#export MMG_HTTP2_ENABLED=##MMG_HTTP2_ENABLED##

#export MMG_SERVER_ACCESS_LOG_ENABLED=##MMG_SERVER_ACCESS_LOG_ENABLED##

The following properties are optional and disabled by default. If needed, you can uncomment them and set them to true.

#export OJET_CDN_ENABLED=##OJET_CDN_ENABLED##

##The following Properties are related to EST

export EST ENABLED=##EST ENABLED##

export EST_UI_URL=##EST_UI_URL##

export EST_SERVICE_URL=##EST_SERVICE_URL##

export DATACATALOG_SERVICE_URL=##DATACATALOG_SERVICE_URL##

##DEFAULT CONNECTION POOLING CONFIGURATION FOR EXTERNAL SCHEMA (Time in ISO-8601 format)

export GRAPH_EXT_SCHEMA_ENABLE_CP=##GRAPH_EXT_SCHEMA_ENABLE_CP##

export GRAPH_EXT_SCHEMA_CP_MAX_IDLE=##GRAPH_EXT_SCHEMA_CP_MAX_IDLE##

export GRAPH_EXT_SCHEMA_CP_MIN_IDLE=##GRAPH_EXT_SCHEMA_CP_MIN_IDLE##

export GRAPH_EXT_SCHEMA_CP_INITIAL_SIZE=##GRAPH_EXT_SCHEMA_CP_INITIAL_SIZE##

export GRAPH_EXT_SCHEMA_CP_MAX_TOTAL=##GRAPH_EXT_SCHEMA_CP_MAX_TOTAL##

export

GRAPH_EXT_SCHEMA_CP_MAX_WAIT_MILLIS=##GRAPH_EXT_SCHEMA_CP_MAX_WAIT_MILLIS##



export

GRAPH_EXT_SCHEMA_CP_MIN_EVICTABLE_IDLE_TIME=##GRAPH_EXT_SCHEMA_CP_MIN_EVICTABL
E_IDLE_TIME##

export

GRAPH_EXT_SCHEMA_CP_SOFT_MIN_EVICTABLE_IDLE_TIME=##GRAPH_EXT_SCHEMA_CP_SOFT_MIN_EVICTABLE_IDLE_TIME##

export GRAPH_DOCUMENT_DOWNLOAD_PATH=##GRAPH_DOCUMENT_DOWNLOAD_PATH##

##Proxy Configurations

export GRAPH NO PROXY=##GRAPH NO PROXY##

##Data Studio Ports

Following are the default ports 7008, 7009, 7012,-1. If needed, you can change the port numbers other than the default values.

export DATASTUDIO_SERVER_PORT=##DATASTUDIO_SERVER_PORT##

export

DATASTUDIO_MARKDOWN_INTERPRETER_PORT=##DATASTUDIO_MARKDOWN_INTERPRETER_PORT##

export

DATASTUDIO_PYTHON_INTERPRETER_PORT=##DATASTUDIO_PYTHON_INTERPRETER_PORT##

export DATASTUDIO_R_INTERPRETER_PORT=##DATASTUDIO_R_INTERPRETER_PORT##

export DATASTUDIO_JDBC_INTERPRETER_PORT=##DATASTUDIO_JDBC_INTERPRETER_PORT##

export

DATASTUDIO_PYTHON_INTERPRETER_REST_SERVER_PORT=##DATASTUDIO_PYTHON_INTERPRETER REST SERVER PORT##

DATASTUDIO_PGX_PYTHON_INTERPRETER_REST_SERVER_PORT=##DATASTUDIO_PGX_PYTHON_INTERPRETER REST SERVER PORT##

export

DATASTUDIO_THRIFT_EVENT_HANDLER_PORT=##DATASTUDIO_THRIFT_EVENT_HANDLER_PORT##

export DATASTUDIO_PGX_INTERPRETER_PORT=##DATASTUDIO_PGX_INTERPRETER_PORT##

export MMG_COHERENCE_CLUSTER_PORT=##MMG_COHERENCE_CLUSTER_PORT##

##MMG Gateway Configuration

export MMG_GATEWAY_PORT=##MMG_GATEWAY_PORT##

The following property can be set to control the pages where MMG can be embedded:

Set to 'self' to allow embedding only from the same origin (recommended for most setups).

Set to 'all' or '*' to allow embedding from any origin. (less secure)

Set to a comma-separated list of origins to allow embedding from those specified origins and from the same origin.

By default, this is set to 'self'. export
MMG CSP FRAME ANCESTORS=##MMG CSP FRAME ANCESTORS##

If the Gateway is enabled, this property can be set to control the pages where Data Studio can be embedded:

Set to '*' to allow embedding from any origin (less secure).

Set to a comma-separated list of origins to allow embedding from those specified origins and from the same origin.



By default, this is set to MMG Gateway URL.

If a load balancer or an external gateway is configured for MMG Gateway, the URL must be included in the list of origins.

export DATASTUDIO_CSP_FRAME_ANCESTORS=##DATASTUDIO_CSP_FRAME_ANCESTORS##

##MMG Model Serving Configuration

export

MMG_MODEL_ENDPOINT_RESTART_INTERVAL=##MMG_MODEL_ENDPOINT_RESTART_INTERVAL##

##Enhanced Configurability for Data Studio Performance and Session Parameters

Cleanup timeout for Jobs

export

DATASTUDIO JOBS INCOMPLETE TIMEOUT=##DATASTUDIO JOBS INCOMPLETE TIMEOUT##

export

Studio session timeouts

export

DATASTUDIO_SECURITY_SESSION_TIMEOUT_MS=##DATASTUDIO_SECURITY_SESSION_TIMEOUT_M

export

DATASTUDIO_SECURITY_ABSOLUTE_SESSION_TIMEOUT_MS=##DATASTUDIO_SECURITY_ABSOLUTE _SESSION_TIMEOUT_MS##

export

DATASTUDIO_INTERPRETER_IDLE_SESSION_TIMEOUT=##DATASTUDIO_INTERPRETER_IDLE_SESSION_TIMEOUT##

Interpreter idle session timeout and cleanup

export

DATASTUDIO_INTERPRETER_IDLE_SESSION_TIMEOUT=##DATASTUDIO_INTERPRETER_IDLE_SESSION_TIMEOUT##

export

DATASTUDIO_INTERPRETER_CLEANUP_ENABLED=##DATASTUDIO_INTERPRETER_CLEANUP_ENABLE

export

DATASTUDIO_INTERPRETER_CLEANUP_CRON="##DATASTUDIO_INTERPRETER_CLEANUP_CRON##"

export

DATASTUDIO_INTERPRETER_EXECUTION_RUN_TIMEOUT=##DATASTUDIO_INTERPRETER_EXECUTIO N_RUN_TIMEOUT##

Studio Server Thread Pool Size

export.

DATASTUDIO_SERVER_ASYNC_THREADPOOL_SIZE=##DATASTUDIO_SERVER_ASYNC_THREADPOOL_SIZE##

export

DATASTUDIO_SERVER_SCHED_THREADPOOL_SIZE=##DATASTUDIO_SERVER_SCHED_THREADPOOL_SIZE##

Hikari Pool Configurations



- export DATASTUDIO_HIKARI_MAXPOOLSIZE=##DATASTUDIO_HIKARI_MAXPOOLSIZE##
- export DATASTUDIO_HIKARI_CONN_TIMEOUT_MS=##DATASTUDIO_HIKARI_CONN_TIMEOUT_MS##
- export DATASTUDIO_HIKARI_MINIMUM_IDLE=##DATASTUDIO_HIKARI_MINIMUM_IDLE##
- export DATASTUDIO_HIKARI_MAX_LIFE_TIME=##DATASTUDIO_HIKARI_MAX_LIFE_TIME##

Tomcat Threads

export

DATASTUDIO SERVER TOMCAT THREADS MAX=##DATASTUDIO SERVER TOMCAT THREADS MAX##

Zeppelin Scheduler Threadpool Size



(i) Note

Ensure that DATASTUDIO ZPLN SCHED THREADPOOL SIZE is greater than DATASTUDIO_SERVER_TOMCAT_THREADS_MAX+DATASTUDIO_SERVER_A SYNC_THREADPOOL_SIZE

export

DATASTUDIO_ZPLN_SCHED_THREADPOOL_SIZE=##DATASTUDIO_ZPLN_SCHED_THREADPOOL_SIZE#



(i) Note

Gateway is enabled by default and gateway port is mandatory to access gateway enabled UI.

#ENDOFFILE#

Table 5-1 config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##APPLICATION_NAM E##	Title of the application. If it is not replaced, then the default value	YES	The defaults value is "Model Management and Governance."
	is : Model Management and Governance.		Note: Provide double quotes for the application name if it is long or contains spaces.
##WALLET_LOCATIO N##	The wallet is the folder that contains the sqlnet.ora, wallet.sso, and .p12 files.	YES	/scratch/users/ wallet
##TNS_ADMIN_PATH#	The folder that contains the tnsnames.ora file.	YES	/scratch/users/tns
##WALLET_ALIAS##	The wallet alias name that is configured for the MMG application config schema.	YES	MMG CONFIG



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##LOG_HOME##	A writable folder designated for storing application and MMG Studio logs.	YES	/scratch/users/ logs Note : Ensure that the log folder is created before installation.
##FTPSHARE##	This can be any writable folder accessible to the process owner.	YES	/scratch/users/ ftpshare Ensure that the ftpshare folder is created before installation. This should be same as the metadata directory mentioned above.
##LOG_TIMEZONE##	Specifies the timezone used for displaying log timestamps	YES	GMT+05:30, GMT-04:00, Asia/ Kolkata
##DATASTUDIO_URL# #	URL for MMG Studio.	YES	By default, Data Studio is assumed to run on the same server. If it is running on a different host, uncomment the line below and provide the appropriate URL. https:// <hostname ip="">:7008/ <contextpath> NOTE: The default port for MMG Studio is 7008 and should not be modified.</contextpath></hostname>
##BE_HOSTNAME##	Hostname on which the backend service (mmg-service) runs. Use the same hostname wherever applicable.	YES	HostIP or FQDN
##BE_PORT##	Port on which the backend service (mmg-service) needs to be run.	YES	7002
##SMTP_HOST##	Hostname of the SMTP server used for email notification service. This value will be populated in the AAICL_SC_COMPON ENT_DETAILS table.	YES	Defaults to localhost. Provide SMTP server hostname is external mail server is used.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##SMTP_PORT##	Port number for the SMTP server used for email notification service. This value will be populated in the AAICL_SC_COMPON ENT_DETAILS table.	YES	Defaults to 25. Update this if your SMTP service listens on a different port.
##UI_PORT##	Port on which the UI service (mmg-ui) needs to run.	YES	7001
##SCHEMA_PORT##	Port on which the Schema Creator service needs to run.	YES	7003
##CONTEXT_PATH##	Context path of the application.	YES	MMG
##STUDIO_AUTH_TY PE##	FCC_SSO – for SAMLRealm based authentication in FCC Studio MMG_AAI- AAI based authentication for MMG Studio	YES	Can be either MMG_AAI or FCC_SSO
RESTRICT_UNMAPPE D_DATASTORES_ACC ESS	Restriction of users access to Data Stores from a workspace for unmapped datasources: If the value is set as <i>True</i> , then only the current workspace attached data stores will get a connection in the notebook sessions using get_conn(). This will only work if the session-mode in the application.yml is set to NOTEBOOK_USER. Only True/False is allowed.	Yes	The default value is False
##SSL_ENABLED##	This enables HTTPS.	YES	Example: true



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##SSL_KEYSTORE##	Absolute path for the keystore file. Note: Run the following command to create a keystore: keytool -genkey -v - alias demoalias -keyalg RSA -keysize 2048 - keystore server.keystore -validity 3650 -keypass secret - storepass secret - storetype PKCS12	YES	/conf/server.keystore. Include the file name in the path. NOTE: If ##SSL_ENABLED## is set to false, then you must configure the keystore for mmgstudio, as it is SSLenabled by default. MMG application and MMG Studio can share the same SSL configuration if set up on the same server.
##SSL_KS_SECRET# #	Keystore secret The value passed in the aforementioned command for -keypass	YES	Example: secret
##SSL_KS_TYPE##	Keystore type The value passed in the aforementioned command for - storetype. Can be either JKS or PKCS12	YES	Example: PKCS12
##SSL_KS_ALIAS##	Keystore alias The value passed in the aforementioned command for -alias.	YES	Example: demoalias
##SESSION_TOKEN_ CREDENTIALS##	The password used to generate the Authorization header token to communicate with mmg-services.	YES	NOTE: If not applicable, enter NA
##FCC_API_USER##	API user for FCC Studio.	YES	NOTE: If not applicable, enter NA
##MMG_DATASOURC E_MAX_POOL_SIZE# #	Maximum connection pool size allowed for Config Datasource.	YES	The default value is 80. You can set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 16 Low concurrency (10 users)+High usage (time intensive): 80 High concurrency (50 users) + Normal usage (low cpu): 80



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##MMG_DATASOURC E_MINIMUM_IDLE##	Hikari minimum idle is a configuration property for the HikariCP database connection pool that sets the minimum number of idle connections the pool should maintain. Its default value is 10, and it ensures that there are always a certain number of connections ready for use, which helps reduce latency when new requests arrive. If the number of idle connections falls below this value, HikariCP will quickly create new ones to restore the desired minimum.	YES	
##EXT_DATASOURCE _MINIMUM_IDLE##	Hikari minimum idle is a configuration property for the HikariCP database connection pool that sets the minimum number of idle connections the pool should maintain. Its default value is 10, and it ensures that there are always a certain number of connections ready for use, which helps reduce latency when new requests arrive. If the number of idle connections falls below this value, HikariCP will quickly create new ones to restore the desired minimum.	YES	
##MMG_DATASOURC E_IDLE_TIMEOUT##	Idle timeout for config Datasource.	YES	It defaults to 30000. You can edit it if required.
##MMG_DATASOURC E_CONN_TIMEOUT##	Connection timeout for Config Datasource.	YES	It defaults to 80000. You can edit it if required.
##EXT_DATASOURCE _MAX_POOL_SIZE##	Maximum connection pool size allowed for meta/data schemas.	YES	It defaults to 10. You can edit it if required.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##EXT_DATASOURCE _IDLE_TIMEOUT##	Idle timeout for meta/ data schemas.	YES	It defaults to 30000. You can edit it if required.
##EXT_DATASOURCE _CONN_TIMEOUT##	Connection timeout for meta/data schemas.	YES	It defaults to 80000. You can edit it if required.
##MMG_HTTP_MAX_ CONN##	The maximum number of connections allowed across all routes.	YES	The default value is 200. Set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 50 Low concurrency (10 users)+High usage (time intensive): 100 High concurrency (50 users) + Normal usage (low cpu): 200
## MMG_HTTP_MAX_CO NN_PER_ROUTE##	The maximum number of HTTP connections allowed for a route.	YES	It defaults to 100. Low concurrency (10 users) + Normal usage (low cpu): 40 Low concurrency (10 users)+High usage (time intensive): 50 High concurrency (50 users) + Normal usage (low cpu): 100
##MMG_HTTP_CONN ECT_TIMEOUT##	The connection timeout for HTTP connection. A timeout value of 0 specifies an infinite timeout.	YES	The default value is 30000.
##MMG_HTTP_READ _TIMEOUT##	The socket read timeout for HTTP connection. A timeout value of 0 specifies an infinite timeout.	YES	The default value is 120000.
##APPLICATION_ID##	The id will be stored as app_id and must be the same as mentioned in the APP_ID column of MMG_PATCHES table. Currently the UI displays the MMG Version <version application="" number="" of=""> and last applied MMG version.</version>	YES	The APPLICATION_ID should be without spaces.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##APPLICATION_FAVI CON_PATH##	Icon for the application. If not specified, it will default to the icon at the following location: css/images/ favicon.ico	NO	css/images/ favicon.ico
##UI_AUTH_TYPE##	aai – if using an existing AAI instance as the identity provider. saml – for saml based authentication Idap – for Idap based authentication NOTE: This is case sensitive.	YES	Can be one of the following: aai, or saml or Idap.
##AAI_AUTH_URL##	Base URL of the AAI instance. Will be used for ##UI_AUTH_TYPE## = aai Note: If the target AAI is https, then it is necessary to import the AAI host certificate into the MMG server Java keystore.	YES	http(s):// whfxxxxx.in.oracle.com: 7110/mmg
##SAML_IDP_URL##	This is the endpoint on the IDP side where SAML requests are posted. The Service Provider (SP) needs to obtain this information from the Identity Provider (IDP).	Yes	http(s)://idcs- xxxx.com/fed/v1/idp/ss o This is used only if ##UI_AUTH_TYPE## is SAML.
##SAML_SP_ENTITY# #	Enter a globally unique name for SAML entity. It typically takes the URL of an identity provider or a service provider as a value.		http(s):// <ui_host>:<ui_por t="">/mmg This is used only if ##UI_AUTH_TYPE## is SAML.</ui_por></ui_host>
##SAML_SRV_URL##	UI Landing Page URL.	YES	http(s):// <ui_host>:<ui_por t="">/mmg/home This is used only if ##UI_AUTH_TYPE## is SAML.</ui_por></ui_host>
##SAML_LOGOUT_U RL##	Initiated SAML Single Logout URL.	Yes	http(s)://idcs- xxxx.com/sso/v1/user/ logout This is used only if ##UI_AUTH_TYPE## is SAML.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##SAML_SIGN_AUTH N_REQ##	NA	Yes	NA
##SAML_PRIVATE_KE Y_PATH##	NA	NA	/scratch/mmg8131/ config/sp- privatekey.pem
##SAML_SP_X509_CE RT_PATH##	NA	NA	/scratch/mmg8131/ config/sp-certificate.cer
##SAML_SIGN_ALGO RITHM##	NA	NA	##SAML_SIGN_ALGO RITHM##
##AUTH_SAML_REQU EST_TYPE##	This can be set to REDIRECT (default) / POST.	NA	NA
##AUTH_SAML_INCL UDE_SP_CERT##	This can be set to True/ False (default).	NA	NA
##LDAP_URL##	LDAP URL Will be used for ##UI_AUTH_TYPE## = LDAP	YES	Idap://whf00xyz:3060/
##LDAP_SEARCH_BA SE##	LDAP Search Base Will be used for ##UI_AUTH_TYPE## = LDAP	YES	"cn=Users,dc=oracle,dc =com"
##LDAP_USER_FILTE R##	LDAP User Filter Will be used for ##UI_AUTH_TYPE## = LDAP	YES	"cn={0}"
#LDAP_USER_SEARC H_FILTER##	LDAP User Search Filter Will be used for ##UI_AUTH_TYPE## = LDAP	YES	NA
##LDAP_GROUP_SEA RCH_FILTER##	LDAP Group Search Filter Will be used for ##UI_AUTH_TYPE## = LDAP	YES	NA
##LDAP_GROUP_SEA RCH_BASE##	LDAP Group Search Base Will be used for ##UI_AUTH_TYPE## = LDAP	YES	NA
##LDAP_GROUP_ME MBER#	LDAP Group Member Will be used for ##UI_AUTH_TYPE## = LDAP	YES	NA



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##SERVER_COOKIE_ DOMAIN##	The domain name.	YES	This should be the domain name of the host server. Example: .in.xyz.com Note: If the MMG application is configured with the IP address, then provide the same.
##SERVER_COOKIE_ NAME##	The name for the cookie.	YES	If not set it will default to ORA_OLDS_SESSION
##SERVER_COOKIE_ TIMEOUT##	Timeout/expiry duration in seconds.	YES	If not set, it defaults to 9999999
##SERVER_COOKIE_I S_SECURE##	Specifies if we are using cookies to add an additional security layer to prevent cross-origin requests. Can be either true or false	YES	If not set, it defaults to true.
##AAI_AUTHZ_ENABL ED##	This is disabled and is set to <i>False</i> by default. Set AAI_AUTHZ_ENABLE D to <i>True</i> to enable.	YES	
##AAI_CLIENT_ID##	Create this in the OFSAA application in the location where you had configured the AAI Auth URL. a. Login as sysadmn in OFSAA b. From System Configuration, navigate to Configure Instance Access Token c. Add a new access token.	YES	
##AAI_CLIENT_SECR ET##		YES	
##SERVER_COOKIE_ DOMAIN##		YES	
##SERVER_COOKIE_ NAME##		YES	
##SERVER_COOKIE_ TIMEOUT##		YES	
##SERVER_COOKIE_I S_SECURE##			



