# **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 8.1.2.7.0

G22774-04

Copyright © 2024, 2024, 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	
Deployment Topology	1
Components of Oracle Financial Services Model Management Governance	2
Installation Check List	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	2
Create MMG Graph Schema	3
Create the Installation, Download, and Metadata Repository Directories	3
Configure the OS File System Settings and Environment Settings in the .profile File	4
Configure Operating System and File System Settings	4
Configure the Environment Settings	5
Java Settings	6
Oracle Database Server and Client Settings	6
TNS entries in the tnsnames.ora file for Non-TCPS	6
Time Zone Settings	7
Setup Password Stores with Oracle Wallet	7
Setup the Password Stores for Database User Accounts	7

# 5 Installation

6

Prerequisites	1
Download the OFS MMG Installer Kit	2
Extract the Software	2
Configure the config.sh file	3
Import Server Certificate to Java Keystore	26
Run the MMG Installer	27
Starting MMG Services	28
Stopping MMG Services	29
Generate GRAPH-KEYSTORE.P12	30
Installing Python Library	30
Prerequisites	30
Procedure	31
Setting up the Environment for Hive Data Sourcing	33
Remote MMG Studio Configuration	34
PGX Installation	35
Configure the config.sh File of PGX	36
Starting PGX Server	38
Stopping PGX Server	38
R Interpreter	38
ORD-3.6.1 Installation	38
R 4.1.2 Installation	39
Configuring R Interpreter	39
MMG Connection Objects Library Setup	40
Installing ROracle Library	40
Installing RODBC Library	41
Using MMG Studio to Oracle Connection Objects	42
Workspaces	42
Connections	42
Installing Conda	43
Multi Level Approval	43
Interpreter Configuration	44
Network Configuration for Load Balancer	45
Post-installation Steps	
Access the Application	1
Create Application Users	1
Map Application User(s) to User Group	1

	Model Techniques/ Model Library	3
	.PEM file creation for Model Service	4
	Access and Permissioning Management	4
	Access MMG Using AAI Realm	5
	Prerequisites	5
	Access MMG using SAMLRealm	6
	AAI User Provisioning SQL Scripts Generator Utility	6
	IDCS Server Configuration	7
	Conda Environment Migration and Restoration	9
	Manual Configuration of Email Notification for Scheduler Batch Execution	11
7	Upgrade Installation	
	Upgrading to 8.1.2.6.0	1
9	Update Utility to Reconfigure Installation Parameters  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
10	Frequently Asked Questions (FAQs) and Error Dictionary	
	Frequently Asked Questions	1
	Frequently Asked Questions	1
	Application Pack 8.1.2.0.0 FAQs	17

# **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

### (i) Note

In the Tech Matrix, the Java Version column is no longer applicable for release 8.1.2.7.0. Instead, Java"11.0.13" 2021-10-19 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**

- Deployment Topology
- Components of Oracle Financial Services Model Management Governance

# **Deployment Topology**

Web Tier

Application Tier

HTTP/ILS

Interpreters
Spark
(IDP)
Python
Python
Web Tier

Application Tier

HTTP/ILS

Interpreters
Spark
Spark
Spark
INTERPRES

Oracle DB

MMG Schema

JUBE/Oracle
Wallet Connection

ML Built-in Models

Cracle wallet connection

Figure 2-1 The logical architecture implemented for OFS MMG Application Pack



# Components of Oracle Financial Services Model Management Governance

The following are the components of Oracle Financial Services Model Management 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 OFS Model Management and Governance User Guide.

# **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	
1	Install all the prerequisite <i>hardware and software</i> as per the OFS Analytical Applications 8.1.2.0.0 Technology Matrix.
2	Configure the Database Instance Settings.
3	<u>Create</u> the Installation, Download, and Metadata Repository Directories:
	<ul> <li>Installation Directory</li> </ul>
	<ul> <li>Temporary Directory</li> </ul>
	<ul> <li>Staging Area/Metadata Repository</li> </ul>
	<ul> <li>Download Directory</li> </ul>



Table 2-1 (Cont.) Installation Checklist

SI. No.	Activity
4	Configure the following Operating System and File System Settings:
	File Descriptor
	<ul> <li>Total number of processes</li> </ul>
	<ul><li>Port(s)</li></ul>
	<ul> <li>.profile file permissions</li> </ul>
	<ul> <li>Add FTP or SFTP Configuration for file transfer (to access Staging Area and Metadata Directory)</li> </ul>
5	<u>Update</u> the following Environment Settings as required for the installation in the .profile file:
	<ul> <li>Java Settings</li> <li>Oracle Database Server and Client Settings</li> </ul>
	<ul> <li>Add TNS entries in the TNSNAMES.ORA file</li> </ul>
	<ul> <li>Time Zone Settings</li> </ul>
Installation Steps	
6	Download the Installer Kit.
7	Extract the Installer Kit.
8	Configure the config.sh file.
9	Trigger the Application Installation.
Post-Installation Steps	
10	Access the MMG Application.
11	Create Application Users.
12	Map Application User(s) to User Groups.