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##BE_AUTH_TYPE##	Auth Type on which the backend service (mmg-service) runs.	YES	It defaults to public.
##MMG_PYTHON_INT ERPRETER##	A comma separated value without whitespaces that specifies python interpreter python,fcc-ml4aml	YES	If not set, it defaults to python.
##DATACATALOG_SE RVICE_URL##	Only used when EST application is integrated with MMG.	NO	NA
##STUDIO_WALLET_ ENABLED##	Set as true when using a wallet for the MMG Studio Schema. Can be either true/ TRUE or false/FALSE	YES	true/TRUE
##LOGIN_SHOW##	(all caps or all small) Can be either true/ TRUE or false/FALSE (all caps or all small) Note: Set as true when the login screen of Studio is required. This property should be set as true if MMG application is non-SSL.	YES	It defaults to true.
##SESSION_MODE##	Can be either NOTEBOOK or NOTEBOOK_USER.	YES	If not set, it defaults to NOTEBOOK.
##STUDIO_REALM##	Can be either OFSAARealm or saml.OFSAASamlReal m	YES	OFSAARealm – the default realm for studio auth type FCC_AAI, MMG_AAIauth.saml.OFSAASaml Realm – for SAML specific studio authentication
##OFSAA_URL##	AAI login IDM Service URL. This is applicable only if ## STUDIO_AUTH_TYPE ## is " MMG_AAI ".	YES	Format: http:// <ofsa- web-host>:<port>/ <context>/rest-api For example, http:// ABC00abc:4325/LLFP/ rest-api The /rest-api is mandatory for OFSAA URL.</context></port></ofsa-



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

			I
Parameter	Description	Is Mandatory	Comments
##API_USERS##	This is the API user with which the token is generated; if not set, it defaults to MMG_API_USER.	YES	MMG_API_USER
	Note: Use the same <api_user> as given in the ##SSO_TOKEN##</api_user>		
##VALID_ROLES##	MDLUSR,MDLREV,MD LAPPR	YES	MDLBATCHUSR, DSUSRGRP,
	The comma separated values for Studio-related roles in USER-ROLE mapping.		DSREDACTGRP
##DATASOURCE_URL ##	The connection address to the database where the MMG Studio Schema is created.	YES	NA
	##WALLET_ENABLED ## is false-		
	jdbc:oracle:thin:@ <hos t>:<port>/ <service_name></service_name></port></hos 		
	When ##WALLET_ENABLED ## is true-		
	jdbc:oracle:thin:@ <ds ALIAS></ds 		
	where <ds_alias> is the wallet alias configured for the MMG Studio Schema.</ds_alias>		
##DATASOURCE_USE RNAME##	MMG Studio Schema/ User name; required only when ##WALLET_ENABLED ## is false	YES	dsschema
##DATASOURCE_PAS SWORD##	MMG Studio Schema/ User Password; required only when ##WALLET_ENABLED ## is false	YES	password
##DATASOURCE_DRI VER##	Database Driver used in connection	YES	oracle.jdbc.OracleDrive r
##JPA_DB_PLATFOR M##	Hibernate Class or SQL Dialect used in Database	YES	org.hibernate.dialect.Or acle12cDialect
##STUDIO_LOG_LEV EL##	Logging level for logs.	YES	info, warn, debug or error logs



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##PYTHON_HOME##	Home Path of Python Library. It defaults to python3 during installation. For a custom installation of python3 where the soft link is not configured, you can mention the complete path up to python3.	YES	python3
##SPARK_HOME##	Absolute path of Apache Spark Library.	NO	NA
##R_ENABLED##	This can be set to TRUE/true or FALSE/false depending on which R interpreter will be started and will be present in the interpreters list. Note:	YES	The default is False.
	If you are using an older Studio schema with an R-interpreter already present and then install with R_ENABLED set as FALSE; the R-interpreter will remain in the interpreter's menu of Studio and must be deleted from there.		
##RS_CONF_PATH##	Absolute path to Rserve.conf file for running Rserve.	YES	/scratch/users/ datastudio/conf/ Rserve.conf
##RS_KEYSTORE##	Absolute path for the Keystore file made for Rserve.conf.	YES	/scratch/users/ datastudio/conf/ rinterpreterkeystore
##RS_KS_SECRET##	Keypass for rinterpreterkeystore.	YES	Example: changeit
##TEMPLATE_CONFI G_PATH##	Configuration path of the Template.	NO	NA
##TEMPLATE_DEFAU LT_LINK##	Default link of the template.	NO	NA
##AUTH_SERVICE_U RL##	The AUTH service URL that is activated after the fccstudio.sh file runs.	NO	Example: https:// <hostname>:7041/ authservice</hostname>
##META_SERVICE_U RL##	The metaservice URL that is activated after the fccstudio.sh file runs.	NO	Example: https:// <hostname>:7045/ metaservice</hostname>



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

_			_
Parameter	Description	Is Mandatory	Comments
##ER_SERVICE_URL# #	Used for the entity resolution service.	NO	Example: https:// <hostname>:<port></port></hostname>
##BATCH_SERVICE_U RL##	Used for the batch service.	NO	Example: https:// <hostname>:<port>/ batchservice</port></hostname>
##SAML_ISSUER##	The SAML entity ID (Studio URL) configured in the IDP.	YES	https:// <hostname>.xyz.com:7 008</hostname>
##SAML_DESTINATIO N##	The SAML IDP URL that the Identity Provider provides after creating the SAML application.	YES	https://idcs- xyzgvh.com/fed/v1/idp/ sso
##SAML_ASSERTION ##	The SAML Consume URL (Studio/URL/saml/consume) that is configured in IDP.	YES	https:// <hostname>.xyz.com:7 008/saml/consume</hostname>
##SAML_ROLE_ATTRI BUTE##	The SAML client identifier provided by the SAML Administrator for the role and attributes information while creating the SAML application for MMG Studio. The attribute will contain the role required for the application.	YES	Example: group
##SAML_STUDIO_LO GOUT_URL##	The SAML client identifier provided by the SAML Administrator for the Logout URL information, while creating the SAML application for MMG Studio.	YES	https://idcs- xyzgvh.com/sso/v1/ user/logout
##SAML_COOKIE_DO MAIN##	Domain of the server.	YES	Example: in.xyz.com
Pipeline Services Spec	_		
##DATAPIPELINE_SE RVICE_PORT1##	The port where the pipeline service resides.	YES	By default, it is set as 18005.
##DATAPIPELINE_SE RVICE_PORT2##	The port where the data pipeline service resides.	YES	By default, it is set as 18006.
##DATA PIPELINE_METADATA _ARCHIVE_PATH##	The dump path for the pipeline service.	YES	/OFS_MMG/mmg- pipeline/pipeline/ pipeline-service- x.x.x.x.x.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##DATA PIPELINE_METADATA _IMPORT_SERVICE_P ORT##	Meta data import service port.	YES	By default, it is set as 18007.
##DATAPIPELINE_ER XMLPATH##	The XML path in which the schema details are stored.	NO	NA
##DATAPIPELINE_GAT EWAY_SERVICE_POR T##	'' '	YES	NA
##PIPELINE_UI_SERV ICE_PORT##	Pipeline UI service port.	YES	NA
##DATA_PIPELINE_UI _SERVICE_PORT##	Data pipeline UI service port	YES	NA
##MATCHRULE_BASE _URL##	The host and port where the match rule service resides.	NO	http(s):// abc.in.xyz.com:7051
##LOADGRAPH_BAS E_URL##	The host and port where the load graph service resides.	YES	http(s)://abc.in.xyz.com: 7059/graph-service
##MATCHSRVC_UI_U RL##	Matching Service UI resource path.	NO	NA



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##GRAPH_INDEX_BA	Indicates the Graph	NO	NA
SE_URL##	Index resource path.		
##LOADINDEX_UI_UR	Indicates the Graph	NO	NA
L##	Load Index UI resource		
	path.		



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Darameter	Description	le Mandatory	Comments
Parameter	Description	Is Mandatory	Comments
##MATCHING_MECHA NISM##	Indicates the matching mechanism for Entity Resolution and Graph Enter the value that is used for matching mechanism for Entity Resolution and Graph. The options are OS, and OT. Where OS refers to OpenSearch 2.7.0, and OT refers to Candidate Selection with Oracle Text matching. By default, the value of the parameter is OT. For MATCHING_MECHANI SM = OT, no additional installation or configuration is required as it is part of the Oracle Database.	NO	NA
	NOTE: This is applicable for Compliance Studio.		
##CANDIDATE_SELEC TION_SERVICE_URL# #	Enable this service for Entity Resolution and Graph use cases when MATCHING_MECHANI SM is set to OT. NOTE: This is applicable for Compliance Studio.	NO	NA
##LOAD_TO_OS_URL ##	Provide the URL for Entity Resolution and Graph use cases when MATCHING_MECHANI SM is set to OT. NOTE: This is applicable for Compliance Studio.	NO	NA
##AAI_COOKIE_DOM AIN##	The domain of the server.	YES	Example: in.xyz.com
##MMG_KEYS_LOC##	Indicates public and private key location.	YES	Example:< MMG Installation Path> / OFS_MMG/conf
##GRAPH_INSTALLAT ION_PATH##	The installation path of the Graph.	YES	<pre><mmg installation="" path="">/ OFS_MMG/ mmg-load-to-graph/ graph-service</mmg></pre>
##GRAPH_KEYSTOR E_PASSWORD##	Graph Keystore Password.	YES	Password



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##GRAPH_SERVICE_ PORT##	Graph Service Port	YES	By default, it is set as 7059. You should not modify the Graph Service Port if graphs are already created and executed.
##GRAPH_HIKARI_CP _MIN_IDLE##	Defines the minimum number of idle connections that HikariCP maintains in the pool.	YES	
##GRAPH_HIKARI_CP _MAX_POOL_SIZE##	Specifies the maximum number of total connections (active + idle) allowed in the pool.	YES	
##GRAPH_HIKARI_CP _IDLE_TIMEOUT##	Sets the maximum time a connection can remain idle before being removed from the pool.	YES	
##GRAPH_HIKARI_CP _MAX_LIFETIME##	Determines the maximum lifetime of a connection before it is retired and replaced.	YES	
##GRAPH_HIKARI_CP _TIMEOUT##	Defines how long HikariCP waits for a connection from the pool before timing out.	YES	
##MMG_DB_SERVER _NAME##	Name of the MMG Database Server.	YES	NA
##MMG_DB_PORT##	The port of the MMG database server.	YES	NA
##MMG_DB_SERVICE _NAME##	Name of the MMG Database Service.	YES	NA
##PGX_SERVER_URL S##	Indicates the pgx server resource path. Skip this if not installing pgx.	YES	http(s):// <hostname>.xyz.com:< pgx port>/<pgx context="" name=""></pgx></hostname>
##MAX_TOTAL_SHAR ED_DATA_MEMORY_S IZE##	Maximum total shared data memory size.	YES	Edit if required; default value is 20 GB.
##MAX_TOTAL_PRIVA TE_DATA_MEMORY_S IZE##	Maximum total private data memory size.	YES	Edit if required; default value is 8 GB.
##MAX_PER_SESSIO N_DATA_MEMORY_SI ZE##	Maximum per session data memory size.	YES	Edit if required; default value is 700 MB.
##MAX_DATA_MEMO RY_SIZE_DSUSRGRP ##	Maximum data memory size allowed for DSUSRGRP.	YES	Edit if required; default value is 10 GB.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##MAX_DATA_MEMO RY_SIZE_DSBATCH##	Maximum data memory size allowed for DSBATCH.	YES	Edit if required; default value is 10 GB.
##MAX_DATA_MEMO RY_SIZE_DSINTER##	Maximum data memory size allowed for DSINTER.	YES	Edit if required; default value is 5 GB.
##MAX_DATA_MEMO RY_SIZE_DSAPPROV ER##	Maximum data memory size allowed for DSAPPROVER.	YES	Edit if required; default value is 5 GB.
##MAX_DATA_MEMO RY_SIZE_DSUSER##	Maximum data memory size allowed for DSUSER.	YES	Edit if required; default value is 5 GB.
##GRAPH_SCHEMA_ WALLET_ALIAS##	Wallet alias created for the Graph Schema.	YES	NA
##GRAPH_SCHEMA_ DB_SCHEMA_NAME# #	Name of the Graph schema.	YES	NA
##EST_ENABLED##	Only used when EST application is integrated with MMG.	YES	The default value is FALSE.
##EST_UI_URL##	The URL of EST application. This is set based on ##EST_ENABLED## property.	NO	NA
##DATASTUDIO_SERV ER_PORT##	The port of the Data Studio server.	NO	The default value is 7008.
##DATASTUDIO_MAR KDOWN_INTERPRET ER_PORT##	The port of the Data Studio Markdown Interpreter.	NO	The default values are 7009, 7029.
##DATASTUDIO_PYTH ON_INTERPRETER_P ORT##	The port of the Data Studio Python Interpreter.	NO	The default values are 7012, 7032, 6012.
##DATASTUDIO_JDBC _INTERPRETER_POR T##	The port of the Data Studio JDBC Interpreter.	NO	The default values are 7011, 7031.
##DATASTUDIO_PYTH ON_INTERPRETER_R EST_SERVER_PORT# #	•	NO	The default value is 6012
##DATASTUDIO_PGX_ PYTHON_INTERPRET ER_REST_SERVER_P ORT##	The port of the Data Studio PGX Python Interpreter Rest server.	NO	The default value is 6022
##DATASTUDIO_THRI FT_EVENT_HANDLER _PORT##	The port of the Data Studio Thrift Event handler.	NO	The default value is 8432
##DATASTUDIO_PGX_ INTERPRETER_PORT ##	The port of the Data Studio PGX Interpreter.	NO	The default value is 7022.



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##MMG_COHERENCE _CLUSTER_PORT##	This is the port where the Coherence Cluster for the Notification service runs. By default, this value is 7574.	YES	NA
MMG Gateway Configu	ıration	_	_
##MMG_GATEWAY_E NABLED##	Enables or disables the MMG Gateway. By default, this is set to 'self'. i Note Gateway is enabled by default and gateway port is mandatory to access gateway enabled UI.	YES	NA
##MMG_GATEWAY_P ORT##	Specifies the port on which the MMG Gateway will run.	YES	NA
##MMG_CSP_FRAME _ANCESTORS##	Configure this attribute to all or the AAI origin when MMG has to be embedded from AAI. By default, MMG pages cannot be embedded if Gateway is enabled. This is to prevent CLickjacking vulnerability.	YES	NA



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##DATASTUDIO_CSP_ FRAME_ANCESTORS ##	If the Gateway is enabled, this property can be set to control the pages where Data Studio can be embedded: Set to '*' to allow embedding from any origin (less secure). Set to a commaseparated list of origins to allow embedding from those specified origins and from the same origin. By default, this is set to MMG Gateway URL.	YES	NA
##DATASTUDIO_JOBS _INCOMPLETE_TIME OUT##	You can configure how often a cleanup job will be run to mark unfinished jobs as incomplete by setting studioserver.jobs.incomplete. scheduled-cleanupdelay value. A job can be in incomplete state if the server restarts in the middle of a Job execution or if it takes longer the specified timeout (studioserver.jobs.incomplete.t imeout).		NA
##DATASTUDIO_JOBS _INCOMPLETE_SCHE DULED_CLEANUP_D ELAY##	You can configure how often a cleanup job will be run to mark unfinished jobs as incomplete by setting studio-server.jobs.incomplete. scheduled-cleanup-delay value. A job can be in incomplete state if the server restarts in the middle of a Job execution or if it takes longer the specified timeout (studio-server.jobs.incomplete.t imeout).		NA



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##DATASTUDIO_SEC URITY_SESSION_TIM EOUT_MS##	You can configure user login sessions to be renewed by activity. The number specifies the amount of milliseconds a user session will be renewed after activity. When there is no useractivity within the specified relative timeout, the session expires and is no longer valid. Here activity means edit of paragraph content, addition or deletion of paragraph, execution of paragraph, etc.	Yes	43200000
##DATASTUDIO_SEC URITY_ABSOLUTE_S ESSION_TIMEOUT_M S##	You can configure user login sessions to be logged out automatically after a set amount of time. The number specifies the amount of milliseconds to pass after a session is created. When the timeout expires, the session is no longer valid Here login means when the user logs in to the Compliance Studio. When a user session ends, all associate executions will also be killed.	Yes	It is recommended to set this. A value of 0 means that the sessions will never time out. Users can set this to 86400000 to set a timeout of 24 hours or it can be set to 234000000 to set a timeout of 65 hours, in case the jobs are needed to be run the entire weekend.
##DATASTUDIO_INTE RPRETER_IDLE_SES SION_TIMEOUT##	The amount of time an interpreter session can stay idle before getting invalidated by the application to free resources. An interpreter session is considered idle while it isn't running any tasks. Example values are: 1h (one hour), or use ISO 8601 duration format for more complex values like PT2H30M (2h30m).	NA	



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Darameter	Description	Is Mandatory	Comments
Parameter ##DATASTUDIO INTE	•	Is Mandatory Yes	Comments NA
##DATASTUDIO_INTE RPRETER_EXECUTIO N_RUN_TIMEOUT##	You can configure to automatically cancel a paragraph execution after a set amount of time. When the timeout expires, Data Studio will automatically interrupt the execution of the paragraph.	res	INA
##DATASTUDIO_HIKA RI_CONN_TIMEOUT_ MS##	This setting controls the maximum time (in milliseconds) that the HikariCP connection pool will wait for an available database connection. If no connection is available within that time frame, it will throw an error.	Yes	NA
##DATASTUDIO_HIKA RI_MINIMUM_IDLE##	This determines the minimum number of idle connections HikariCP will maintain in the connection pool at all times. Even if there are no active requests for database connections, HikariCP will ensure that at least this number of idle connections remain open, ready to be used.	NA	
##DATASTUDIO_ZPLN _SCHED_THREADPO OL_SIZE##	This controls the number of threads in the ZPLN scheduler's thread pool, determining how many concurrent tasks the system can handle	Yes	NA
##DATASTUDIO_SERV ER_TOMCAT_THREA DS_MAX##	This controls the maximum number of concurrent threads that the Tomcat server can use to process incoming requests.	Yes	Set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 100 Low concurrency (10 users)+High usage (time intensive): 200 High concurrency (50 users) + Normal usage (low cpu): 200 High concurrency (50 users) + High usage (time intensive): 500



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
	NA	YES	NA
##SAML_SIGN_AUTH N_REQ##	INA	165	NA .
##SAML_PRIVATE_KE Y_PATH##	NA	NA	/scratch/mmg8133/ config/sp- privatekey.pem
##SAML_SP_X509_CE RT_PATH##	NA	NA	/scratch/mmg8133/ config/sp-certificate.cer
##SAML_SIGN_ALGO RITHM##	NA	NA	##SAML_SIGN_ALGO RITHM##
##AUTH_SAML_REQU EST_TYPE##	NA	NA	##AUTH_SAML_REQU EST_TYPE##
##AUTH_SAML_INCL UDE_SP_CERT##	NA	NA	##AUTH_SAML_INCL UDE_SP_CERT##
##GIT_ENV_ID##	This variable represents the environment in which your GIT operations or deployments are being performed. Common environment names include dev (development), QA (quality assurance), SIT (system integration testing), PROD (production), and so on. It helps you to understand with which branch you need to work.	NA	##GIT_ENV_ID## A new group GIT ADMIN is created which is mapped to GIT_ADV role which is mapped to below functions. Kindly map the above group to the user to access the required GIT functionality: GITVIEW GITPUSH GITPULL
##GIT_USERNAME##	This is your GitHub username.	NA	##GIT_USERNAME##
##GIT_PAT_SECRET# #	This variable stores your GitHub Personal Access Token (PAT). A PAT is a secret token used for authenticating API requests or GIT operations over HTTPS instead of a password.	NA	##GIT_PAT_SECRET# #
##GIT_PROXY_URL##	If your environment requires you to access external resources like GitHub through a proxy server, this variable defines the proxy server's hostname or IP Address.	NA	##GIT_PROXY_URL##



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##GIT_PROXY_PORT ##	This defines the port number on which the proxy server listens. It works together with GIT_PROXY_URL to route your Git traffic through the proxy.	NA	##GIT_PROXY_PORT ##
##DATASTUDIO_INTE RPRETER_CLEANUP _ENABLED##	Idle sessions will never get invalidated if cleanup is not enabled by the additional settings described below, or when the idlesession-timeout value is set to 0.	Yes	False
##DATASTUDIO_INTE RPRETER_CLEANUP _CRON##	Refers to a scheduled cleanup task for an interpreter (such as the one that executes reports or queries) within a system like Data Studio, which helps optimize system performance by clearing out temporary or unused data at regular intervals.	Yes	
##DATASTUDIO_SERV ER_ASYNC_THREAD POOL_SIZE##	This is a configuration parameter for controlling the number of concurrent threads allocated to handle asynchronous tasks (like data processing, report rendering, etc.) on a Data Studio (or similar) server. It helps manage system performance, especially when many users are interacting with the system or large amounts of data need to be processed.	Yes	The default value is 128. Set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 32 Low concurrency (10 users)+High usage (time intensive): 128 High concurrency (50 users) + Normal usage (low cpu): 128



Table 5-1 (Cont.) config.sh File Parameters and Descriptions

Parameter	Description	Is Mandatory	Comments
##DATASTUDIO_SERV ER_SCHED_THREAD POOL_SIZE##	This is a setting that controls the number of concurrent threads available for executing scheduled tasks on the Data Studio server. By adjusting this, you can manage how many tasks, such as data refreshes or scheduled reports, can be processed in parallel.	Yes	The default value is 128. Set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 32 Low concurrency (10 users)+High usage (time intensive): 128 High concurrency (50 users) + Normal usage (low cpu): 128
##DATASTUDIO_HIKA RI_MAXPOOLSIZE##	This setting controls the maximum number of database connections that can be maintained by the HikariCP connection pool in the Data Studio server. When the system needs to interact with a database (for data fetching, report generation, etc.), it requests a connection from the pool.	Yes	The default value is 650. Set this to any one of the following values: Low concurrency (10 users) + Normal usage (low cpu): 250 Low concurrency (10 users)+High usage (time intensive): 350 High concurrency (50 users) + Normal usage (low cpu): 650

Note

If changes are done directly on the UI and then restarted, then the overwrite will not be triggered and the configuration will not change. It is only dependent on the file system JSON.



(i) Note

- In case of ##OFSAA URL## and ##MMG SVC URL##, do not add any ending '/' in the URLs
- If the pool size, connection timeout and idle timeout are not configured, then it will proceed with the default Hikari Configurations.
- The default session timeout is 3600 seconds (60 mins). You can configure the timeout by using the server.servlet.session.timeout property.
- If the AUTH type specified is AAI, make sure the AAI System has appropriate user groups mapped for the users. WKSPADMIN, IDNTYADMN, IDNTYAUTH need to be present for a successful subsequent logins.
- The name for the MMG Studio cookie is ORA OLDS SESSION.
- If the ##SSL ENABLED## is set to false, the keystore configuration must be done for mmg-studio as it is SSL enabled by default. MMG application and MMG Studio can use the same SSL configuration if configured in the same server.
- The wallet is the same for all the MMG services including MMG Studio. Hence, if you want to use MMG Studio with wallet configurations, then configure it in the same wallet.
- If MMG Studio is remotely configured, then the MMG Application Configuration Schema wallet alias and tnsnames.ora file entries need to be added to the MMG Studio configured wallet and tnsnames.ora file.
- If the MMG application is Non SSL, set the below property to "false" in the application.yml file inside MMG Studio and restart the services. security:

cookies:

secure: false

(i) Note

The flag *IS SELF USR GRP AUTH ALLOWED* in the NEXTGENEMF CONFIG table is used to check if the Self Groups Approvers/Reviewers are allowed or not. If it is 'Y', then the Requested User as well Approvers and Reviewers can belong to the same user group. Otherwise, 'Self User Group Authorization is not allowed' is returned.

Support of Alphanumeric Name in SAML Entity ID for SAML Configuration

MMG Application: mmg ui/application.properties file contains SAML entity saml.auth.sp.entity placeholder that contains values like https://<FQDN of compliance server>:7001/cs

However, the requirement is to have this parameter value as amlcs8126 i.e. support of SAML Entity ID as alphanumeric name.

The expected value from SAML team:

saml.auth.sp.entity="amlcs8126":



Entity ID should not contain below special character as per ForgeRock MF Authentication (Policy sets in the UI: ForgeRock Identity Cloud docs).

The regex allowed is: $^[A-Za-z0-9, .\/\]=\$$, and it should be passed in a string.

Import Server Certificate to Java Keystore

You must import the server certificate (.cer) file to the Java keystore.

To import the server certificate, perform the following steps:

1. Create a .cer file from the server.keystore.

```
keytool -export -alias <alias>-file <filename>.cer -keystore
<path_to_Keystore>/server.keystore -storepass secret
```

Example:

keytool -export -alias demo alias -file server.cer -keystore OFS MMG/config/ server.keystore -storepass secret

2. Import .cer file generated from the above step to java keystore.

```
keytool -import -file "<path_to_Keystore>/<filename>.cer" -alias <alias>-
keystore "<java_home>/lib/security/cacerts" -storepass "changeit"
```



The above step should be performed by the Root user.

Run the MMG Installer

To run the MMG Installer, follow these steps:

1. Navigate to following path:

Go to <MMG_INSTALLATION_PATH>/bin directory.

2. Run the following command:

./install.sh



(i) Note

When ./install.sh command is triggered, pre-installation utility validates install configurations such as availability of ports, ftpshare/log folders, database connections, and so on.

This step will install the configurations and has to be executed only once per deployment. This will also bring up the Schema Creator Service in nohup mode.

A message similar to following means a successful startup:

<MMG_INSTALLATION_PATH>/OFS_MMG/bin>./install.sh

PIPELINE HOME: <MMG_INSTALLATION_PATH>/OFS_MMG/mmg-pipeline/pipeline

<MMG_INSTALLATION_PATH>/OFS_MMG/mmg-pipeline/pipeline

PIPELINE HOME: <MMG INSTALLATION PATH>/OFS MMG/mmg-pipeline/pipeline



Installing Pipeline Data Model. Please Wait ...

Pipeline Data Model installation finished.

Starting Gateway ...

Starting Pipeline UI Service ...

Starting Pipeline Service ...

Starting Data Pipeline UI Service ...

Starting Data pipeline services ...

Inserting DataMeta Data ...

Stopping Graph-Service service...

Graph-Service stopped.

Schema Creator executed successfully for config schema

Schema Creator for config executed successfully.

If Graph Schema is configured, the below message is displayed.

Now triggering for graph-schema

./../mmg-schema-creator/bin/startup.sh: line 70: 126438 Killed nohup java -jar -Doracle.net.tns admin=/scratch/ofsaadb -Doracle.net.wallet location=/scratch/ofsaadb/ wallet -Dspring.config.location=../conf/ -Dspring.datasource.url=idbc:oracle:thin:@conf_als -Dspring.liquibase.change-log=file:../scripts/changelog-master.xml \$JAVA OPTS ../lib/ mmg-schema-creator.war > nohup.out 2>&1

Schema Creator executed successfully for graph schema

nohup: ignoring input and redirecting stderr to stdout

Stopping Graph-Service service...

Graph-Service stopped.

nohup: ignoring input and redirecting stderr to stdout

You can check mmg-schema-creator/bin/nohup.out to check if the service comes up properly.

Started BuildSchemaCreatorApplication in 20.317 seconds (JVM running for 21.26)



🛕 Warning

If you notice any errors, do not proceed further. Contact My Oracle Support (MOS) and provide the applicable error code and log files.

Execute shutdown.sh and trigger startup.sh for the services to come up. For more details, refer to the below sections.



(i) Note

The MMG Application is installed with or without OFSAA, depending on the configuration provided in the config.sh file.



Modify the Logging Level

The log rotation policy in the default OFS MMG setup is set to create a new log file whenever the size crosses 5MB and also to retain the last 5 log files. This feature is customizable. Additionally, the logging level can be modified to mmg-service, mmg-ui, mmg-studio etc.

The log config file for each service is present at the following locations. A backup can be taken and updates can be performed on the same:

- <MMG_INSTALLATION_PATH>/mmg-service/conf/log4j2.xml
- <MMG_INSTALLATION_PATH>/mmg-ui/conf/log4j2.xml
- <MMG_INSTALLATION_PATH>/mmg-studio/conf/log*.xml
- <MMG_INSTALLATION_PATH>/mmg-gateway/conf/log4j2.xml

Sample 1 (Size based rolling)

```
<Appenders>
<RollingFile name="RollingFile">
  <FileName>##LOG HOME##/services/mmg-service.log</FileName>
  <FilePattern>##LOG HOME##/services/mmg-service.%i.log.zip/FilePattern>
  <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG TIMEZONE##}
[%18.18t] %-5p [%c{1}] - %m%n"/>
  <Policies>
   <!-- New log once the current size crosses 10MB -->
  <SizeBasedTriggeringPolicy size="10MB"/>
  </Policies>
  <!-- keep last 20 log files -->
  <DefaultRolloverStrategy max="20"/>
 </RollingFile>
 <RollingFile name="HealthCheckFile">
  <FileName>##LOG HOME##/services/mmq-health.log/FileName>
  <FilePattern>##LOG HOME##/services/mmq-health.%i.log.zip/FilePattern>
  <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG_TIMEZONE##}
[%18.18t] %-5p - %m%n"/>
  <Policies>
   <!-- New log once the current size crosses 12MB -->
  <SizeBasedTriggeringPolicy size="12MB"/>
  </Policies>
  <!-- keep last 15 log files -->
  <DefaultRolloverStrategy max="15"/>
 </RollingFile>
 <Console name="Console" target="SYSTEM OUT">
  <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG TIMEZONE##}
[\$18.18t] \$-5p [\$c{1}] - \$m\$n"/>
 </Console>
</Appenders>
```

Sample 2 (Day based logs rolling at midnight)



```
dd \ . loq . qz " >
<PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG TIMEZONE##}</pre>
[\$18.18t] \$-5p [\$c\{1\}] - \$m\$n"/>
<Policies>
 <!-- rollover at midnight -->
 <TimeBasedTriggeringPolicy interval="1" modulate="true"/>
</Policies>
<!-- keep last 10 daily logs -->
<DefaultRolloverStrategy max="10"/>
</RollingFile>
 <RollingFile name="HealthCheckFile"
              fileName="##LOG HOME##/services/mmg-health.log"
              filePattern="##LOG HOME##/services/mmq-health-%d{yyyy-MM-
dd}.log.gz">
  <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG_TIMEZONE##}
[%18.18t] %-5p - %m%n"/>
  <Policies>
  <!-- rollover at midnight -->
  <TimeBasedTriggeringPolicy interval="1" modulate="true"/>
 <!-- keep last 15 daily logs -->
 <DefaultRolloverStrategy max="15"/>
</RollingFile>
<Console name="Console" target="SYSTEM_OUT">
   <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG_TIMEZONE##}</pre>
[%18.18t] %-5p [%c{1}] - %m%n"/>
 </Console>
</Appenders>
```

Sample 3 (Hourly log files)

```
<Appenders>
  <RollingFile name="RollingFile"</pre>
               fileName="##LOG_HOME##/services/mmg-service.log"
               filePattern="##LOG_HOME##/services/mmg-service-%d{yyyy-MM-dd-
HH \ . loq . qz " >
    <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG_TIMEZONE##}
[\$18.18t] \$-5p [\$c\{1\}] - \$m\$n"/>
    <Policies>
      <!-- rollover every hour on the clock -->
      <TimeBasedTriggeringPolicy interval="1" modulate="true"/>
    </Policies>
    <!-- keep last 10 hourly logs -->
    <DefaultRolloverStrategy max="10"/>
  </RollingFile>
  <RollingFile name="HealthCheckFile"
               fileName="##LOG_HOME##/services/mmg-health.log"
               filePattern="##LOG_HOME##/services/mmg-health-%d{yyyy-MM-dd-
HH \ . loq . qz " >
    <PatternLayout pattern="%d{dd-MM-yyyy HH:mm:ss.SSS}{##LOG_TIMEZONE##}
[%18.18t] %-5p - %m%n"/>
    <Policies>
      <!-- rollover every hour on the clock -->
      <TimeBasedTriggeringPolicy interval="1" modulate="true"/>
    </Policies>
```



Starting MMG Services

To start the MMG service, run the following command:

Navigate to <mmg_INSTALLATION_PATH>/bin directory. ./startup.sh

A message similar to following means a successful startup:

Starting MMG UI...

MMG UI started successfully.

Starting MMG Service...

MMG Service started successfully.

Starting Data Studio...

Data Studio started successfully.

Starting Gateway ...

Starting Pipeline UI Service ...

Starting Pipeline Service ...

Starting Data Pipeline UI Service ...

Starting Data pipeline services ...

You may check <MMG_INSTALLATION_PATH>/mmg-ui/bin/nohup.out to check if the UI service comes up properly.

A message similar to following means a successful startup:

Started BuildUIServiceApplication in 27.981 seconds (JVM running for 29.365)

You can check <MMG_INSTALLATION_PATH>/mmg-service/bin/nohup.out to check if the backend service comes up properly.

A message similar to following means a successful startup:

Started BuildServiceBuildApplication in 20.317 seconds (JVM running for 21.26)

You can check <MMG_INSTALLATION_PATH>/mmg-studio/bin/nohup.out to check if the backend service comes up properly.

A message similar to following means a successful startup:

05:06:02.155 Thread-9] INFO oracle.datastudio.starter.App - Data Studio Server is ready to use

This will start the successful installation of application.





Warning

If you notice any errors, do not proceed further. Contact My Oracle Support (MOS) and provide the applicable error code and log files.

(i) Note

Unset the https/http proxy details before starting the services.

Add the relevant entries in no proxy with mmg hosted server details.

Stopping MMG Services

To stop the MMG services, run the following command: ./shutdown.sh

A message similar to following means a successful shutdown:

Stopping Graph-Service service...

Graph-Service stopped.

MMG UI shutdown is complete.

MMG Service shutdown is complete.

MMG Schema Creator shutdown is complete.

Data Studio shutdown is complete.

Data Pipeline Service shutdown is complete.

Generate GRAPH-KEYSTORE.P12

Graph services should be up and running.

To generate GRAPH-KEYSTORE.P12 file, perform the below steps:



Note

The Keystore generation fails if graph service is down.

- Execute graph-keystore-generator.sh using PUTTY.
- Enter the values as below when prompted.

Enter Wallet Alias: <GRAPH_SCHEMA_WALLET_ALIAS> as given in the config.sh file.

Enter Password: <GRAPH_SCHEMA_DB_SCHEMA> password

Enter Keystore alias: <GRAPH SCHEMA DB SCHEMA NAME> as given in the config.sh file.

Check the below location for the graph-keystore.p12

<mmg installation path>/OFS MMG/mmg-load-to-graph/graph-service/conf/



Installing Conda

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.



Note

The supported version is 4.14.0.

To install the Conda, perform the following:

- Download the miniconda.
- Copy it to your server where the Conda needs to be installed.
- Grant execute permission to the Conda folder.
- Execute the following command: \$./Miniconda3-latest-Linux-x86_64.sh
- Update the PATH variable with miniconda installation path:

<install_path>/miniconda3/bin



Note

In the current release, the Conda feature is not supported in Solaris Operating

For more details on the Roles and privileges, see MMG User Guide.

Installing Python Library

This section provides detailed steps to install the Python Library.

Prerequisites

Python 3.8.x and above



Note

Ensure the libraries, bzip2-devel, sqlite-devel, ncurses-devel, and xz-devel, libffidevel are installed before you install the Python package.

For Example:



(i) Note

Install the below libraries as a root user.

- bzip2-devel: Execute the command yum install bzip2-devel
- sqlite-devel: Install as a root user using the command yum install sqlitedevel
- ncurses-devel: Install as a root user using the command yum install ncurses-devel
- xz-devel: Install as a root user using the command yum install xz-devel
- libffi-devel: Install as a root user using the command yum install libffidevel

(i) Note

In order to download the latest python packages after MMG upgrade, the pip cache in the server user home directory needs to be cleared.

Path: \$HOME/.cache/pip

(i) Note

Documentation of Mandatory Parameter to be passed as part of Python environment install.sh Installation

Now, parameters is supported as below for python-env-install.sh file:

The user needs to be passed in case the packages has to be installed at a user level. The above is not required in case of CONDA environment. In this case make sure the Python3 is pointed to proper Conda environments.

Procedure

- 1. Set system python3 to the one that is to be used. Navigate to bin folder.
- 2. To install the mmg library with dependencies from conf/requirements.txt, execute the following command:

./python-env-install.sh

The following parameter is supported for python-env-install.sh file:

--user need to be passed in case the packages has to be installed at user level.

This is not required in case of Conda environment but python3 must be configured to the appropriate Conda environments.





(i) Note

Expose Python variables/functions to derive File Path when saving files directly from Notebook script (without using Save Widget).

To install the mmg library with flexible dependencies or using already installed dependent packages, execute the following command:

```
./python-env-install.sh -S
```

OR

```
./python-env-install.sh --skip
```

This will skip the installation of dependency based on the version mentioned in the conf/ requirements.txt. The installation will be with whatever version available in the pypi server.

To install the Apache Flink packages, execute the following command:

```
./python-env-install.sh --include-flink
```



Note

Ignore the below error message during Apache Flink package installation. ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behavior is the source of the following dependency conflicts. modin 0.19.0 requires pandas==1.5.3, but you have pandas 1.3.5 which is incompatible. Successfully installed numpy-1.21.4 pandas-1.3.5 python-dateutil-2.8.0 Installing with dependencies ERROR: pip's dependency resolver does not currently take into account packages that are installed. This behavior is the source of the following dependency conflicts. pemja 0.2.6 requires numpy==1.21.4, but you have numpy 1.24.2 which is incompatible. apache-flink 1.16.1 requires numpy<1.22.0,>=1.21.4; python full version >= "3.7", but you have numpy 1.24.2 which is incompatible. apache-flink 1.16.1 requires pandas<1.4.0,>=1.3.0; python_full_version "3.7", but you have pandas 1.5.3 which is incompatible. apache-flink 1.16.1 requires python-dateutil==2.8.0, but you have python-dateutil 2.8.2 which is incompatible. apache-beam 2.38.0 requires numpy<1.23.0,>=1.14.3, but you have numpy 1.24.2 which is incompatible.

Setting up the Environment for Hive Data Sourcing

This section is applicable if you want to use Hive Data Source.

In the MMG Home directory, a lib folder is available for the Hive specific jars and a conf folder is available for the Kerberos Configuration and Keytab files.

Hive Source Connection Requirements:

MMG_HOME/conf: kbank.keytab and krb5.conf files

MMG HOME/lib: hive-jdbc-uber-2.6.3.0-235.jar

MMG-Studio/conf: kbank.keytab, krb5.conf and hive-jdbc-driver.jar





(i) Note

The Datastudio placement of jars are for creating a connection from python library and the other is from java, for data sourcing.

Configure the Hive Jars and Configuration Files

For Hadoop version 3.1.1 and hive version 3.1.2, below is the list of jar files that needs to be copied into the OFS MMG/lib location:

zookeeper-3.4.9.jar

woodstox-core-5.0.3.jar

stax2-api-3.1.4.jar

slf4j-log4j12-1.7.25.jar

slf4j-api-1.7.25.jar

re2j-1.1.jar

log4j-1.2.17.jar

libthrift-0.9.3.jar

libfb303-0.9.3.jar

httpcore-4.4.4.jar

httpclient-4.5.2.jar

htrace-core4-4.1.0-incubating.jar

hive-service-3.1.2.jar

hive-metastore-3.1.2.jar

hive-jdbc-3.1.2.jar

hive-exec-3.1.2.jar

hadoop-hdfs-client-3.1.1.jar

hadoop-common-3.1.1.jar

hadoop-auth-3.1.1.jar

curator-client-2.12.0.jar

commons-logging-1.0.4.jar

commons-io-2.4.jar

commons-configuration2-2.1.1.jar

commons-collections-3.2.2.jar

commons-cli-1.2.jar

The MMG service requires a restart after copying the Hive jars and configuration files. For more information, see the MMG User Guide.



Remote MMG Studio Configuration

For Solaris Operating System, the MMG Studio has to be configured in Linux machine remotely. The MMG Studio URL must be the same as that of the remote studio during MMG Application Installation.

In the OFS_MMG/bin/config.sh, update the following properties with the remote server where the MMG Studio will be running:

Copy the mmg-studio folder to the remote machine where you want to configure the same.

Navigate to mmg-studio/bin and update the config.sh file with respect to studio server values. For more details, see the Configure the config.sh file section.

```
export DATASTUDIO_URL=##DATASTUDIO_URL##
export SSL_KEYSTORE=##SSL_KEYSTORE##
export SSL_KS_SECRET=##SSL_KS_SECRET##
export SSL_KS_TYPE=##SSL_KS_TYPE##
export SSL_KS_ALIAS=##SSL_KS_ALIAS##
```

Note

The keystore must be generated for the remote machine and the path must be present in the remote server.

```
export DS_TNS_ADMIN_PATH=##DS_TNS_ADMIN_PATH##
export DS_WALLET_LOCATION=##DS_WALLET_LOCATION##
```

TNS admin and wallet must be configured in the remote server and the wallet must contain the mmg config schema wallet configurations.

```
export MMG_TNS_ADMIN_PATH=##MMG_TNS_ADMIN_PATH##
export MMG_LIB_WALLET_ALIAS=##MMG_LIB_WALLET_ALIAS##
```



Note

The Self signed certificate needs to be generated and imported to the Java keystore. In case self-signed certificate is being used, perform the below step:

 Delete the already imported certificate from the Java keystore using the below command keytool:

-delete -noprompt -alias <alias_name> -keystore "<java_home>/lib/ security/cacerts" -storepass "changeit"

2. Run the following command to re-create a Java keystore:

keytool -genkey -v -alias <alias_name> -keyalg RSA -keysize 2048 - keystore server.keystore -validity 3650 -keypass secret -storepass secret -storetype PKCS12

3. Create a .cer file from the server.keystore. keytool -export -alias <alias>-file <filename>.cer -keystore <path_to_Keystore>/ server.keystore -storepass secret
Example: keytool -export -alias demo_alias -file server.cer -keystore OFS MMG/config/server.keystore -storepass secret

4. Import .cer file generated from the above step to java keystore. keytool -import -file "<path_to_Keystore>/<filename>.cer" -alias <alias>-keystore "<java_home>/lib/security/cacerts" -storepass "changeit"

(i) Note

The above step should be performed by the Root user.