# Hardware and Software Requirements

See the <u>Oracle Financial Services Analytical Applications 8.1.2.0.0 Technology Matrix</u> document for the hardware and software requirements.

# **License Information**

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

# Preinstallation

# **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:

- Metadata Schema
- Data Schema
- MMG Application Schema
- 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

Ensure that you create an Oracle user to create the application configuration schema. For more details, see Creating an Oracle User section.

### **Assign the Grants**

This section discusses the various grants required for the Oracle Database User.

### Assign the following grants:

```
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 <oracle_database_user>;
grant create TRIGGER to <oracle_database_user>;
grant create VIEW to <oracle_database_user>;
grant create MATERIALIZED VIEW to <oracle_database_user>;
grant select on SYS.V_$PARAMETER to <oracle_database_user>;
grant create SYNONYM to coracle_database_user>;
grant select on sys.v_$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>;
```

### (i) Note

This is required for the configuration schema and the workspace schema.

# Create the MMG Studio Schema

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

### Assign the following grants:

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



# 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.

### 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

# **Parameter Configuration Action** File Descriptor Settings In the sysctl.conf file, to change the number of file descriptors, do the following as the root user: Edit the following line in the /etc/sysctl.conf fs.file-max = <value> where <value> is greater than 15000 Apply the change by running the following command: # /sbin/ sysctl -p (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.



### (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.

# 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:

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

# Oracle Database Server and Client Settings

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

Table 4-3 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 4-4 TNS entries in the TNSNAMES.ORA file for Non-TCPS

Description	Example Value
Ensure that an entry (with SID or SERVICE NAME) is added in the tnsnames.ora file on the OFSAA server.	<pre> <sid_name> = DESCRIPTION =(ADDRESS_LIST</sid_name></pre>
<sid name=""> =</sid>	
(DESCRIPTION =	
(ADDRESS_LIST =	



# 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 MMG User Guide.

Table 4-5 Time Zone Settings

Description	Example Value
Time Zone	TZ=Asia/Calcutta

# 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 the <u>Oracle Database Security Guide</u>.

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.

A password store for the application and Application Server User Accounts must also be installed. However, the installer takes care of this entire process.

# 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.
- 2. Create a wallet in the <wallet\_location> using the following command: mkstore -wrl <wallet\_location> -create

### 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:
Enter wallet password:
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:
Enter wallet password:
Enter your secret/Password:
Enter wallet password:
Enter wallet password:
Enter wallet password:
Enter wallet password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password:
Enter wallet password:
Enter wallet password is missing in the command line
Enter your secret/Password is missing in the command line
Enter your secret/Password:
Enter wallet password:
```

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



mkstore -wrl <wallet\_location> -createCredential <alias-name> <mmg-</pre> schema-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.



### (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.

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 this think the above details or create new trsnames.ora file and add the required entries.
- <alias-name> is a user-defined value.
- 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:



### 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:
```

To verify if you are connected to the correct user:

```
SQL> show user
The output is similar to:
USER is "<database-user-name>"
```

Oracle Database 12c

# Installation

This section provides detailed steps to install the application.

# **Prerequisites**

Below is the list of prerequisites:

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

 For Solaris Operating System, the MMG Studio has to be configured in Linux machine remotely. And the studio URL must be the same as that of the remote studio during MMG Application installation. To configure Remote MMG Studio, see the <u>Remote MMG Studio</u> <u>Configuration</u>.

Two new flags are introduced in configuration table:

- ENB CSTM GRP if this is set, then custom groups can be created.
- ENB\_GRP\_SYNC if this is set, then unmapping of users and groups are enabled.

Based on these flags we can enable unmapping of users with groups on user login.

User needs to execute the following scripts manually in Config Schema:

- Merge into configuration:
  - Using (SELECT 1 FROM DUAL)
  - On (a.PARAMNAME= 'ENB\_CSTM\_GRP'). When not matched, insert: (a.PARAMNAME, a.PARAMVALUE, a.DESCRIPTION) values ('ENB\_CSTM\_GRP', 'true', 'Enable custom group creation during JIT').
- Merge into configuration:
  - Using (SELECT 1 FROM DUAL)
  - On (a.PARAMNAME= 'ENB\_GRP\_SYNC'). When not matched, insert: (a.PARAMNAME, a.PARAMVALUE, a.DESCRIPTION) values ('ENB\_GRP\_SYNC', 'true', 'Enable unmapping operation during JIT').