(i) Note

The token is automated. Once the token is generated, ignore '-e' character present in the token.out file.

PGX Installation

Note

PGX Installation is recommended to be installed in a different server other than the MMG Installation Server.

To install the PGX, follow these steps:

- 1. Copy the mmg-pgx.zip file from MMG Server and copy it to the target server where PGX has to be installed remotely to MMG.
- 2. Unzip the mmg-pgx.zip file.

For Example: unzip -a mmg-pgx.zip.



The below files will be displayed:

- bin
- conf
- pgx-25.4.0
- Give 0755 permission to mmg-pgx folder.
- Configure the config.sh of pgx. For more details, see <u>Configure the config.sh File of PGX</u> section.
- Copy the graph-keystore.p12 from MMG Installation server to <pgx installation path>/mmgpgx/conf. For more details, see Generate GRAPH-KEYSTORE.P12 section.
- 6. Copy the below key files from <MMG Installation path>/OFS_MMG/conf to <pgx installation path>/mmg-pgx/conf.
 - public.key
 - private.key
- Run the install.sh from <pgx installation path>/mmg-pgx/bin
- 8. Update the pgx-server URL in config.sh for ##PGX_SERVER_URLS## in the <MMG Installation path>/bin and run the install.sh -u command and restart the MMG services. For more details, see Configure the config.sh File of PGX section.
- 9. Start the Server. For more details, see Starting PGX Server section.
- 10. Stop the Server. For more details, see Stopping PGX Server section.

Configure the config.sh File of PGX

To configure the config.sh file for installing PGX with MMG, follow these steps:

- Login to the server as a non-root user.
- Navigate to the <OFS_MMG>/mmg-pgx/bin directory.
- 3. Configure the applicable config.sh attributes as shown in the following table:

Sample Config.sh file

```
#!/bin/sh
export PGX_PORT=##PGX_PORT##
export PGX_CONTEXT_PATH=##PGX_CONTEXT_PATH##
export PGX_SSL_ENABLED=##PGX_SSL_ENABLED##
export PGX_SSL_KEYSTORE=##PGX_SSL_KEYSTORE##
export PGX_SSL_KS_SECRET=##MMG_SSL_KS_SECRET##
export PGX_SSL_KS_TYPE=## PGX_SSL_KS_TYPE ##
export PGX_SSL_KS_ALIAS=## PGX_SSL_KS_ALIAS##
export GRAPH_SERVICE_URL=## GRAPH_SERVICE_URL##
export GRAPH_KEYSTORE_PASSWORD=## GRAPH_KEYSTORE_PASSWORD##
export LOG_HOME=##LOG_HOME##
export LOG_LEVEL=##LOG_LEVEL##
```



Table 5-2 Config.sh file of PGX

Parameter	Description	Is Mandatory	Comments
##PGX_PORT##	Port on which PGX server needs to be run.	YES	If not set, Port defaults to 7007.
##PGX_CONTEXT_PA TH##	Context path of PGX server	YES	If not set, Context path defaults to PGX.
##PGX_SSL_ENABLE D##	The values can be true /false.	YES	
	If true, follow the below steps if Self Signed is being used:		
	 Import PGX server.cer file to MMG server java keystore 		
	 Import MMGserver.cer file to PGX server java keystore 		
	For more details, see Import Server Certificate to Java Keystore section.		
Properties if ##PGX_SS	SL_ENABLED## is set to	true.	
##PGX_SSL_KEYSTO RE##	Absolute path for the keystore file.	YES	/conf/server.keystore. Include the file name in
	This is applicable only if ##PGX_SSL_ENABLE D## is set to true.		the path.
	NOTE:		
	Run the following command to create a keystore:		
	keytool -genkey -v - alias demoalias -keyalg RSA -keysize 2048 - keystore		
	server.keystore -validity 3650 -keypass secret - storepass secret - storetype PKCS12		
##PGX_SSL_KS_SEC RET##	Value passed in above command for keypass.	YES	Keystore password
	This is applicable only if ##PGX_SSL_ENABLE D## is set to true.		
##PGX_SSL_KS_TYP E##	The type of the PGX keystore.	YES	PKCS12
	This is applicable only if ##PGX_SSL_ENABLE D## is set to true.		



Table 5-2 (Cont.) Config.sh file of PGX

Parameter	Description	Is Mandatory	Comments
##PGX_SSL_KS_ALIA S##	The Alias of the PGX keystore.	YES	password123
	This is applicable only if ##PGX_SSL_ENABLE D## is set to true.		
Properties for graph service			
##	Graph Service URL.	YES	http(s):// <mmg< td=""></mmg<>
GRAPH_SERVICE_UR L ##	The value is same as ##LOADGRAPH_BASE _URL## in the MMG.config.sh		Host>: <graph port="" service="">/graph-service</graph>
## GRAPH_KEYSTORE_	Graph Keystore password.	YES	password123
PASSWORD ##	The value is same as ##GRAPH_KEYSTOR E_PASSWORD## in the MMG.config.sh		
Properties for setting log	g path		
##LOG_HOME##	A writable folder that stores PGX logs.		/scratch/users/logs
##LOG_LEVEL##			The values can be DEBUG/INFO/WARN

Starting PGX Server

To start the PGX Server, run the following command:

Navigate to <MMG_INSTALLATION_PATH>/bin directory. ./startup.sh

You may check <mmg-pgx/pgx-<pgx-version/bin/nohup.out to check if the UI service comes up properly.

A message similar to following means a successful startup:

INFO: Starting ProtocolHandler ["http-nio-7007"]

This will start the successful installation of PGX Server.

Stopping PGX Server

To stop the PGX Server, run the following command:

./shutdown.sh

A message similar to following means a successful shutdown:

PGX Server shutdown is complete.

R Interpreter

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



Installing Oracle R Distribution

https://docs.oracle.com/en/database/oracle/machine-learning/oml4r/2.0.0/oread/install-oracle-rdistribution-linux.html

- Check installation:
 - R --version

R 4.1.2 Installation



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:

- Set Proxy, (pseudo user):
 - curl -O https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
 - b. yum install epel-release-latest-7.noarch.rpm
 - curl -O https://cdn.rstudio.com/r/centos-7/pkgs/R-\${R_VERSION}-1-1.x86_64.rpm
 - sudo yum install R-\${R_VERSION}-1-1.x86_64.rpm
 - sudo In -s /opt/R/\${R_VERSION}/bin/R /usr/bin/R
- Check installation:
 - a. R --version

MMG Connection Objects Library Setup

This section describes the MMG Connection Objects Library Setup.

For detailed description on installation and installation of Oracle R Enterprise Client libraries (ROracle, and so on), check following link:

https://docs.oracle.com/cd/E83411 01/OREAD/installing-ORE-client.htm#OREAD167

Installing ROracle Library

Prerequisites

DBI is one of the dependencies for using this library.

- **Installing DBI**
 - 1. curl O https://cran.r-project.org/src/contrib/Archive/DBI/DBI 1.1.1.tar.gz
 - 2. R CMD INSTALL DBI_1.1.1.tar.gz

Procedure

To install ROracle Library, follow these steps:

For ORD 3.6.1/R 4.1.2



- curl O https://cran.r-project.org/src/contrib/ROracle 1.3-1.1.tar.gz
- 2. Install Oracle Instant Client Sdk Package. This is required for additional header files and an example makefile for developing Oracle Applications with Instant Client.
- 3. Oracle client lib must be present in PATH. In the .profile file, set PATH to include the appropriate \$ORACLE_HOME/bin path.

For example:

PATH=\$JAVA HOME/bin:\$ORACLE HOME/bin

4. R CMD INSTALL --configure-args='--with-oci-lib=<absolute-path-to-oracle-client-lib> --with-oci-inc=<absolute path to instantclient_21_5>/include' ROracle_1.3-1.1.tar.gz

For example:

R CMD INSTALL --configure-args='--with-oci-lib=/scratch/users/oracle/app/oracle/product/19.3.0/client_1/lib --with-oci-inc=/scratch/users/oracle/instantclient-sdk/instantclient_21_5/sdk/include' ROracle_1.3-1.1.tar.gz

Installing RODBC Library

- For ORD 3.6.1
 - 1. curl -O https://cran.r-project.org/src/contrib/Archive/RODBC/RODBC 1.3-16.tar.gz
 - 2. R CMD INSTALL RODBC_1.3-16.tar.gz

(i) Note

It needs write permission to $\widtharping{'usr/lib64/R/library'}$ or similar root directory for system installation.

- For R 4.1.2
 - curl -O https://cran.r-project.org/src/contrib/Archive/RODBC/RODBC 1.3-16.tar.gz
 - 2. R CMD INSTALL RODBC_1.3-16.tar.gz



(i) Note

LD_LIBRARY_PATH should contain path to \$ORACLE_HOME/lib and check that file 'libsqora.so.19.1' exists in \$ORACLE_HOME/lib. Now, set an environment variable named RODBC_DRIVER with value 'libsqora.so.19.1' whichever is present in \$ORACLE_HOME/lib/ directory based on the Oracle Client Version Installation. Now for RODBC Connection to work for Sandbox, check the TNS_ADMIN path set, and then in tnsnames.ora, add the connection string details with alias as Sandbox Name. For example, if Sandbox Name is SAND1 for which the datasource is on host abc.in.oracle.com , port 1234 and service name — ABCXYZ, then in tnsnames.ora file add the following entry-

```
SAND1 =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL =
TCP)(HOST=abc.in.oracle.com)(PORT=1234))
)
(CONNECT_DATA =
(SERVICE_NAME = ABCXYZ)
)
)
```

If this only does not resolve the connections, then configure odbcinst.ini. / odbc.ini files as well as mentioned in Oracle Client Installation and Setup (figured by: > odbcinst -j)

Installing the IRkernel

To configure IRkernel, follow these steps:

1. The IRkernel requires jupyter installed. Create a conda environment with the IRkernel dependencies:

```
conda create -p <PATH TO YOUR CONDA ENV> -y --override-channels -c
<CHANNEL URL> jupyter_client==7.2.2 grpcio protobuf
```

Install the IRkernel in R: Open your R REPL (R does not have to be installed with conda) and run:

```
install.packages('IRkernel')
```

Softlink jupyter and register IRkernel:

```
ln -s <PATH TO YOUR CONDA ENV> /usr/bin/jupyter && R -e
"IRkernel::installspec(user = FALSE)"
```



Configuring R Interpreter

Make sure to configure the interpreter config and set the correct path for R binary (zeppelin.R.cmd) and the path to Python environment which has the IRkernel installed (zeppelin.python).

In config.sh there is a new parameter:

export R PYTHON HOME=##R PYTHON HOME##

This is the Python Conda environment path till <PATH TO YOUR CONDA ENV>/bin/python created following steps mentioned in 5.9, which will set the zeppelin.python property in mmg-studio/server/builtin/interpreters/r.json file.

The other property mentioned for R binary (zeppelin.R.cmd) is also set in above path the default value is R so make sure that the user starting the services has access to R and alias is set or change the property according to the path of R binary.



(i) Note

Steps to enable TLS during R Interpreter Installation

Steps to enable TLS during R Interpreter Installation:

- Download Rserve 1.8-10.tar.qz from this link: https://cran.r-project.org/src/contrib/ Archive/Rserve/
- Install openssl-devel [yum install openssl-devel]
- Execute below cmd:

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

The below steps can be executed from R session to check that Rserve configuration is working before starting the services:

Start R from Terminal/Putty and execute the below commands:

- library(Rserve)
- Rserve(args="--RS-conf <Rserve.conf file location including file name> --nosave")

Using MMG Studio to Oracle Connection Objects

This section describes the Using MMG Studio to Oracle Connection Objects.

Workspaces

- mmg.list_workspaces(): Used to fetch a vector of all workspaces. For example: vec <- mmq.list workspaces() vec will be vector object
- mmg.attach_workspace("workspace_name"): A method used to set workspace. Sets a global mmg attached WS variable with value of workspace name

Sets a mmg_DS_Vec Vector Object with name and order of all datasources for attached workspace.



Sets a mmg_WL_Vec Vector Object with name and wallet of all datasources for attached workspace.

For example:

mmq.attach workspace("SB1")

Connections

Following is the list of datasources related to workspace using:

 mmg.list_datasources("SB1", 1): will list datasources related to SB1 workspace with order 1 as passed in second argument

 ${\tt mmg.list_datasources("workspace_name", order)}$ order is integer for specific order or null for all datasources.

For example:

df <- mmg.list datasources("workspace name",order) df will be Data.Frame Object.

From the datasource name or order for the attached workspace, we can get the **ROracle** or **RODBC** Connection Object.

mmg.get_connection():

datasource_name is the string name of the datasource, order is integer, library is one of "RODBC" or "ROracle"

```
conn <- mmg.get_connection(datasource=order,conn_type="library");
```

conn <- mmg.get_connection(datasource="datasource_name",conn_type="library");</pre>

conn <- mmg.get_connection(datasource="datasource_name","library");</pre>

conn <- mmg.get connection(datasource=order,"library");</pre>

conn <- mmg.get_connection("datasource_name",conn_type="library");</pre>

conn <- mmg.get_connection(order,conn_type="library");</pre>

conn <- mmg.get connection("datasource name","library");</pre>

conn <- mmg.get connection(order,"library");</pre>

sets the conn variable to connection object of relevant library

Multi Level Approval

Model Pipeline deployment process by default requires one level of approval for every stage including model pipeline acceptance, promotion to production, and so on.

The requestor is allowed to select Reviewer and Approver user groups. All the user groups with MDLREVIEW function mapped to them are displayed in the Reviewers selector field. Similarly, the user groups with the MDLAPPROVE function mapped to them are displayed in the Approvers selector field. Applicable Pending requests are shown in the Reviewer/Approver tabs.

To add multi level approvers or reviewers, perform the following:

Navigate to <installation directory>/MMG/OFSMMG/conf/workflow/model-pipeline/default.yml

Following are the default values:

workflow:

workflow-name: Default Workflow



```
num-approver-levels: 1
levels:
- level: 1
approvers:
escalation-approvers:
escalation-trigger-time: 0
lock-approver-selection: false
enable-approver-notification: true

Modify the approver levels based on your requirements as shown below.
```

Figure 5-1 Multi level approval

```
workflow:
workflow-name: Default Workflow
num-approver-levels: 2
levels:
- level: 1
approvers: APPROVER1
escalation-approvers: MDLAPPR
- level: 2
approvers: APPROVER2
escalation-approvers: MDLAPPR
escalation-trigger-time: 12
enable-approver-notification: true
lock-approver-selection: true
```

Kafka Installation

Windows

Apache Zookeeper (Download the bin tar):

https://shaaslam.medium.com/installing-apache-zookeeper-on-windows-45eda303e835#.fgofwm6n6

Apache Kafka:

https://shaaslam.medium.com/installing-apache-kafka-on-windows-495f6f2fd3c8

Linux

Method 1(Using separate zookeeper)

Apache Zookeeper

- 1. Download Apache Zookeeper bin files from https://www.apache.org/dyn/closer.lua/zookeeper/zookeeper-3.7.1/apache-zookeeper-3.7.1-bin.tar.gz
- 2. Untar apache-zookeeper-3.7.1-bin.tar.gz tar -zxvf apache-zookeeper-3.7.1-bin.tar.gz
- 3. In conf folder, rename zoo-sample.cfg to zoo.cfg



- 4. Make a folder named data in /scratch/users/ofsaa/zookeeper/apache-zookeeper-3.7.1-bin
- 5. Change the value of dataDir in zoo.cfg to dataDir=/scratch/users/ofsaa/ zookeeper/apache-zookeeper-3.7.1-bin/data
- 6. Open a terminal and set the below values:

```
export ZOOKEEPER_HOME=/scratch/users/ofsaa/zookeeper/apache-
zookeeper-3.7.1-bin
```

export PATH=/scratch/users/ofsaa/zookeeper/apache-zookeeper-3.7.1bin/bin:\$PATH

7. Start the zookeeper server:cd /scratch/users/ofsaa/zookeeper/apache-zookeeper-3.7.1-bin/bin ./zkServer.sh start

A few commands for zookeeper server:

- ./zkServer.sh start
- ./zkServer.sh start-foreground
- ./zkServer.sh stop
- ./zkServer.sh status

Apache Kafka

Download Apache Kafka bin files

wget https://downloads.apache.org/kafka/3.3.1/kafka_2.13-3.2.0.tgz

- Untar kafka_2.13-3.2.0.tgz
 tar -zxvf kafka_2.13-3.2.0.tgz
- Make a folder "logs" in the directory /scratch/users/ofsaa/kafka/kafka_2.13-3.2.0/.
- Go to config folder in Apache Kafka and edit server.properties using any text editor. Change value of logs.dir to log.dirs=/scratch/users/ofsaa/kafka/kafka_2.13-3.2.0/logs in server.properties
- Go to config folder in Apache Kafka and edit zookeeper.properties using any text editor. Change value of dataDir dataDir=/scratch/users/ofsaa/zookeeper/apache-zookeeper-3.7.1-bin/data
- Start Apache Kafka broker

```
cd /scratch/users/ofsaa/kafka/kafka_2.13-3.2.0/bin/
```

./kafka-server-start.sh ../config/server.properties

Method 2(Using zookeeper shipped with Apache Kafka)

Apache Zookeeper and Kafka

Download Apache Kafka bin files

wget https://downloads.apache.org/kafka/3.3.1/kafka_2.13-3.2.0.tgz

- Untar kafka_2.13-3.2.0.tgz
 tar -zxvf kafka 2.13-3.2.0.tgz
- Go to config folder in Apache Kafka and edit zookeeper.properties using any text editor. Change value of dataDir to dataDir=/scratch/users/ofsaa/zookeeper/apache-zookeeper-3.7.1-bin/data or any folder where the snapshot of zookeeper metadata can be stored.
- Make a folder "logs" in the directory /scratch/users/ofsaa/kafka/kafka_2.13-3.2.0/.



- Go to config folder in Apache Kafka and edit server.properties using any text editor. Change value of logs.dir to log.dirs=/scratch/users/ofsaa/kafka/kafka_2.13-3.2.0/logs in server.properties.
- Start Apache Zookeeper

cd /scratch/users/ofsaa/kafka/kafka/kafka_2.13-3.2.0

bin/zookeeper-server-start.sh config/zookeeper.properties

Start Apache Kafka Broker

cd /scratch/users/ofsaa/kafka/kafka_2.13-3.2.0

bin/kafka-server-start.sh /config/server.properties

Create Keys and Certificates

- keytool -keystore kafka.server.keystore.jks -alias localhost -keyalg RSA -validity 365 -genkey -storepass oracle123 -keypass oracle123 -ext
 SAN=DNS:ofss-mum-1035.snbomprshared1.gbucdsint02bom.oraclevcn.com
- openssl req -new -x509 -keyout ca-key -out ca-cert -days 365
- keytool -keystore kafka.client.truststore.jks -alias CARoot -storepass oracle123 -keypass oracle123 -importcert -file ca-cert keytool -keystore kafka.server.truststore.jks -alias CARoot -storepass oracle123 -keypass oracle123 -importcert -file ca-cert
- keytool -keystore kafka.server.keystore.jks -alias localhost -storepass oracle123 -keypass oracle123 -certreq -file cert-file
- openssl x509 -req -CA ca-cert -CAkey ca-key -in cert-file -out cert-signed days 365 -CAcreateserial -passin pass:oracle123
- keytool -keystore kafka.server.keystore.jks -alias CARoot -storepass oracle123
 -keypass oracle123 -importcert -file ca-cert keytool -keystore
 kafka.server.keystore.jks -alias localhost -storepass oracle123 -keypass
 oracle123 -importcert -file cert-signed

Converting Keystore file from jks to pem

- keytool -exportcert -alias localhost -keystore kafka.server.keystore.jks -rfc
 -file certificate.pem -storepass oracle123
- keytool -v -importkeystore -srckeystore kafka.server.keystore.jks -srcalias localhost -keystore cert_and_key.p12 -storetype PKCS12 -storepass oracle123 srcstorepass oracle123
- openssl pkcs12 -in cert_and_key.p12 -nodes -nocerts -out key.pem -passin pass:oracle123
- keytool -exportcert -alias CARoot -keystore kafka.server.keystore.jks -rfc file CARoot.pem -storepass oracle123

Inside the Kafka/config path we have to create client-ssl.properties and server-ssl.properties.

The client-ssl.properties file contains the following:

- security.protocol=SSL
- ssl.truststore.location=/scratch/mmg8133/kafka/ssl/kafka.server.truststore.jks
- ssl.truststore.password=oracle123
- ssl.keystore.location=/scratch/mmg8133/kafka/ssl/kafka.server.keystore.jks
- ssl.keystore.password=oracle123



ssl.key.password=oracle123

server-ssl.properties contains the same content as server.properties, but we have to update the ssl parameters in it.

Configuring Interpreters

This section provides information on configuring interpreters.

Spark Interpreter



Java 17 supports the Spark3.4+ version, excluding the 4.0.0 series.

Spark Configuration and Execution in Python

The following sections provide information on the Spark configurations and executions in Python.

Set the Hadoop Clusters

Configure \$ORACLE_HOME, \$HADOOP_HOME to validate the installation directories.

- 1. Copy the following required Oracle JDBC/security JARS to HDFS.
- 2. Execute the following commands that need to be run on the Hadoop cluster:
 - \$HADOOP_HOME/hdfs dfs -mkdir -p /spark/jars/
 - \$HADOOP_HOME/bin/hdfs dfs -put \$ORACLE_HOME/lib/ojdbc8.jar /spark/jars/
 - \$HADOOP_HOME/bin/hdfs dfs -put \$ORACLE_HOME/jlib/oraclepki.jar /spark/ jars/
 - \$HADOOP_HOME/bin/hdfs dfs -put \$ORACLE_HOME/jlib/osdt_core.jar /spark/jars/
 - \$HADOOP_HOME/bin/hdfs dfs -put \$ORACLE_HOME/jlib/osdt_cert.jar /spark/jars/
- 3. Verify that the jars are copied by executing the following command: hdfs dfs -ls /spark/jars

Configuration on the OFS MMG Instance Server

Perform the following steps to complete the configurations on the OFS MMG instance server.

- Create a local folder on the OFS MMG install server, for example, hive_conf_dir:
 - /scratch/ofsaaapp/spark-lib/hive_conf_dir/yarn-site.xml (suggested some edits now to make the code work)
 - /scratch/ofsaaapp/spark-lib/hive_conf_dir/mapred-site.xml
 - /scratch/ofsaaapp/spark-lib/hive_conf_dir/hdfs-site.xml
 - /scratch/ofsaaapp/spark-lib/hive conf dir/core-site.xml



- /scratch/ofsaaapp/spark-lib/hive_conf_dir/hive-site.xml
- /scratch/ofsaaapp/spark-lib/hive_conf_dir/kbank.keytab
- /scratch/ofsaaapp/spark-lib/hive_conf_dir/krb5.conf
- 2. Configure the yarn-site.xml file with the following property:

Note

property>

Ensure that the Hadoop cluster and the OFS MMG instance are configured to run on the same Java version.

Set Apache Spark in the OFS MMG Installed Server

Perform the following steps to set Apache Spark in the OFS MMG installed server:

 Download Apache Spark from the Apache site and copy it to the OFS MMG installed server:

```
https://archive.apache.org/dist/spark/spark-3.5.0/spark-3.5.0-bin-hadoop3.tgz
```

- 2. Unzip the downloaded file by executing the following command: gunzip spark-3.5.0-bin-hadoop3.tgz
- 3. Untar the file with the relevant command.
- 4. Restart the OFS MMG services after placing the configs and installing Spark.

Code Snippet to Create the Spark Dataframe and Trigger Executions in to Spark Cluster (with PySpark)

You must perform the following steps to create the Spark Dataframe and trigger the executions in the Spark cluster with PySpark



- 1. Log in to the OFS MMG application
- 2. Navigate to the **Notebook** tab of the Model pipeline.
- 3. Create and execute following paragraphs:
 - a. Environment Configuration
 - b. Connection
 - c. Spark Session
 - d. Sample Query

Environment Configuration

```
%python
import sys, os
jdk_path = "/usr/java/jdk1.8.0_401" # Hadoop supports Java 1.8 and Java 11. Refer
matrix for actual versions. Both cluster and client must be in same version
spark_path = "<Path of spark installed location in MMG Server>" # This is
spark3.5.5 + hadoop3.
# Ensure that spark version from the above matches with the pyspark installed in
this python runtime.
# Also ensure that hadoop version from the above matches with the hadoop version
of cluster
hadoop_conf_dir = "<Path to Hadoop xml directory which is copied local to MMG
Server>" # Local Path containing hadoop XMLs. Ensure that yarn-site has the
correct addresses for resource manager.
krb5_conf = "<Path_To_krb5.conf' # Conf file for hive connectivity</pre>
default_service_principal = "<service>/<fully-qualified-</pre>
hostname>@<KERBEROS REALM>" # Service Principal, not the user. Needed while doing
impersonation.
default_keytab_file = "<Path_To_Keytab file copied local to the MMG server>" #
Keytab file for initializing connection
default_user_principal = "<service>/<fully-qualified-hostname>@<KERBEROS_REALM>"
"""Setup environment variables for Spark and Hadoop."""
os.environ["JAVA_HOME"] = jdk_path
os.environ["SPARK HOME"] = spark path
os.environ["PYSPARK_PYTHON"] = sys.executable
os.environ["PYSPARK DRIVER PYTHON"] = sys.executable
os.environ["KRB5_CONFIG"] = krb5_conf
os.environ["HADOOP_CONF_DIR"] = hadoop_conf_dir
os.environ["YARN_CONF_DIR"] = hadoop_conf_dir
os.environ["PATH"] = f"{jdk path}/bin:{spark path}/bin:" + os.environ["PATH"]
```



```
os.environ["SPARK_SUBMIT_OPTS"] = f"-Djava.security.krb5.conf={krb5_conf}" # If
we still get "KrbException: Cannot locate default realm", then this property
needs to be added in .profile
print("Environment Configured")
```

Connection

```
%python
from pyspark.sql import SparkSession
def get_yarn_connection(conn_type='kerberos',
user_principal=default_user_principal, keytab_file=default_keytab_file,
extraConfigs=None, hive support=False):
extraConfigs = extraConfigs or {}
if not isinstance(extraConfigs, dict):
raise TypeError("extraConfigs must be a dict")
if conn type.lower() == 'kerberos':
builder = (
SparkSession.builder
.appName("MMGYarnKerberos")
.master("yarn")
.config("spark.submit.deployMode", "client")
.config("spark.kerberos.principal", user_principal)
.config("spark.kerberos.keytab", keytab_file)
)
elif conn_type.lower() == 'impersonation':
builder = (
SparkSession.builder
.appName("MMGYarnImpersonate")
.master("yarn")
.config("spark.submit.deployMode", "client")
.config("spark.sql.hive.thriftServer.singleSession", "true")
.config("spark.hadoop.hive.server2.proxy.user", user_principal) # Proxy/
Impersonated user. Ensure proxy login is enabled in HIVE.
.config("spark.hadoop.hive.metastore.kerberos.principal",
default_service_principal) # Service principal
.config("spark.hadoop.hive.metastore.kerberos.keytab.file", keytab_file) # Should
be the keytab of service principal, not the user
.config("spark.hadoop.hive.metastore.sasl.enabled", "true")
```



```
.config("spark.sql.catalogImplementation", "hive")
)
else:
raise ValueError(f"Unsupported connection mode {conn_type} to get YARN
connection.")
for k, v in extraConfigs.items():
builder = builder.config(k, v)
if hive_support:
builder = builder.enableHiveSupport()
spark = builder.getOrCreate()
return spark
```

Spark Session

```
%python
spark = get_yarn_connection(conn_type='kerberos')
sc = spark.sparkContext
print("Spark app name:", sc.appName)
print("Spark app ID:", sc.applicationId)
print("Spark master :", sc.master)
print("Local Spark URL :", sc.uiWebUrl)
```

Sample Query

```
%python
spark.sql("show databases").show()
spark.sql("SHOW TABLES IN default").show()
```



The example above uses Spark's Hive support (Metastore).

Interpreter Configuration

For JDBC

Update the below proeprty in jdbc.json underOFS_MMG/mmg-studio/server/builtin/interpreters "propertyName": "default.url" "defaultValue": "<JDBC_URL>"



For example: jdbc:oracle:thin:@ofss-

mum-1033.snbomprshared1.gbucdsint02bom.oraclevcn.com:15 21/MMG19PDB

"propertyName": "default.user",

"defaultValue": "<schameusername>"

For example: The schema user to which you want to connect, for example: datastudio schema name.

"propertyName": "default.password",

"defaultValue": "<schemapassword>"

For example: Password of the provided schema user.

Start the jdbc interpreter by executing below command under //OFS_MMG/mmg-studio/interpreter-server/jdbc-interpreter-22.4.3/bin ./

jdbc-interpreter

If the jdbc interpreter needs to be included in the datastudio startup script remove the below entry from /OFS_MMG/mmg-studio/bin/startup.sh --jdbc -1.

For Spark

Configuration with Kerberos enabled remote spark cluster:

- 1. Copy the configured Spark directory from hadoop cluster to <MMG Studio>/interpreter-server/spark-interpreter/extralibs. For example: spark-2.4.8-bin-hadoop2.7
- 2. Copy the below files to the <MMG Studio>/interpreter-server/spark-interpreter/extralibs krb5.conf <keytabfile>.keytab
- 3. To run Spark in yarn-client mode, configure the following parameters in this file OFS_MMG/mmg-studio/server/builtin/interpreters/spark.json spark.master = yarn-client

spark.driver.bindAddress = 0.0.0.0

spark.driver.host = <host> -> Apache Spark host

(i) Note

When using the Kubernetes interpreter lifecycle, <host> can be the IP address or hostname of any node in your Kubernetes cluster. When using the Host interpreter lifecycle, <host> should be the IP address or hostname of the node that runs the Spark interpreter.

(i) Note

When connecting to a YARN cluster, the Spark driver authenticates as the UNIX user that runs the Spark interpreter. You can set the HADOOP_USER_NAME environment variable to make the Spark driver authenticate as a different user. If you use the Host interpreter lifecycle, then you can do this by exporting the HADOOP_USER_NAME environment variable before starting the Spark interpreter process. If you us the Kubernetes interpreter lifecycle, then you can do this by setting the HADOOP_USER_NAME environment variable in the resource manifest (spark.yml).



- Update file spark-defaults.conf keytab location to the location where <keytabfile>.keytab file is copied
- 5. Update file spark-env.sh with the krb5.conf location to the location where krb5.conf file is copied.

For example: Djava.security.krb5.conf=/OFS_MMG/mmg-studio/interpreter-server/spark-inte rpreter-22.4.2/extralibs/krb5.conf".

Post-installation Steps

On successful installation of the application, refer to the below topics for post-installation procedures.

Access the Application

To access the application, follow these steps:

Open a browser and enter the URL in the following format:

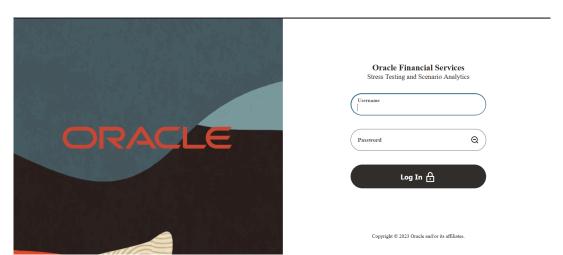


Figure 6-1 Login window – AAI Authentication

For more information, see the User Access and Permissioning Management section.

Create Application Users

Create the application users in the setup before starting to use the application. For more information, see the User Access and Permissioning Management section.

Map Application User(s) to User Group

User Groups seeded with the OFS MMG Application Pack are listed in the Seeded User Groups table.



(i) Note

Run the following scripts manually for the user user-group mapping/unmapping in OFS AAI:

The following two flags must be added in the configuration table of the config schema:

```
MERGE INTO CONFIGURATION a USING (SELECT 1 FROM DUAL) b ON ( a.PARAMNAME= 'ENB_CSTM_GRP' ) WHEN NOT MATCHED THEN INSERT ( a.PARAMNAME, a.PARAMVALUE, a.DESCRIPTION) VALUES ('ENB_CSTM_GRP', 'true', 'Enable custom group creation during JIT')
```

```
MERGE INTO CONFIGURATION a USING (SELECT 1 FROM DUAL) b ON ( a.PARAMNAME= 'ENB_GRP_SYNC' ) WHEN NOT MATCHED THEN INSERT ( a.PARAMNAME, a.PARAMVALUE, a.DESCRIPTION) VALUES ('ENB_GRP_SYNC', 'true', 'Enable unmapping operation during JIT')
```

Two new flags are introduced in configuration table:

- **ENB_CSTM_GRP** If this flag is set as *true*, then custom groups can be created. If this flag is set as *false*, then the user will not be able to create custom groups and assign it to the user.
- **ENB_GRP_SYNC** If this flag is set as *true*, then the unmapping of users and groups are enabled. If this flag is set as *false*, then the user will not be able to unmap groups and users.

Based on these flags, unmapping of users with groups on user login is enabled.

Table 6-1 Seeded User Groups

User Group Name	User Group Description
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 all the menu items in the OFS MMG Application. Additionally, they have authorization rights to create and populate workspaces.
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.



Table 6-1 (Cont.) Seeded User Groups

User Group Name	User Group Description
DSUSRGRP	General Role
	Users mapped to this group have permission to access/modify MMG Studio 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.

Note

Admin link in the application home page is accessible only if the below seeded groups are mapped to the user:

- IDNTYADMN
- IDNTYAUTH

User Access and Permissioning Management

The application uses a realm based on unique authentication and authorization for its users. Realm indicates the functional grouping of Database Schemas and roles that must be secured for an application.

Realms protect data from access through system privileges and do not provide its owner or participants additional privileges. Realm based authorization establishes a set of database accounts and roles that can manage, or access objects protected in realms and are authorized to use its system privileges. It provides a runtime mechanism to check logically if a user's command can access objects specified in the command and proceed with its execution. Realms (AAIRealm, SAMLRealm) are selected based on the Identity Provider (IDP) during the installation. For more information, see the OFS MMG Installation Guide. After you select the realms, you can register a set of schema objects or roles (secured objects) for realm protection and authorize a set of users or roles to access the secured objects. The Application is accessed using the following realms that you have selected during the installation of the Application:

• AAIRealm: This uses Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Identity Management System for User Authentication. Users, Roles, and Groups



are created in the OFSAAI. The OFSAAI facilitates System Administrators to provide access, monitor, and administer users along with the infrastructure metadata operations. The required permissions to roles or groups are authorized in the application by using the Permission feature.

SAMLRealm: The SAMLRealm uses an identity provider (IDP) Identity Management
System for User Authentication. Security Assertion Markup Language (SAML) is an open
standard that allows Identity Providers (IDP) to pass authorization credentials to Service
Providers (SP). IDP acts as the Single Sign-On (SSO) service. Users and Roles are
created in the IDP. The required permissions to Users and Roles are authorized in the
application by using the Permission feature.

The following image illustrates the authentication and authorization process in the application.

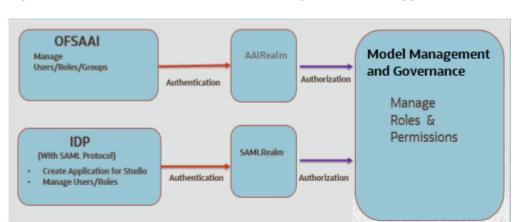


Figure 6-2 Authentication and Authorization process in the application

Mapping User Groups

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

User Groups

Table 6-2 User Groups

User Group	Description
IDNTYADMN	Identity Administrator group
IDNTYAUTH	Identity Authorizer group
MDLREV	The Modeling Reviewer Group. Users mapped to this group have access to the menu items in the 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.



Table 6-2 (Cont.) User Groups

User Group	Description
WKSPADMIN	The Workspace Administrator Group. Users mapped to this group have access to create and populate workspaces. For viewing the 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 application that is related to model creation.
DSUSRGRP	Data Studio User Group This User Group provide access to modify Interpreter configurations.
GRPADMIN	The Graph Administrator Group Users mapped to this group have access to all the menu items in the application related to graph as well as 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 application related to graph as well as Pipeline/Refresh graphs related health services.
DSREDACTGRP	Roles for applying redaction in graph. This group will be applicable to only those users for whom graph redaction is required. Note: This group has to be created manually in AAI and map it to the users.
FILEADMINUSER	User Group for admin level access to files
FILEREADUSER	User Group for read access to files
FILEWRITEUSER	User Group for read and write access to files

(i) Note

- At the first-time login, User Group mappings are initialized from AAI/IDCS for the newly provisioned users. These will be reflected in Admin Console in the next login.
- If User Group mappings are deleted in AAI/IDCS, it would not delete in the Admin Console. Admin needs to delete this in the 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.

User Group - Role Mapping

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

Table 6-3 User Group to Role Mapping

Group Name	Role Name
DSREDACTGRP	DSREDACT
IDNTYADMN	Batch Advance Role



Table 6-3 (Cont.) User Group to Role Mapping

	5.1.30
Group Name	Role Name
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
MDLAPPR	Model Deployment
MDLAPPR	Workspace Read
MDLAPPR	Model Read
MDLAPPR	Model Access
MDLAPPR	Workspace Access
MDLAPPR	DSAPPROVER
MDLBATCHUSR	DSBATCH
MDLREV	Workspace Read
MDLREV	Model Review
MDLREV	Model Access
MDLREV	Workspace Access
MDLREV	DSUSER
MDLREV	Model Read
MDLUSR	Model Advanced
MDLUSR	Model Write
MDLUSR	Model Read
MDLUSR	Batch Advance Role
MDLUSR	Model Execute
MDLUSR	DSUSER
MDLUSR	Model Access
MDLUSR	Workspace Access
MDLUSR	Workspace Read
MDLUSR	Datastore Access
MDLUSR	Datastore Write
MDLUSR	Datastore Read
WKSPADMIN	Workspace Access
WKSPADMIN	DSADMIN
WKSPADMIN	Identity MGMT advanced
	ye waranood



Table 6-3 (Cont.) User Group to Role Mapping

Group Name	Role Name
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

Functions and Roles required to perform CRUD operations for Conda

The following table provides details about the Functions and Roles required to perform CRUD operations for Conda in the application.

Table 6-4 Functions and Roles

Function	Role	Groups Mapped	Access
CONDAENVSUMM	CONDAENVACCESS	• MDLUSR	Summary view
		 MDLREV 	
		 MDLAPPR 	
CONDAENVVIEW	CONDAENVREAD	• MDLUSR	Read
		 MDLREV 	
		 MDLAPPR 	
CONDAENVEXP	CONDAENVREAD	• MDLUSR	Export yml file
		 MDLREV 	
		 MDLAPPR 	
CONDAENVEXP	CONDAENVWRITE	• MDLREV	Export yml file
		 MDLAPPR 	
CONDAENVDEL	CONDAENVWRITE	• MDLREV	Delete a registered
		 MDLAPPR 	conda environment
CONDAENVEDIT	CONDAENVWRITE	• MDLREV	Edit a conda
		 MDLAPPR 	environment
CONDAENVADD	CONDAENVWRITE	 MDLREV 	Add a conda
		 MDLAPPR 	environment

Access MMG using AAI Realm

This section provides information on creating users who can access MMG using the AAI Realm method of authentication through Oracle Financial Services Analytical Applications Infrastructure (OFSAAI). The users with SYSADMN and SYSAUTH roles in OFSAAI can create and authorize users, respectively.

Identity Management in the OFSAAI facilitates System Administrators to provide access, monitor, and administer users along with the infrastructure metadata operations. The Security Management System (SMS) component is incorporated with Password Encryption, Single



Logon, Role and DataBased Security, Access Control, and Audit Trail feature to provide a highly flexible security envelope. Administrators can create, map, and authorize users defining a security framework that can restrict access to the data and meta-data in the warehouse, based on a fine-grained access control mechanism. These activities are done at the initial stage and then on a required basis. For more information on creating and authorizing users in OFSAAI, see the Oracle Financial Services Analytical Applications Infrastructure User Guide. The following table describes the ready-to-use roles and the corresponding user groups who can access MMG using AAIRealm.

(i) Note

Only in AAIRealm, users are mapped to user groups. The default permissions mapped to these users and user groups are available in the Permission section. However, these permissions can be added or modified.

Prerequisites

Configuring WebLogic for REST Services Authorization.

To enable REST API authorization by OFSAA in WebLogic server, perform the following steps:

- a. Open the config.xml file located in the domain where OFSAA is deployed that is: <domain_home>/config/config.xml.
- b. Add the following in the security-configuration tag: <enforce-valid-basic-auth-credentials>false</enforce-valid-basic-authcredentials>.
- 2. In OFSAA Application, **Allow user to log in from multiple machines** option should be enabled.