# Download the OFS MMG Installer Kit

To download the software as a .zip folder, download the mandatory minor release patch **36885021** 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 8.1.2.7.0 installer archive file in the download directory with the following command:

```
unzip OFS_MMG_8.1.2.7.0_<OS>.zip
```

After unzipping the OFS\_MMG\_8.1.2.7.0 folder, following zip folders are displayed under OFS MMG folder.

- mmg-installer.zip
- mmg-metadata-manager.zip
- mmg-pgx.zip
- OFSMMG\_8.1.2.7.0\_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
- 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 MMG, follow these steps:

- 1. Login to the server as a non-root user.
- 2. Navigate to the <installation directory>/OFS\_MMG/bin directory.
- 3. Configure the applicable config.sh attributes as shown here:

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

```
#!/bin/sh ##
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 FTPSHARE=##FTPSHARE##
##By default, Data Studio is assumed to run on the same server. If it's
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##
export UI PORT=##
UI_PORT## export SCHEMA_PORT=##SCHEMA_PORT##
export CONTEXT PATH=##CONTEXT PATH##
export STUDIO_AUTH_TYPE=##STUDIO_AUTH_TYPE##
export SSO_TOKEN=##SSO_TOKEN##
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=10
export MMG_DATASOURCE_IDLE_TIMEOUT=30000
export MMG_DATASOURCE_CONN_TIMEOUT=80000
export EXT_DATASOURCE_MAX_POOL_SIZE=10
export EXT_DATASOURCE_IDLE_TIMEOUT=30000
export EXT_DATASOURCE_CONN_TIMEOUT=80000
export MMG_HTTP_MAX_CONN=20
export MMG_HTTP_MAX_CONN_PER_ROUTE=2
export MMG_HTTP_CONNECT_TIMEOUT=30000
export MMG_HTTP_READ_TIMEOUT=120000
export APPLICATION_ID=##APPLICATION_ID##
```

### ## Properties for MMG-ui

```
export APPLICATION FAVICON PATH=##APPLICATION FAVICON PATH##
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##
```