REST API Calls

There is need for Model Scripts to access input parameters passed by REST API Calls:

- **a.** The REST API optional parameters are mapped to runtime parameters, this mapping needs to be documented. It is not apparent from runtime parameter section.
- b. The mapped simple params can be accessed with script segment like below , but it is not supporting base64 encoding and/or json hp = '\${hyperparams}' cnf = '\${conf}' 3.the enhancement against json is retracted as we can do base64 encrypted value passing

```
"RUN_ID": 2,

"RUN_CODE": 200440,

"PARAM_SET": {

"CALL_TYPE": "PASS_THROUGH_RATE",

"BOLLINGER_PERIOD": 3,

"BOLLINGER_COEFFICIENT": 1,

"DATA_SNAP_FREQ": 1,

"DATA_SNAP_FREQ_MULTIPLIER": "D",

"MISSING_VALUE_TECH": "inpol",

"MISSING_THRESHOLD PERCENTAGE": 30,
```



```
"START_DATE": 20230131,
"END DATE": 20230431,
"EXECUTED_BY": "TESTUSR",
"REFRESH DATA": "Yes"
},
"SLICE": [
"SEQ": 1,
"LOGICAL_PARTITION_NAME": "DEFAULT",
"TABLE NAME": "FSI I NM MLSTAGE"
}
],
"OUTPUT DATA": {
"PLOTS_TAB": "FSI_O_NM_PLOTS",
"MILESTONE_TAB": "FSI_O_NM_MILESTONE",
"SCORE TAB": "FSI O NM MODEL"
}
```

For example: base64 encryption

Access MMG using SAMLRealm

This section provides information on managing users who can access MMG with Identity Provider (IdP or IDP). An Identity Provider (IdP) is a service that stores and verifies user identity. IdPs are cloud-hosted services, and they often work with single sign-on (SSO) providers to authenticate users. An Identity Provider stores and manages users' digital identities. An IdP checks user identities via username-password combinations and other factors, or it may simply provide a list of user identities that another Service Provider (like an SSO) checks. The following are the ready-to-use roles that can access MMG using SAMLRealm. To integrate MMG with IdP as the SSO Provider, follow these steps:

Create the following roles in the IDP System:

For MMG:

- IDNTYADMN
- IDNTYAUTH
- MDLREV
- MDLAPPR
- MDLBATCHUSR
- WKSPADMIN
- MDLUSR
- DSUSRGRP
- DSREDACTGRP



- GRPADMIN
- GRPUSR

Note

IDNTYADMN role is required only if you need the Admin Access.

2. Map the user groups to the respective user based on the user roles. The default permissions mapped to these users are available in the Permission section. However, these permissions can be added or modified.

(i) Note

It is recommended to use AAIRealm or SAMLRealm.

Access the Application by Using SAML Realm

This section provides information on managing users who can access the application with Identity Provider (IdP or IDP). The IdP acts as the Single Sign-On (SSO) service provider for implementations between the application, Investigation Toolkit, and Enterprise Case Management. This configuration prevents separate login for each application.

An identity provider (IdP) is a service that stores and verifies user identity. IdPs work with single sign-on (SSO) providers to authenticate users. An identity provider (IdP or IDP) stores and manages users' digital identities. An IdP checks user identities via username-password combinations and other factors, or it may simply provide a list of user identities that another service provider (like an SSO) checks.

See the User Groups section for Pre-configured Groups to access the application using SAML Realm.

(i) Note

You can configure SAML in the following ways:

- SAML for Authentication and AAI for Authorization
- SAML for Authentication and SAML for Authorization
 For more information, see the respective sections in the Installation Guide.

To integrate the application with IDP as the SSO provider, follow these steps:

- Create the following Group in the IDP system. For more information on creating groups in IDP, see the OFS Admin Console User Guide.
 - Create the new groups with the same name as the pre-configured groups. For more information, see the User Groups section.
- Create a SAML application in IDP.
- Configure the SAML application. Key configurations in the SAML application is as follows:
 - Entity ID: https://<FQDN of <application name> Linux Server>:7001/
 <application name>



 Assertion Consumer URL: http://<FQDN of <application name> Linux Server>:7001/<application name>/home

(i) Note

- If the application Gateway service is enabled, then the value of this port will be the GATEWAY PORT. For example, 7071.
- Response in SAML response must be signed.
- Include Signing Certificate in Signature: Enabled
- Signature hashing algorithm: SHA-256
- Enable Single Logout: Enabled
- Logout Binding: POST
- **Single Logout URL (SAML_LOGOUT_URL)**: http://<FQDN of <application name>>:7001/<application name>/signoff

(i) Note

If the application Gateway service is enabled, then the value of this port will be the GATEWAY PORT. For example, 7071.

Logout Response URL: http://<FQDN of <application name>>:7001/
 <application name>/signoff

Note

If the application Gateway service is enabled, then the value of this port will be the GATEWAY PORT. For example, 7071.

- Encrypt Assertion: Disabled
- SAML Attribute Configuration
 Update the SAML attribute configuration as described in the following table.

Table 6-5 Attribute Configuration

Name	Format	Туре	Value	Condition
ofs_mapped_gro ups	Basic	User Attribute	Group Member	All Groups
email	Basic	User Attribute	Primary Email	-
username	Basic	User Attribute	Last Name	-

4. Create a user and map the user groups to the respective user based on the user roles.

Access Data Studio Using SAML Realm

This section provides information on managing users who can access Data Studio with Identity Provider (IdP or IDP). The IdP acts as the Single Sign-On (SSO) service provider for



implementations between the application, MMG and Data Studio. This configuration prevents separate login for each application.

An identity provider (IdP) is a service that stores and verifies user identity. IdPs work with single sign-on (SSO) providers to authenticate users. An identity provider (IdP or IDP) stores and manages users' digital identities. An IdP checks user identities via username-password combinations and other factors, or it may simply provide a list of user identities that another service provider (like an SSO) checks.

Users should map the following user groups to access the Data Studio and Investigation Toolkit:

DSUSRGRP: Grants admin privileges for Data Studio

To integrate Data Studio with IDP as the SSO provider, follow these steps:

- Create the following Group in the IDP system. For more information on creating groups in IDP, see the OFS Admin Console User Guide.
 - Create the new groups with the same name as the pre-configured groups. For more information, see the User Groups section.
- 2. Create a SAML application in IDP for Data Studio.
- 3. Configure the SAML application. Key configurations in the SAML application is as follows:
 - Entity ID: https://<Hostname>:7008/<application name>
 - Assertion Consumer URL: http://<Hostname>:7008/<application name>/saml/ consume
 - Include Signing Certificate in Signature: Enabled
 - Signature hashing algorithm: SHA-256
 - Enable Single Logout: No
 - Require encrypted assertion: No
- 4. Update the SAML attribute configuration as described in the following table.

Table 6-6 SAML Attribute Configuration

Name	Format	Туре	Value	Condition
ofs_mapped_grou ps	Basic	User Attribute	Group Member	All Groups
email	Basic	User Attribute	Primary Email	-
username	Basic	User Attribute	Last Name	-
group	Basic	User Attribute	Group Member	All Groups

- 5. Create a user and map the user groups to the respective user based on the user roles.
- 6. After creating the application, download the "Signing Certificate" of the SAML application of the Data Studio and rename it to "key.cert" file and place in the following location:
 - <<application name>_PATH>/mmg-home/mmg-studio/conf
- 7. Restart the application.

Model Techniques/ Model Library

Following are the prerequisites to use the model techniques from the older version when you upgrade to 8.1.2.4.0 version.





MMG_TECHNIQUE_MASTER table had no V_WORKSPACE_ID column, which has been added in this release and then the primary key is updated to (V_TECHNIQUE_ID, V_WORKSPACE_ID).

To use the existing Techniques in the upgraded setup, perform the below:

The V_WORKSPACE_ID column will have the value set as ##WORKSPACE## for the existing records by default. If the same records has to be used in the latest version of MMG, you must update the table MMG_TECHNIQUE_MASTER with relevant Workspace ID.

.PEM file creation for Model Service

You must create **server.pem** file from **server.keystore** in the same path where server.keystore file is present using the below command:

openssl pkcs12 -in <Path_To_server.keystore> -out <Path_To_Server.pem> -nodes

For Example:

openssl pkcs12 -in

/scratch/users/ofsaa/dev_home/config/server.keystore -out

/scratch/users/ofsaa/dev_home/config/server.pem -nodes

AAI User Provisioning SQL Scripts Generator Utility

This utility allows you to use AAI for authN in MMG. Identity administrators can create new user groups or roles, perform appropriate roles, usergroup and domain mapping, and so on.

This is provided as a SQL generator utility. This SQL scripts is executed in the AAI's configuration schema to create the required metadata.

Ensure that you run this script multiple times against each username. Additionally, generate the merge scripts accordingly.

Execute the following command from <mmg-home>/bin folder

./userprovisioning-script-generator.sh <user> <comma separated listof user groups or ALL> <infodom> <segment>

Sample Commands:

- ./userprovisioning-script-generator.sh SCRIPTUSER ALL OFSAAAIINFO EMFLD
- $./userprovisioning-script-generator.sh \ SCRIPTUSER \ MDLREV, MDLUSR, IDENTITY_ADMIN$

OFSAAAIINFO EMFLD

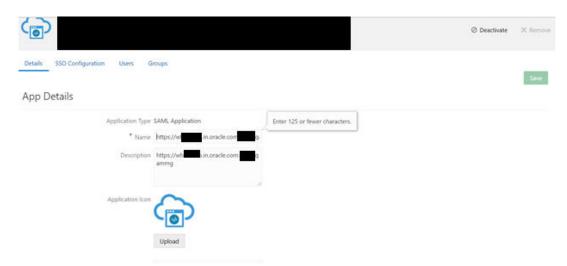
IDCS Server Configuration

To perform IDCS Server Configuration, follow these steps:

- Navigate to SAML IDCS Admin.
- 2. Navigate to Details section and add the app details in IDCS Server as shown below:



Figure 6-3 IDCS Server



Navigate to SSO Configuration section and add the app details in IDCS Server as shown below:

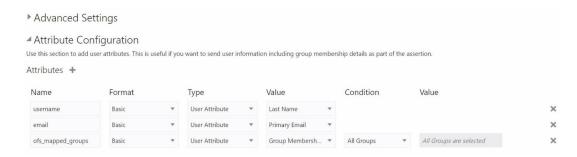


Figure 6-4 SSO Configuration section





Figure 6-5 IDCS Server





The following attributes such as username, email, and ofs_mapped_groups needs to configured as shown in the above image.

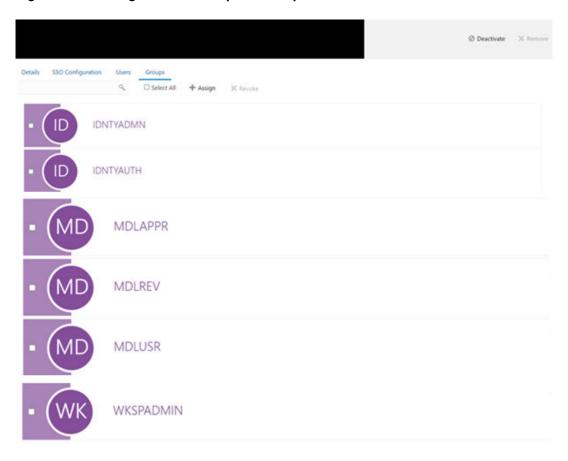
The user has to upload the certificate.

The user can generate the certificate by following these steps:

- Create .cer and .pem file from the below command.
- Upload the same .cer to the idcs Signing certificate option as well: openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout sp-privatekey.pem -out sp-certificate.cer
- 4. Navigate to Group section and Configure User Groups.



Figure 6-6 Configure User Groups in Group section



Conda Environment Migration and Restoration

Use this process when migrating or restoring Conda environments from a reliable source (such as production) to a sandbox instance

Source Environment (Export Steps)

Perform the following steps:

- 1. Prepare the backup directory:
 - Create a directory to store the exported environments by executing the following command:

mkdir conda_export

- Ensure that the mount point has sufficient storage.
- 2. Activate the environment.

Activate the environment you wish to export:

conda activate <conda_env_name>
or, for older conda:
source activate <conda_env_name>

3. Export the Environment.

Pack the active conda environment (ignore missing files):



conda pack -n <conda_env_name> -o <conda_env_name>.tar.gz --ignore-missingfiles



Do not use this method for custom environments without prior validation.

4. Deactivate the Environment.

Deactivate the environment after packing:

conda deactivate
or, for older conda:
source deactivate

- 5. **Pre-requisite** Install the conda-pack (if it is missing).
 - If the conda-pack is not installed, run: pip install conda-pack
 - Use index-url or extra-index-url if your repository requires them.

Target Environment (Import Steps):

Perform the following steps to transfer the environment archive:

- Transfer the environment archive.
 - Copy the exported .tar.gz file to the target server (For example, use scp or another file transfer tool).
- 2. Prepare the target directory.
 - a. Log in to the target server.
 - b. Navigate to the <<miniconda3>>/envs directory: cd <<miniconda3>>/envs
 - **c.** Create a directory for the environment to be imported:

```
mkdir <Env_to_be_imported>
```

For example:

 ${\tt mkdir ml4aml_8.1.2.6.0}$

- **d.** Rename the existing tar files if present, to avoid overwrites.
- 3. Untar the environment.

Extract the archive contents into the newly created environment directory:

```
tar -xzf <Env_to_be_imported>.tar.gz -C <Env_to_be_imported>
```

- 4. Test the python executable.
 - a. Before activating, you can test the environment's python:

```
./<Env_to_be_imported>/bin/python
```

- **b.** You must see the python prompt; exit with quit().
- Activate the environment.
 - Activate the imported environment: source <Env_to_be_imported>/bin/activate





(i) Note

Your prompt must now reflect the activated environment.

6. Verify the python version.

Confirm you are running python from within the activated environment:

python

Cleanup prefixes.

Fix any hardcoded prefixes in the environment:

```
conda-unpack
```

This step can be performed either within the activated environment or by specifying the path to the python binary.

Register the imported environment.

Add the environment path to your environment tracking file (optional but recommended for automation):

```
echo "<<miniconda3>>/envs/<Env_to_be_imported>" >> ~/.conda/Environment.txt
```

List the conda environments.

Confirm the new environment appears in your environment list:

```
conda env list
```

Enroll or use the environment as needed. Proceed with any application-specific enrollment steps.

Manual Configuration of Email Notification for Scheduler Batch Execution

This chapter provides information how to manually configure the SMTP settings.

Prerequisites

- Database access privileges to update the configuration schema.
- Knowledge of your SMTP server's host name and port.

To manually enable email notifications for OFS MMG, administrators must populate the SMTP configuration details into the AAICL SC COMPONENT DETAILS configuration table present in the OFS MMG configuration schema. This is required for environments running versions of OFS MMG released prior to the automated installer enhancement.

Execute the Insert Statements

Replace {{<smtphost>}} and {{<smtpport>}} with your actual SMTP server host name and port number.

```
INSERT INTO AAICL SC COMPONENT DETAILS
(V_COMPONENT_ID, V_PROPERTY_ID, V_PROPERTY_VALUE, V_CREATED_BY, D_CREATED_DATE,
V_MODIFIED_BY, D_MODIFIED_DATE)
VALUES
('AAICL EMAIL', 'host', '<smtphost>', NULL, NULL, NULL, NULL);
```



Validate the Configuration

Verify the entries by using a SELECT statement:

SELECT * FROM AAICL_SC_COMPONENT_DETAILS WHERE V_COMPONENT_ID = 'AAICL_EMAIL';

Fine Tuning Studio Sessions

Setting Timeouts

Data Studio user login sessions are logged out automatically after a set amount of time. It is recommended to set the appropriate session timeouts in the config.sh file.



(i) Note

Trigger the install.sh file with the -u option if any changes have been made in the config.sh file post installation.

Table 6-7 Timeout Settings

Timeout Type	Description	Example Configuration	Additional Remarks
Absolute Session Timeout	You can configure the user login sessions to be logged out automatically after a set amount of time. The number specifies the amount of milliseconds that are required to be passed after a session is created. When the timeout expires, the session is no longer valid. Here, the word login referes to when the user logs in to the Compliance Studio application. When a user session ends, all associated executions will also be killed.	studio-server: security: absolute-session- timeout-ms: 86400000 // 24 hours	Ends the session and kills the executions. It is recommended to set this. A value of 0 means that the sessions will never time out.



Table 6-7 (Cont.) Timeout Settings

Timeout Type	Description	Example Configuration	Additional Remarks
Relative Session Timeout	You can configure the user login sessions to be renewed as per the activity. The number specifies the amount of milliseconds a user session will be renewed to after an activity. When there is no user-activity within the specified relative timeout, the session expires and is no longer valid.	studio-server: security: session- timeout-ms: 3600000 // 1 hour	This is based on the user activity such as editing or executing. It is recommended to set this. A value of 0 means that the sessions will never be renewed.
	Paragraph Execution Timeout		
Execution Idle Timeout	The amount of time an interpreter session can stay idle before being invalidated by the application to free up resources. An interpreter session is considered idle when it is not running any tasks.	studio-server: interpreter: deployment: host: idle-session-timeout: 1h	Frees up memory and compute. It is recommended to set this.
	You can consider the interpreter session to be equivalent to the python sessions that gets created for a notebook execution. Here idle refers to paragraphs that are not being executed.		



Table 6-7 (Cont.) Timeout Settings

Timeout Type	Description	Example Configuration	Additional Remarks
Incomplete Job Cleanup	You can configure how often a cleanup job will be run to mark unfinished jobs as incomplete by setting studioserver.jobs.incomplete.s cheduled-cleanup-delay value. A job can be in incomplete state if the server restarts in the middle of a Job execution or if it takes longer the specified timeout (studioserver.jobs.incomplete.timeout).	studio-server: jobs: incomplete: timeout: 18h scheduled-cleanup- delay: 2h	Useful for crash recovery.

Upgrade Installation

This chapter describes the Upgrade Installation.

Upgrading to 8.1.3.3.0

To update an already installed MMG Application, perform the following steps:

Prerequisite:

- A valid working setup should be available before performing the upgrade.
- OFS MMG 8.1.2.7.0 must be installed before upgrading to OFS MMG 8.1.3.3.0.
- Use the MMG Config Schema, MMG Graph Schema, and MMG Datastudio Schema, from the existing version, along with the wallet configurations.
- Shutdown all the services of the existing installation using shutdown.sh.
- Backup the existing MMG Installation to a backup folder.

Upgrade:

Follow steps mentioned in the **Installation** section.



Compare and copy the placeholder values from the existing installed MMG.config.sh to the new one mentioned in the Installation section.

Update Utility to Reconfigure Installation Parameters

If you need to update any of the existing configuration related values, perform the following steps:



This Utility is applicable from 8.1.2.3.0 version onwards.

Procedure:

- 1. Shut down all the services using shutdown.sh command.
- 2. Reconfigure the config.sh file with the required changes.
- 3. Execute the command install.sh -u from the following path: <mmg installation path>/OFS MMG/bin

A successful update message as follows:

4. Start all the MMG services using startup.sh command.

Cloning the MMG Instance

There is a consistent requirement for a faster and effective approach of replicating an existing MMG Instance for further project developments. The approach is to set up the MMG Instances that are exact copies of the current MMG Instance.

Copying the Directories

The Installation Directory structure in the base environment has to be replicated in the clone environment.

 Copy the MMG base directory (OFS_MMG, by default) in the base environment with all of its contents to the clone environment.

The base directory in the clone environment will have the following folders upon copying:

- mmg-ui
- mmg-studio
- mmg-service
- mmg-schema-creator
- mmg-pipeline
- lib
- bin
- conf

Note

You need to copy LOG and FTPSHARE directories to the cloned environment.

Copying the Database Schemas

To copy the Database Schemas:

Create a copy each of the MMG Config Schema and the Data Studio Schema.
 You may use Oracle Data Pump Export/Import or the Database Copy feature of Oracle SQL Developer. For more details, see <u>Database Copy using Oracle SQL Developer</u>.

The Cloned Schemas can be created either in the same database instance or in a different one

Similarly, create copies of Workspace Schemas or other Data Source Schemas as required.

Configuring Password Store with Oracle Wallet

To configure the password store with Oracle Wallet:



Setup an Oracle wallet in the clone environment. For more details, see Setup Password Stores with Oracle Wallet.



Note

It is recommended to use the same wallet aliases used in the base environment.

Updating the WALLET_LOCATION and TNS_ADMIN_PATH

Update the WALLET LOCATION and TNS ADMIN PATH values in config.sh file present in the following path: OFS_MMG/bin with configured corresponding values of the cloned environment.

Updating the Host Details

Update the HOST and PORT values in config.sh file present in the following path: OFS_MMG/bin with configured corresponding values of the cloned environment.



(i) Note

It is recommended to use the same ports and context used in the base environment.

Replace the placeholders and update the host name in the MMG Config schema using the following command:

```
update NEXTGENEMF_CONFIG set V_VALUE =
'http(s)://##HOST NAME##:##BE PORT##/##CONTEXT##' where V NAME in ( '
BASE_URL', 'EMFSTUDIO_SERVICE_URL')
update NEXTGENEMF_CONFIG set V_VALUE =
'http(s)://##HOST NAME##:7008/##CONTEXT##' where V NAME = 'DATASTUDIO URL'
update AAICL_SS_BATCH_URL set V_URL =
'http(s)://##HOST_NAME##:##BE_PORT##/##CONTEXT##' whereV_URL_NAME in
('CS_SERVICE_URL', 'MMG_SERVICE_URL', 'WORKSPACE_URL')
```

Update LOG HOME and FTPSHARE

Update the LOG HOME and FTPSHARE values in config. sh file present in the following path:

OFS MMG/bin with configured corresponding values of the cloned environment.



Replace the ##LOG_HOME## and ##FTPSHARE## placeholders and update the LOG_HOME and FTPSHARE values in the MMG Config Schema using the following command:

```
update NEXTGENEMF_CONFIG set V_VALUE = '##LOG_HOME##' where V_NAME = 'LOG_HOME'
/
update NEXTGENEMF_CONFIG set V_VALUE = '##FTPSHARE##' where V_NAME = 'FTPSHARE'
```

Setting up the SSL Keystore

To run on HTTPS, you must create a Keystore for MMG Application. For more details, see the SSL Keystore in the Configure the config.sh file.

Update the Keystore path, Password and Storetype values in config.sh file present in the following path: OFS_MMG/bin with configured corresponding values of the cloned environment.

Updating Wallet Aliases for Oracle Schemas

(i) Note

It is recommended to use the same wallet aliases used in the base environment.

In case if the same wallet aliases cannot be used, perform the following:

- Update the MMG Config Schema Wallet Alias values in config.sh file present in the following path: OFS_MMG/bin with configured corresponding values of the cloned environment.
- Replace the placeholders and update the wallet alias for Workspace Schemas or other Oracle datasources using the following command:

```
update MMG_DB_MASTER set V_PROPERTY_VALUE = `##WALLET_ALIAS##' where
V_PROPERTY_NAME = `WALLET_ALIAS' and V_DB_NAME = `##DATASOURCE NAME##'
/
```

Updating Context and Ports

(i) Note

It is recommended to use the same context and ports used in the base environment.

In case if the same context and ports aliases cannot be used, perform the following:

- Update the references of context path and port values in config.sh file present in the following path: OFS_MMG/bin with configured corresponding values of the cloned environment.
- 2. Replace the ##CONTEXT## and ##BE_PORT## placeholders.



For more details, see **Updating the Host Details**.



Note

For MMG authentication, port access should be enabled for MMG ports to access OFSAA ports.

Starting MMG Services

Post updating all the required parameters in the new config.sh.file, start the services by using the following command: ./install.sh -u

Cloning the Environment

We will not have to manually update the entries or Database details/service URLs for most of the tables.

This can be done in two parts:

Part I

- Execute a utility that updates most of the entries in config tables and configuration/ properties in the file system.
- In the DR setup, after cloning, configure the <base>/mmg-home/bin/config.sh with new values of:
 - WALLET LOCATION (pointing to DR db)
 - TNS_ADMIN_PATH (pointing to DR db)
 - FTPSHARE (Ensure contents of ftpshare are also synced up between Prod and DR instances).



(i) Note

All other values, as applicable (If they differ between Prod and DR environments).

Execute the update script: <base>/mmg-home/bin/install.sh -u



(i) Note

This is not going to install again. Instead this will just update the configuration values in config tables and files.

Part II

For rest of the tables, additionally we can execute sql to update the following:

Replace the PARAMVALUE for PARAMNAME = ¿ER SERVICE URL;

fcc_pgx_m_config|V_GRAPH_PATH|: Replace this if it is not null; as this is applicable only for IN-Memory graphs.| |MMG_MENU|V_MENU_URL|For example:



Table 9-1 Param Value for Param Name with Service URL

V_MENU_CODE	V_MENU_NAME	V_MENU_URL
C003	Match Rules	https://ofss- mum-1750.snbomprshared1.gbuc dsint02bom.oraclevcn.com:7061/f cc/matchrulesetsummary.jsp
C004	Merge Rules	https://ofss- mum-1750.snbomprshared1.gbuc dsint02bom.oraclevcn.com:7061/f cc/mergerulesetsummary.jsp
C005	Data Survival	https://ofss- mum-1750.snbomprshared1.gbuc dsint02bom.oraclevcn.com:7061/f cc/datasurvivalsummary.jsp
C001	Manual Decisioning	https://ofss- mum-1750.snbomprshared1.gbuc dsint02bom.oraclevcn.com:7061/f cc/manualdecisioning.jsp
C006	Merge and Split Global Entities	https://ofss- mum-1750.snbomprshared1.gbuc dsint02bom.oraclevcn.com:7061/f cc/mergeandsplit.jsp

Frequently Asked Questions (FAQs) and Error Dictionary

This section consists of resolution to the Frequently Asked Questions and Error Codes noticed during the installation.

Topics:

Related Topics

- Frequently Asked Questions
- Frequently Asked Questions (FAQs) and Error Dictionary

Frequently Asked Questions

You can refer to the Frequently Asked Questions, which is developed with the interest to help you resolve some of the installation and configuration issues. This intends to share the knowledge of problem resolution to a few of the known issues. This is not an official support document and just attempts to share the knowledge of problem resolution to a few of the known issues.

Frequently Asked Questions

- 1. Why does my console show an unsuccessful message during wallet creation? Please check if you have run the following commands correctly. For more information on wallet creation, see Setup Password Stores with Oracle Wallet.
 - a. mkstore -wrl <wallet_location> -create //creates a wallet in the specified location.
 - b. mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the Studio Schema.
 - c. mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the Atomic Schema.
 - d. mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the configuration schema.

If your issue is still not resolved, contact My Oracle Support (MOS).

- 2. Where can I find my created wallet?
 Your wallet will be in the directory you have set as your wallet location.
 - If your issue is still not resolved, contact My Oracle Support (MOS).
- 3. When should I create a Database link, and if yes, how do I do it? Create a Database link to connect the Atomic and Configuration Database Schemas to the Studio Database Schema if the databases are different. You must create the link in the Studio Database.

In the following example, a link has been created from the Configuration Schema to the Atomic Schema by running the following script:



```
create public database link <studio database link> connect to <Config Schema>
identified by password using ' (DESCRIPTION = ADDRESS_LIST = (ADDRESS =
   (PROTOCOL = TCP) (HOST =<host name> (PORT = <port number>)) (CONNECT_DATA =
   (SERVICE_NAME = <service name>))) ';

Config Schema : <Config Schema>/password ' (DESCRIPTION = ADDRESS_LIST =
   (ADDRESS = (PROTOCOL = TCP) (HOST =<host name> (PORT = <port number>))
   (CONNECT_DATA = (SERVICE_NAME = <service name>))) ';
```

After running the script, run the FCDM Connector and ICIJ Connector jobs.

- 4. Why does my installed studio setup not have any notebooks? Some default notebooks are ready to use when you install Compliance Studio. If you do not see any notebooks when you log in to the application, you may not be assigned any roles. Check the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory to see if you have been assigned any roles, and if not, contact your Administrator. If your issue is still not resolved, contact My Oracle Support (MOS).
- 5. What can I do if the Schema Creation fails? If the Atomic Schema creation fails, login to the BD and ECM Atomic Schemas and run the following query: select * from fcc_orahive_datatypemapping; The fcc_orahive_datatypemapping table must not have duplicate data types. If the Studio schema creation fails, login as a Studio user and run the following query: select * from fcc_datastudio_schemaobjects Run the following query to replace all Y values with ": update fcc_datastudio_schemaobjects set SCHEMA_OBJ_GENERATED=" After the schema creation is successful, the value of the SCHEMA_OBJ_GENERATED attribute changes to Y. You can also check for errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory. If your issue is still not resolved, contact My Oracle Support (MOS).
- 6. What can I do if the Import_training_model batch execution fails?

 Batch Execution Status always displays success in case of success or failure.

You can also check for errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory. You can fix the failure according to the log details and run the same batch again.

- 7. Why is the sqoop job not successful? The Sqoop job may fail if some of the applicable values are null or if the service name or SID value is not provided. Do one of the following:
 - Check if there are any null values for the applicable configurations in the config.sh and FCC_DATASTUDIO_CONFIG tables. If there are any null values, add the required value.
 - Check for any errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory. If your issue is still not resolved, contact My Oracle Support (MOS).
- 8. Why am I getting the following error when I run the sqoop job:

```
Error: Could not find or load main class
com.oracle.ofss.fccm.studio.batchclient.client.BatchExecute
```

Set the FIC_DB_HOME path in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb directory.

You can also check for any errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory.

- 9. 11. Why is the PGX server is not starting even though the graph service is up and running? Grant execution rights to the PGX folder to start the PGX server.
- **10.** Why is the PGX Server not starting?



The PGX server starts only after the FCDM tables are created after the FCDM Connector Job is run. Check if all FCDM tables are created and then start the PGX Server. You can also check for any errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory. If your issue is still not resolved, contact My Oracle Support (MOS).

11. Why is the ICIJ Connector job failing?

This can happen because of a missing csv file path in the FCC_STUDIO_ETL_FILES table. Add the CSV file path. You can also check for any errors in the application log file in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs directory. If your issue is still not resolved, contact My Oracle Support (MOS).

12. What should I do if there is a below Error while selecting edges in manual Decision UI? java.lang.IllegalStateException: Unable to create PgxSessionWrapperjava.lang.IllegalStateException: Unable to create PgxSessionWrapper at oracle.datastudio.interpreter.pgx.CombinedPgxDriver.getOrCreateSession(Combine dPgxDriver.java:147) at oracle.pgx.graphviz.driver.PgxDriver.getGraph(PgxDriver.java:334) at oracle.pqx.qraphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer.java:22 3) at oracle.pgx.graphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer.java:20 9) at oracle.pgx.graphviz.library.QueryEnhancer.query(QueryEnhancer.java:150) at oracle.pqx.qraphviz.library.QueryEnhancer.execute(QueryEnhancer.java:136) at oracle.pgx.graphviz.interpreter.PgqlInterpreter.interpret(PgqlInterpreter.java :131) at oracle.datastudio.interpreter.pqx.PqxInterpreter.interpret(PqxInterpreter.java org.apache.zeppelin.interpreter.LazyOpenInterpreter.interpret(LazyOpenInterpre ter.java:103) at org.apache.zeppelin.interpreter.remote.RemoteInterpreterServer\$InterpretJob.jo bRun(RemoteInterpreterServer.java:632) at org.apache.zeppelin.scheduler.Job.run(Job.java:188) at org.apache.zeppelin.scheduler.FIFOScheduler\$1.run(FIFOScheduler.java:140) at java.util.concurrent.Executors\$RunnableAdapter.call(Executors.java:515) at java.base/java.util.concurrent.FutureTask.run(FutureTask.java:264) at java.base/ java.util.concurrent.ScheduledThreadPoolExecutor\$ScheduledFutureTask.run(Sched uledThreadPoolExecutor.java:304) at java.base/ java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1128) at java.base/ java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:628) at java.base/java.lang.Thread.run(Thread.java:834)Caused by: java.util.concurrent.ExecutionException: oracle.pgx.common.auth.AuthorizationException: PgxUser(FCCMDSADMIN) does not own session 6007f00a-8305-4576-9a56-9fa0f061586f or the session does not exist code: PGX-ERROR-CQAZPV67UM4H at java.base/ java.util.concurrent.CompletableFuture.reportGet(CompletableFuture.java:395) at java.base/ java.util.concurrent.CompletableFuture.get(CompletableFuture.java:1999) at oracle.pgx.api.PgxFuture.get(PgxFuture.java:99) at oracle.pgx.api.ServerInstance.getSession(ServerInstance.java:670) oracle.datastudio.interpreter.pgx.CombinedPgxDriver.getOrCreateSession(Combine

dPgxDriver.java:145) ... 17 moreCaused by:



```
oracle.pgx.common.auth.AuthorizationException: PgxUser(FCCMDSADMIN) does not
own session 6007f00a-8305-4576-9a56-9fa0f061586f or the session does not exist
code: PGX-ERROR-COAZPV67UM4H at
oracle.pgx.common.marshalers.ExceptionMarshaler.toUnserializedException(Except
ionMarshaler.java:107) at
oracle.pgx.common.marshalers.ExceptionMarshaler.unmarshal(ExceptionMarshaler.j
ava:123) at
oracle.pgx.client.RemoteUtils.parseExceptionalResponse(RemoteUtils.java:130)
oracle.pqx.client.HttpRequestExecutor.executeRequest(HttpRequestExecutor.java:
198) at
\verb|oracle.pgx.client.HttpRequestExecutor.get(HttpRequestExecutor.java:165)| at
oracle.pqx.client.RemoteControlImpl$10.request(RemoteControlImpl.java:313) at
oracle.pqx.client.RemoteControlImpl$ControlRequest.request(RemoteControlImpl.j
ava:119) at
oracle.pgx.client.RemoteControlImpl$ControlRequest.request(RemoteControlImpl.j
ava:110) at
oracle.pgx.client.AbstractAsyncRequest.execute(AbstractAsyncRequest.java:47)
at oracle.pgx.client.RemoteControlImpl.request(RemoteControlImpl.java:107) at
oracle.pgx.client.RemoteControlImpl.getSessionInfo(RemoteControlImpl.java:296)
oracle.pqx.api.ServerInstance.lambda$qetSessionInfoAsync$14(ServerInstance.jav
a:490) at java.base/
java.util.concurrent.CompletableFuture.uniComposeStage(CompletableFuture.java:
1106) at java.base/
java.util.concurrent.CompletableFuture.thenCompose(CompletableFuture.java:2235
) at oracle.pgx.api.PgxFuture.thenCompose(PgxFuture.java:158)
```

Then, perform the below steps as a workaround -

Export the "Manual Decision" Notebook

Add the link parameter just below Description

for Ex - "link": "manualDecision",

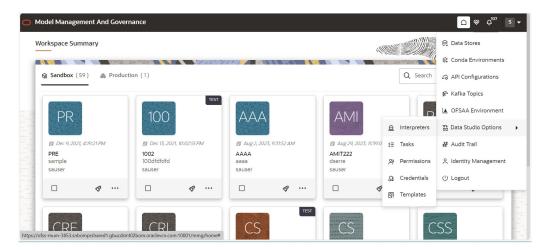
Figure 10-1 Manual Decision

```
[ {
    "name" : "manual Decision",
    "description" : null,
    "link": "manualDecision",
    "tags" : null,
    "version" : "5",
    "layout" : "zeppelin",
    "type" : "Default",
    "readOnly" : false,
```

Truncate the table "fcc_er_paragraph_manual" in Studio Schema. Import the modified notebook again.

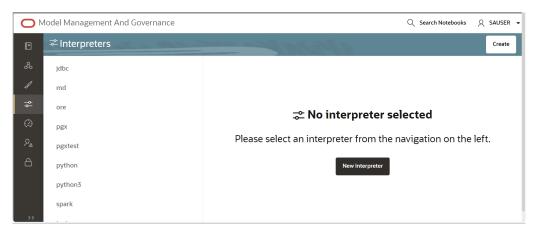
- 13. Data Extraction is truncated to default limit (Approx 197 records) in Python paragraph widget output in MMG. Setting the ZEPPELIN_LIMIT_INTERPETER_OUTPUT in Python Interpreter
 - From UI: Using Wizard screen
 Go to Interpreters screen in MMG-Studio from Datastudio Options tab.

Figure 10-2 Datastudio Options tab



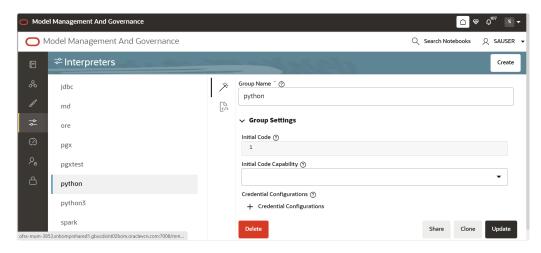
b. Once on the interpreters option screen select the Python Interpreter for which we want to configure the zeppelin.limit.interpreter.output.

Figure 10-3 Interpreter screen



Select python from the LHS options.

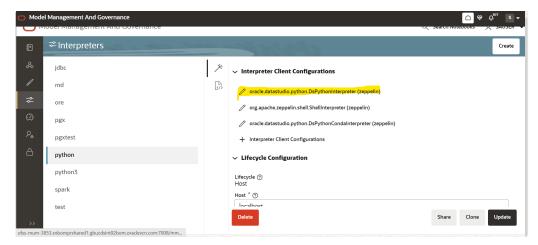
Figure 10-4 Python Interpreter





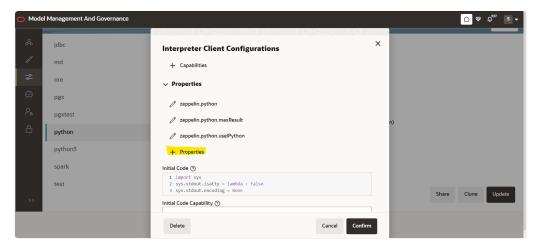
d. Now scroll down in the RHS side and click on the oracle.datastudio.python.DsPythonInterpreter under Interpreter Client Configurations it will open a popup.

Figure 10-5 Interpreter Client Configurations



e. In the popup scroll down and click on + Properties under Properties as shown:

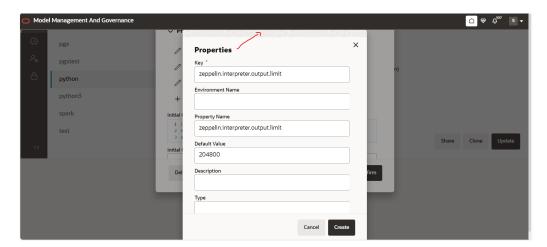
Figure 10-6 Properties screen



f. Another popup will open fill the options as shown and set the default value according to your needs if you are not able to see the Create and Cancel button, click on the part of the popup pointed by red arrow. The default value for zeppelin.interpreter.output.limit if not set is 102400 (in bytes).



Figure 10-7 Popup box

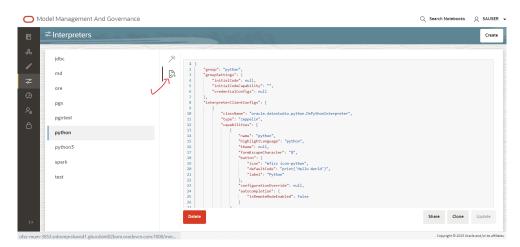


(i) Note

Increasing the default option from 102400 to some bigger value will slow down the rendering of outputs of python paragraphs.

- g. Once filled click on Create (you will see zeppelin.interpreter.output.limit under the Properties section), then click on **Confirm** (if you are not able to see the Confirm button on the UI, either click on the same shaded area on popup as highlighted in above image or zoom out in UI of browser) and then click on Update in the lower right side of the screen.
- h. After following all the above steps, restart the MMG-Studio for changes to reflect.
- a. Using JSON screen
 - i. Follow the steps i, ii and iii from above, then click on the following icon on UI pointed by red arrow and following json config view will open.

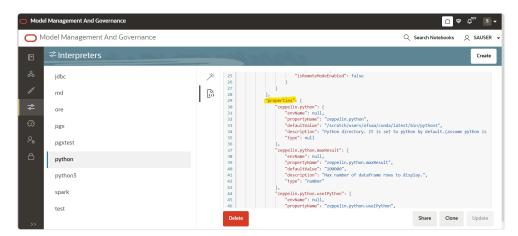
Figure 10-8 JSON Config View



ii. Scroll down under interpreterClientConfigs with className oracle.datastudio.python.DsPythonInterpreter you will find following properties section with bunch of zeppelin configurations.

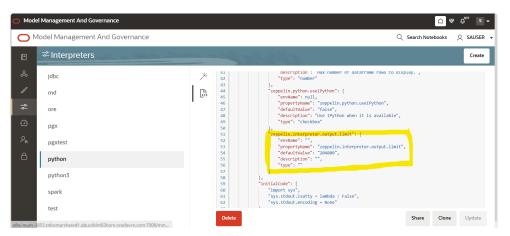


Figure 10-9 Interpreter Client Config



iii. After the last entry in properties add the zeppelin.interpreter.output.limit also as shown in the following image:

Figure 10-10 Properties screen



- iv. After doing the change the Update button will get enabled in the bottom right corner click on it, you will get a message as "python interpreter updated".
- v. Now restart the MMG-Studio service for changes to reflect.
- **b.** From filesystem: (Datastudio version 23.4.x onwards)
 - i. Go to the Python Interpreter option as pointed out in From UI using wizard screen option above, if you have already ran the MMG services before you will see the python interpreter listed there. Delete it, if you are running the MMG Application for the first time on a fresh schema then you do not need to do this step.
 - ii. After deleting the Python Interpreter or if start has not been done yet, go to filesystem inside mmg-home/mmg-studio/server/builtin/interpreters, open python.json in a text editor.
 - iii. Scroll down under interpreter ClientConfigs with className oracle.datastudio.python.DsPythonInterpreter you will find following properties section with bunch of Zeppelin configurations. After the last entry in properties add the Zeppelin Interpreter.output.limit also as shown in step iii) of 1) From UI b) using JSON screen (last image of From UI way). Save the python.json with the desired default value and the changes done.



iv. Now restart/start the MMG-Studio for your changes to reflect.