# ## Properties for enabling AAI Authorization. Applicable for SAML/LDAP profiles

## This is disabled by default. Set AAI\_AUTHZ\_ENABLED to true to enable.
## AAI\_AUTH\_URL must be set.

export AAI\_AUTHZ\_ENABLED=##AAI\_AUTHZ\_ENABLED##

export AAI\_CLIENT\_ID=##AAI\_CLIENT\_ID##

export AAI\_CLIENT\_SECRET=##AAI\_CLIENT\_SECRET##



- 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 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##
- # 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##
DATAPIPELINE METADATA ARCHIVE PATH=##DATAPIPELINE METADATA ARCHIVE PATH##
DATAPIPELINE_METADATA_IMPORT_SERVICE_PORT=##DATAPIPELINE_METADATA_IMPORT_SERVI
CE 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##
export GRAPH AUDIT DATASOURCE NAME=##GRAPH AUDIT DATASOURCE NAME##
## 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 GRAPH_SERVICE_CACHE_SERVER_PORT=##GRAPH_SERVICE_CACHE_SERVER_PORT##
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_V
IEW##
## 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##
```

Oracle Financial Services Model Management and Governance Installation and Configuration Guide

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##

Page 7 of 45



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\_EVICTABLE\_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\_INT ERPRETER\_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\_ENABLED=##MMG\_GATEWAY\_ENABLED##

export MMG\_GATEWAY\_PORT=##MMG\_GATEWAY\_PORT##

## If Gateway is enabled, 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).

 $\sharp$  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##

#ENDOFFILE#

Table 5-1 config.sh file

Parameter	Description	Is Mandatory	Comments
Common Properties			
##APPLICATION_NAM E##	Title of the application; if not replaced, default is: Model Management and Governance.	YES	It defaults to "Model Management and Governance."
			NOTE: Provide double quotes for the application name if it is long or contains spaces.
##WALLET_LOCATION ##	The wallet is the folder containing 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 configured for the MMG application config schema.	YES	MMG_CONFIG
##LOG_HOME##	A writable folder designated for storing application and MMG Studio logs.	YES	/scratch/users/logs <b>Note</b> : Ensure that log folder is created before installation.
##FTPSHARE##	This can be any writable folder accessible to the process owner.	YES	/scratch/users/ftpshare
			Ensure that ftpshare folder is created before installation.
			This should be same as the metadata directory mentioned above.



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##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.	YES	HostIP or FQDN
	Use the same hostname wherever applicable.		
##BE_PORT##	Port on which the backend service (mmg-service) needs to run.	YES	7002
##UI_PORT##	Port on which UI service (mmg-ui) needs to run.	YES	7001
##SCHEMA_PORT##	Port on which Schema Creator service needs to run.	YES	7003
##CONTEXT_PATH##	Context path of the application.	YES	mmg
##STUDIO_AUTH_TYP E##	<ul> <li>FCC_SSO – for SAMLRealm based authentication in FCC Studio</li> <li>MMG_AAI- AAI based authentication for MMG Studio</li> </ul>	YES	Can be either MMG_AAI or FCC_SSO



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##SSO_TOKEN##	SSO Token value for Studio authentication.	YES	Note: SSO token value needs to be
	Applicable only when STUDIO_AUTH_TYPE is FCC_SSO and MMG_AAI.	ne	regenerated whenever new keys are generated.
	For FCC_SSO, refer to the <u>Oracle Financial</u> <u>Services Compliance</u> <u>Studio Installation</u> <u>Guide</u> .		
	For MMG_AAI, to create the SSO Token, follow these steps.		
	a. Locate <mmg_installa tion_path="">/  OFS_MMG/bin /  key-generator.sh  and execute it.</mmg_installa>		
	<ul> <li>After successful execution, public key and private key are generated at the following paths</li> </ul>		
	<mmg_installation _PATH&gt;/OFS_MMG/ conf</mmg_installation 	ı	
	and		
	<mmg_installation _PATH&gt;/OFS_MMG/ mmg-studio/conf/</mmg_installation 	I	
	Execute the token- generator.sh file by passing <api_user> as an argument value. This file is located at the following path: <mmg_installation _PATH&gt;/OFS_MMG/bir</mmg_installation </api_user>		
	Example: ./token- generator.sh MMG_API_USER		
	a. After successful execution, a file named token.out is created inside the same bin directory. The token.out file contains the following content:		
	"Generated JWT Token for MMG_API_USER:	ı	



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
	<some-long-random- token-value&gt;"</some-long-random- 		
	<ul> <li>Copy the <some- long-random- token-value&gt; part mentioned above and paste it into ##SSO_TOKEN## in the config.sh file. Ensure that there is no space or end line at either the start or end while copying this value into ##SSO_TOKEN##.</some- </li> </ul>		
##SSL_ENABLED##	This enables HTTPS.	YES	Example: true
##SSL_KEYSTORE##	Absolute path for the keystore file. <b>Note</b> : Run the following	YES	/conf/server.keystore. Include the file name in the path.
	command to create a		NOTE:
	keystore: keytool -genkey -v - alias demoalias -keyalg RSA -keysize 2048 - keystore server.keystore -validity 3650 -keypass secret - storepass secret - storetype PKCS12		If ##SSL_ENABLED## is set to false, you must configure keystore for mmg-studio, as it is SSL-enabled 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	YES	Example: secret
	The value passed in the aforementioned command for -keypass		
##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	<b>NOTE:</b> If not applicable, enter NA
##FCC_API_USER##	API user for FCC Studio.	YES	<b>NOTE</b> : If not applicable, enter NA



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##MMG_DATASOURC E_MAX_POOL_SIZE##	Maximum connection pool size allowed for Config Datasource.	YES	It defaults to 10. You can edit it if required.
##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.