If you have configured the python environment for MMG-Studio (basically you have installed pandas and numpy which are subset of libraries required by MMG as pre-req), you can run the below script on python paragraph.

%python

import pandas as pd

import numpy as np

Create 1000 rows of random data for 20 columns

data = np.random.randn(1000, 20)

Create column names columns = [f"Column_{i+1}" for i in range(20)]

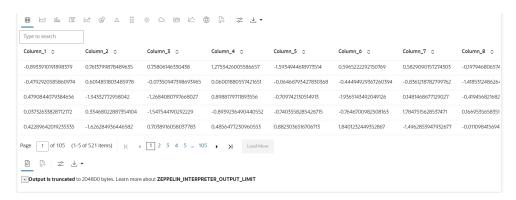
Create DataFrame df = pd.DataFrame(data, columns=columns)

Display the DataFrame

z.show(df)

Output in table view

Figure 10-11 Output in table view

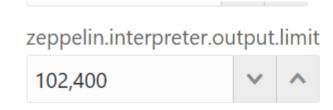


You can see the ZEPPELIN INTERPRETER OUTPUT LIMIT value as warning if the table content is more than the set default value for zeppelin.interpreter.output.limit and accordingly you can modify the default value for same.

- 14. What should I do when the result set is truncated if the size goes above '102400' bytes? Perform the following steps:
 - Login to Compliance Studio.
 - Navigate to interpreter zeppelin.interpreter.output.limit.



Figure 10-12 Zeppelin Interpreter



- c. Set the value to the required size.
- d. Restart the Studio Application.
- 15. What should I do if there is a below KubernetesClientException in load-to-elasticsearch.log, matching-service.log files after Compliance Studio installation? configServicePropertySourceLocator - Could not locate PropertySource: I/O error on GET request for "http://localhost:8888/<Service Name>/default": Connection refused (Connection refused); nested exception is java.net.ConnectException: Connection refused (Connection refused)onfigServicePropertySourceLocator - Could not locate PropertySource: I/O error on GET request for "http://localhost:8888/<Service Name>/default": Connection refused (Connection refused); nested exception is java.net.ConnectException: Connection refused (Connection refused) 20:04:55.686 [main] WARN .cloud.kubernetes.config.ConfigMapPropertySource - Can't read configMap with name: [<Service Name>] in namespace:[null]. Ignoring.io.fabric8.kubernetes.client.KubernetesClientException: Operation: [get] for kind: [ConfigMap] with name: [<Service Name>] in namespace: [null] failed. at io.fabric8.kubernetes.client.KubernetesClientException.launderThrowable(Kubern etesClientException.java:64) ~[kubernetes-client-4.4.1.jar!/:?] at io.fabric8.kubernetes.client.KubernetesClientException.launderThrowable(Kubern etesClientException.java:72) ~[kubernetes-client-4.4.1.jar!/:?] at io.fabric8.kubernetes.client.dsl.base.BaseOperation.getMandatory(BaseOperation .java:229) ~[kubernetes-client-4.4.1.jar!/:?] at io.fabric8.kubernetes.client.dsl.base.BaseOperation.get(BaseOperation.java:162) ~[kubernetes-client-4.4.1.jar!/:?] at org.springframework.cloud.kubernetes.config.ConfigMapPropertySource.getData(Co nfigMapPropertySource.java:96) ~[spring-cloud-kubernetesconfig-1.1.3.RELEASE.jar!/:1.1.3.

You can ignore the error when the following message is displayed at the end of the log; if you do not see this message, contact My Oracle Support (MOS) and provide the applicable error code and log:

13:52:57.698 [main] INFO org.apache.catalina.core.StandardService - Starting service [Tomcat] 13:52:57.699 [main] INFO org.apache.catalina.core.StandardEngine - Starting Servlet engine: [Apache Tomcat/9.0.43]

16. What happens if a new sandbox workspace is created? When a new sandbox workspace is created, the folders of the older workspace are by default being copied into the new workspace. Here, folder means the Model Objectives. The Model Objectives are global objects and will be visible across the workspaces. However, the models created within those objectives will be private. This has been done purposely as you expect multiple modelers working on the common objective in their private workspaces.



- 17. Not able to access any models in the copied folders in the new workspace the folders are being copied as empty folders?
 - Yes, you should not be able to access other workspace's private models. Also, as long as other users are working on the objective and have their models in there, you will not be able to delete the objectives.
- 18. What should you do when UI pages does not load due to less network speed?

 The default time to load all the modules of OJET/REDWOOD page is 1 minute. Reload the page to view the UI pages.
- **19.** What are the Workspace parameters used in MMG Python Scripts? The following parameters are used:
 - workspace.list_workspaces(): Used to fetch a list of all workspaces. This list is populated in the dropdown menu of datastudio.
 - workspace.check_aif(): A method used to check if AIF is enabled or not
 - workspace.attach_workspace("SANDBOX123"): A method used to set workspace
 - workspace.get_workspace(): Used to fetch the selected workspace (for example, SB1)
 - get_mmg_studio_service_url(): Used to fetch the base URL (for example, http://whf999yyy:0000/mmg)
 - **get_user():** Used to fetch current user (for example, mmguser)
- 20. How to take connections for Data access?

You need access to the data to work on it. For the workspace, there are some underlying Data Schemas. You can also create a workspace that allows to select multiple underlying Data Schemas. You can use or remove multiple Data Schemas like multi combo box, where 1, 2, 3, and 4, 5 are schemas underlying. When you work with the models, you can access the notebook to fetch data for all these Data Schemas and create some data frames out of it. That can be used for model reading or other purposes.

This happens in workspace of the sandbox where you are building a Notebook. The same Notebooks gets promoted to production workspace. Therefore, the workspace production has its own set of underlying Data Schemas. When you build the model with getting connection for the underlying Schema 1 and 2, and getting the data and building, it makes rules work and will not be affected if the same Notebooks gets promoted to production or deployment is cloned.

Therefore, the Notebook needs to run which should not be fetching this data because it will be working on any 1 and 2 Schemas.

To avoid this issue, you can use connection feature to connect with a schema. This is a wrapper function where you can specify which workspace you are connecting to.

You can enter the workspace details to get the connection and that starts fetching the data.

When you create the Notebook to production, a script runs to not to connect the workspace. This also uses overloaded methods. This method tells how to get the connection. Simple get connection gets the primary connection as first Data Schema which you are using without any overload.

The second connection gets an ID as the name the Data Source which you are using and for the current one will passes as get connection 1.

In the sandbox, this script looks for 1 and it creates a connection and moves to production.

It will again look for an equivalent 1 and tries to get a connection.

Therefore, whatever you select first, becomes the first Data Schema, Second Schema, Third Schema, therefore, Primary, Secondary, Tertiary and so on. You can also pass the



number while getting the connection to get the first primary Data Schema as a secondary Data Schema. Therefore, when it runs in sandbox, it gets the Secondary Schema. When it runs in the production, it fetches a Secondary Data Schema of production.

- 21. What are parameters to establish the Connection for data access? The following section lists the connection details such as the Data Sources and so on: workspace.get connection(): fetches connection object for the Primary Data Source of the workspace. This is equivalent to executing workspace.get connection(1). workspace.get connection('id'): fetches connection for the Data Source by name. For example, workspace.getconnection ('ws_data_1') - here 'ws_data_1' is one of the underlying Data Source for the workspace. workspace.get connection(n): fetches connection for the Data Source by order. For example, workspace.getconnection(2) - this will fetch connection for the Secondary Data Source. The following section lists the workspace details: After a workspace is attached, we can list Data Sources related to that using: workspace.list datasources(): will list Data Sources related to attached workspace with default order 1 For example, {'Data Source': [{'name': 'newdatasource1', 'order': '1'}]} workspace.list datasources("SB1"): will list Data Sources related to SB1 workspace with default order 1 For example, {'Data Source': [{'name': 'ds1', 'order': '1'}]} workspace.list datasources("SB1", 1): will list Data Sources related to SB1 workspace with order 1 as passed in second argument For example, {'Data Source': [{'name': 'ds1', 'order': '1'}}} Note: This is applicable for Python and Python variants interpreters, and not on any other interpreters.
- 22. What should I do if the Python installation displays the following error message, " If ModuleNotFoundError: No module named ' Izma"? You must install xz-devel library before installing the Python. For more details, see Install MMG Python Library section.

To install, perform the following step:

\$yum install -y xz-devel.

23. What should I do to reconfigure DS Studio server port and its interpreter's default port to available ports?

To reconfigure port numbers:

- a. Run the command install.sh -u to change the current studio port to the desired port number in the configuration files/tables.
- Run the t startup.sh script of Studio at the location: OFS MMG/mmg-studio/bin/ and modify the line numbers 24/25 of OFS-MMG/mmg-studio/bin/startup.sh to specify the interpreter name and port number.

DS version 22.4.3

nohup "\$DIR"/datastudio --jdbc -1 --eventjdbc -1 --shell -1 --eventshell -1 --graalvm -1 --eventgraalvm -1 --pgx -1 --eventpgx -1 --external --port 8008 --jdbc 3011 --eventjdbc 3031 --python 3012 --eventpython 3032 --markdown 3009 --eventmarkdown 3029 -spark 3014 --eventspark 3034 &> "\$DIR"/nohup.out &

For PGX Interpreter, modify: OFS_MMG/mmg-studio/interpreter-server/pgxinterpreter-22.4.3/bin/pgx-interpreter file "\${1:-7022}" "\${2:-7042}" values to "\${1:-3022}" "\${2:-3042}"

DS version 23.3.5

nohup "\$DIR"/datastudio --jdbc -1 --shell -1 --external --port 8008 --jdbc 3011 --python 3012 --markdown 3009 --spark 3014 --pgx 3022 &> "\$DIR"/nohup.out

For event ports in DS 23.3.5



Set the environment variables DS_EVENT_HANDLER_HOST and DS_EVENT_HANDLER_PORT before launching the interpreters, else, default values will be used. You can modify these ports in the startup.sh of the Studio.

Example:

export DS_EVENT_HANDLER_HOST=localhost

export DS EVENT HANDLER PORT=3432

To change the ports configured for events in the Data Studio server, modify the following server configuration:

studio-server:

thrift-server: enabled: true

port: <desired port -defaulted to 8432>

mode: TCP **NOTE:**

Python Interpreter

Beginning with Data Studio 21.4.0, 6012 is default port on which the REST server for the Python interpreter listens. To overwrite this, set the STUDIO_INTERPRETER_PYTHON_INTERPRETER_REST_SERVER_PORT environment variable.

PGX-Python Interpreter

Beginning with Data Studio 23.1.0, 6022 is the default port on which the REST server for the PGX-Python interpreter listens. To overwrite this, set the STUDIO_INTERPRETER_PGX_PYTHON_INTERPRETER_REST_SERVER_PORT environment variable.

Modify the startup.sh to:

export

STUDIO_INTERPRETER_PYTHON_INTERPRETER_REST_SERVER_PORT=3038

export

STUDIO_INTERPRETER_PGX_PYTHON_INTERPRETER_REST_SERVER_PORT= 3039

This configuration changes the default interpreter ports to new ports.

- **c.** Ports mentioned in the interpreter json files should be reconfigured. The interpreter file location is: "OFS MMG/mmg-studio/server/builtin/interpreters/<interpreter>.json" file.
- Execute startup.sh and check the studio/interpreter ports.
- e. Similarly, execute ./datastudio.sh –help from OFS_MMG/mmg-studio/bin/ for all available options.

DS Studio Server port and its interpreters default port can be reconfigured to any available ports by following these steps:

- a. Change the Datastudio URL with the desired DS port. install.sh -u must be triggered to change the current studio port to 8008 in the configuration files/tables.
- b. After successful execution of install.sh. The ports can be updated by the user in the startup.sh of studio in the path OFS_MMG/mmg-studio/bin/.



Edit line no 24/25 of OFS-MMG/mmg-studio/bin/startup.sh and change as below by specifying the interpreter name and port to be modified.

In DS version 22.4.3

nohup "\$DIR"/datastudio --jdbc -1 --eventjdbc -1 --shell -1 --eventshell -1 -graalvm -1 --eventgraalvm -1 --pgx -1 --eventpgx -1 --external --port 8008 -jdbc 3011 --eventjdbc 3031 --python 3012 --eventpython 3032 --markdown 3009 --eventmarkdown 3029 --spark 3014 --eventspark 3034 &> "\$DIR"/ nohup.out

For PGX Interpreter

Change it directly in the OFS MMG/mmg-studio/interpreter-server/pgxinterpreter-22.4.3/bin/pgx-interpreter file "\${1:-7022}" "\${2:-7042}" values to "\${1:-3022}" "\${2:-3042}"

In DS version 23.3.5

nohup "\$DIR"/datastudio --jdbc -1 --shell -1 --external --port 8008 --jdbc 3011 --python 3012 --markdown 3009 --spark 3014 --pgx 3022 &> "\$DIR"/nohup.out & For event ports in DS 23.3.5 You need to set the environment variables DS EVENT HANDLER HOST and DS EVENT HANDLER PORT before interpreters are launched. Otherwise, the default values are used. This can be mentioned in the startup.sh of studio. example:

- export DS EVENT HANDLER HOST=localhost
- export DS EVENT HANDLER PORT=3432

In order to change the port listening for events in the Data Studio server, adapt following server configuration:

- studio-server:
 - thrift-server:
 - enabled: true
 - port: <desired port -defaulted to 8432>
 - mode: TCP

Generic Notes

Python Interpreter

Starting from Data Studio 21.4.0, the REST server for the Python interpreter listens on port 6012 by default. One can overwrite this by setting the STUDIO INTERPRETER PYTHON INTERPRETER REST SERVER POR T environment variable.

PGX-Python Interpreter

Starting from Data Studio 23.1.0, the REST server for the PGX-Python interpreter listens on port 6022 by default. One can overwrite this by setting

STUDIO INTERPRETER PGX_PYTHON_INTERPRETER_REST_SERVER_ PORT environment variable.

The above can be mentioned in the startup. sh of studio as export STUDIO INTERPRETER PYTHON INTERPRETER REST SERVER POR T=3038 export

STUDIO_INTERPRETER_PGX_PYTHON_INTERPRETER_REST_SERVER_ PORT=3039



The above configuration will change the default interpreter ports and reconfigure to listen to the new ports. (For example: As mentioned in the below table).

- Ports mentioned in the interpreter json files also needs to be reconfigured. The interpreter file locations can be found at "OFS MMG/mmg-studio/server/builtin/ interpreters/<interpreter>.json" file.
- iii. Execute startup.sh and check the studio/interpreter ports.
- iv. Similarly you can execute as ./datastudio.sh -help from OFS MMG/mmgstudio/bin/ location for all the available options.

The above steps will reconfigure Server/Interpreter to these ports:

- Server/Interpreters Modified Port
- DS Studio port 8008
- Jdbc 3011
- eventidbc 3031
- python 3012
- eventpython 3032
- markdown 3009
- eventmarkdown 3029
- spark 3014
- eventspark 3034
- pgx 3022
- eventpgx 3042
- 24. Dataset issue with the latest version of pydantic package (2.18.7) Pydantic package (2.18.7) is incompatibile with MMG functionality. So, when you reinstall the package the uninstall and reinstall pydantic package version 1.10.13.
 - python3 -m pip uninstall pydantic
 - python3 -m pip install pydantic==1.10.13 --user
- 25. Installation of Python Packages from Local Repository In order to install the python dependencies in mmg-8.1.2.6.0.tar.gz from a local repository, use the following command.python3 -m pip install mmg-8.1.2.6.0.tar.gz.tar.gz --index-url http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple --extra-index-url http:// artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple --trusted-host artifactory.XYZ.com
- 26. MMG Configuration steps for Interpreters
 - For JDBC

Update the below proeprty in jdbc.json underOFS MMG/mmg-studio/server/builtin/ interpreters

"propertyName": "default.url"

"defaultValue": "<JDBC URL>"

For example: jdbc:oracle:thin:@ofss-

mum-1033.snbomprshared1.gbucdsint02bom.oraclevcn.com:15 21/MMG19PDB

"propertyName": "default.user",



"defaultValue": "<schameusername>"

For example: The schema user to which you want to connect, for example: datastudio schema name.

"propertyName": "default.password",

"defaultValue": "<schemapassword>"

For example: Password of the provided schema user.

Start the jdbc interpreter by executing below command under //OFS_MMG/mmg-studio/interpreter-server/jdbc-interpreter-22.4.3/bin ./

jdbc-interpreter

If the jdbc interpreter needs to be included in the datastudio startup script remove the below entry from /OFS MMG/mmg-studio/bin/startup.sh --jdbc -1.

For Spark

Configuration with Kerberos enabled remote spark cluster:

- a. Copy the configured Spark directory from hadoop cluster to <MMG Studio>/interpreter-server/spark-interpreter/extralibs. For example: spark-2.4.8-bin-hadoop2.7
- **b.** Copy the below files to the <MMG Studio>/interpreter-server/spark-interpreter/extralibs krb5.conf <keytabfile>.keytab
- c. To run Spark in yarn-client mode, configure the following parameters in this file OFS_MMG/mmg-studio/server/builtin/interpreters/spark.json spark.master = yarn-client

spark.driver.bindAddress = 0.0.0.0

spark.driver.host = <host> -> Apache Spark host

(i) Note

When using the Kubernetes interpreter lifecycle, <host> can be the IP address or hostname of any node in your Kubernetes cluster. When using the Host interpreter lifecycle, <host> should be the IP address or hostname of the node that runs the Spark interpreter.

(i) Note

When connecting to a YARN cluster, the Spark driver authenticates as the UNIX user that runs the Spark interpreter. You can set the HADOOP_USER_NAME environment variable to make the Spark driver authenticate as a different user. If you use the Host interpreter lifecycle, then you can do this by exporting the HADOOP_USER_NAME environment variable before starting the Spark interpreter process. If you us the Kubernetes interpreter lifecycle, then you can do this by setting the HADOOP_USER_NAME environment variable in the resource manifest (spark.yml).

- Update file spark-defaults.conf keytab location to the location where <keytabfile>.keytab file is copied
- **e.** Update file spark-env.sh with the krb5.conf location to the location where krb5.conf file is copied.



For example: Djava.security.krb5.conf=/OFS_MMG/mmg-studio/interpreter-server/spark-interpreter-22.4.2/extralibs/krb5.conf".

- 27. MMG SAAS OFS_SRV_ACCT USER is seeded in the User Profile tables: OFS_SRV_ACCT user is available in the following tables as a seeded user:
 - aaicl_sms_user_profile
 - aaicl_sms_user_profile_u

Application Pack 8.1.3.3.0 FAQs

- 1. The the cx_Oracle connection is failing in DS with the following error in OEL 8. Fail to execute line 4: cx_Oracle.connect(dsn=dsn_alias)\nTraceback (most recent call last):\n File "/tmp/1638454321889-0/zeppelin_python.py", line 163, in <module>\n exec(code, _zcUserQueryNameSpace)\n File "<stdin>", line 4, in <module>\ncx_Oracle.DatabaseError: DPI-1047: Cannot locate a 64-bit Oracle Client library: "libnsl.so.1: cannot open shared object file: No such file or directory". Install the libnsl package as below: yum install libnsl or sudo yum install libnsl
- 2. Python Interpreter fails with the 'py4j' Error

When running interpreters locally, they assume all the dependencies to be already installed and available. Python Interpreter needs 'py4j' Package, exact steps to install it depend on the Operating System. If you use 'pip', it can be done with ''bash pip install --user py4j ''Install the package for all users, root user can run this command without '--user'.

3. What is the reason for the http error code 401 when I successfully log in to the MMG Application while MMG Studio is down?

If MMG Studio is not up during the MMG application login, the mmg-ui logs capture the http error code: 401. Since the cookie creation is done during MMG application login, the user must re login to the MMG application once the Studio is up and running.

4. What should I do when the following error message is displayed, and the SSL module is unavailable for Linux 8?

urllib3.exceptions.SSLError: Can't connect to HTTPS URL because the SSL module is not available. During handling of the above expectation, another exception occurred:

- a. Install the compat-openssl10 module on Linux 8.
- b. Log in to the server as a root user where MMG Application is installed.
- c. Run the following Shell command: yum -y install compat-openssl10.
- 5. Why do multiple commits/versions appear in my remote Git repository after my first push, even though I only pushed once?

When you push a local git repository to a remote one for the first time, all the commits that exist in your local repository's history for the branch that is being pushed are transferred to the remote repository. This happens even if the remote is new or empty. This is standard git behavior. If your local repository was initialized earlier, or was linked to a different remote connection, then all its' commits will still remain in your local git history. When you add a new remote connection and perform a push action, git pushes the entire commit history, not just the most recent change. So the remote connection will reflect all your local commits.

Key Points:

- Git tracks the full commit history locally, regardless of remote connections.
- Pushing to a new remote connection brings the entire history of your pushed branch.
- If you want only specific commits in the new remote connection, you must create a new repository or filter your history before pushing.