##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	It defaults to 20.
## MMG_HTTP_MAX_CO NN_PER_ROUTE##	The maximum number of HTTP connections allowed for a route.	YES	It defaults to 2.
##MMG_HTTP_CONN ECT_TIMEOUT##	The connection timeout for HTTP connection. A timeout value of 0 specifies an infinite timeout.	YES	It defaults to 30000.
##MMG_HTTP_READ_ TIMEOUT##	The socket read timeout for HTTP connection. A timeout value of 0 specifies an infinite timeout.	YES	It defaults to 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.
mmg-ui			
##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



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##UI_AUTH_TYPE##	aai – if using an existing AAI instance as the identity provider. saml – for saml based authentication ldap – 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.  Refer Import Server Certificate to Java Keystore for more details.	YES	http(s):// whfxxxxx.in.oracle.com: 7110/mmg
AUTH_SAML_REQUE ST_TYPE	This can be set to REDIRECT (default) / POST.		
AUTH_SAML_INCLUD E_SP_CERT	This can be set to True/ False (default).		
##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/sso 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.	YES	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>



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##SAML_LOGOUT_UR L##	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.
##LDAP_URL##	LDAP URL	YES	ldap://whf00xyz:3060/
	Will be used for ##UI_AUTH_TYPE## = LDAP		
##LDAP_SEARCH_BA		YES	"cn=Users,dc=oracle,dc
SE##	Will be used for ##UI_AUTH_TYPE## = LDAP		=com"
##LDAP_USER_FILTE		YES	"cn={0}"
R##	Will be used for ##UI_AUTH_TYPE## = LDAP		
#LDAP_USER_SEARC H_FILTER##	LDAP User Search Filter	YES	
	Will be used for ##UI_AUTH_TYPE## = LDAP		
##LDAP_GROUP_SEA RCH_FILTER##	LDAP Group Search Filter	YES	
	Will be used for		
	##UI_AUTH_TYPE## = LDAP		
##LDAP_GROUP_SEA RCH_BASE##	LDAP Group Search Base	YES	
	Will be used for ##UI_AUTH_TYPE## = LDAP		
##LDAP_GROUP_ME MBER#	LDAP Group Member	YES	
WIDER#	Will be used for ##UI_AUTH_TYPE## = LDAP		
##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 999999



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##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	YES	If not set, it defaults to true.
	false		
Properties for mmg-serv	rice		
##BE_AUTH_TYPE##	Auth Type on which the backend service (mmg-service) runs.	YES	It defaults to public.
##MMG_PYTHON_INT ERPRETER##	value without whitespaces that specifies python interpreter	YES	If not set, it defaults to python.
""DATA CATALOG OFD	python,fcc-ml4aml	NO	
##DATACATALOG_SER VICE_URL##	application is integrated with MMG.	NO	
Properties for mmg-stud	io		
##STUDIO_WALLET_E NABLED##	Set as true when using a wallet for the MMG Studio Schema.	YES	true/TRUE
	Can be either true/ TRUE or false/FALSE (all caps or all small)		
##LOGIN_SHOW##	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



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##OFSAA_URL##	AAI login IDM Service URL. This is applicable only if ##	YES	Format: http:// <ofsa- web-host&gt;:<port>/ <context>/rest-api</context></port></ofsa- 
	STUDIO_AUTH_TYPE ## is " MMG_AAI ".		For example, http:// ABC00abc:4325/LLFP/ rest-api
			The /rest-api is mandatory for OFSAA URL.
##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
	<b>Note</b> : 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	
	When ##WALLET_ENABLED ## is false-		
	jdbc:oracle:thin:@ <hos t&gt;:<port>/ <service_name></service_name></port></hos 		
	When ##WALLET_ENABLED ## is true-		
	jdbc:oracle:thin:@ <ds ALIAS&gt;</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



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##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
##PYTHON_HOME##	Home Path of Python Library. It defaults to python3 during installation. For a custom	YES	python3
	installation of python3 where the soft link is not configured, you can mention the complete path up to python3.		
##SPARK_HOME##	Absolute path of Apache Spark Library.	NO	
##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.	YES	The default is False.
	Note:		
	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
fcc services specific con	figurations		



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##TEMPLATE_CONFI G_PATH##	Configuration path of the Template.	NO	
##TEMPLATE_DEFAU LT_LINK##	Default link of the template.	NO	
##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>
##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.	YES	Example: group
	The attribute will contain the role required for the application.		
##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 specific	configurations		



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##DATAPIPELINE_SER VICE_PORT1##	The port where the pipeline service resides.	YES	By default, it is set as 18005.
##DATAPIPELINE_SER VICE_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.
##DATA PIPELINE_METADATA _IMPORT_SERVICE_P ORT##	Meta data import service port.	YES	By default, it is set as 18007.
##DATAPIPELINE_ERX MLPATH##	The XML path in which the schema details are stored.	NO	
##DATAPIPELINE_GAT EWAY_SERVICE_POR T##		YES	
##PIPELINE_UI_SERV ICE_PORT##	Pipeline UI service port.	YES	
##DATA_PIPELINE_UI _SERVICE_PORT##	Data pipeline UI service port	YES	
URLS for pipeline,ER an that could be added to a	~	e as it is if not applicable.	Will impact the pipeline
##MATCHRULE_BASE _URL##	The host and port where the match rule service resides.	NO	http(s):// abc.in.xyz.com:7051
##LOADGRAPH_BASE _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	
URLs for index service. I to a graph	_eave as it is if not applica	able. Will impact the pipel	ine that could be added
##GRAPH_INDEX_BA SE_URL##	Indicates the Graph Index resource path.	NO	http(s):// <hostname>xyz.com:70 53/load-to-elastic- search</hostname>
##LOADINDEX_UI_UR L##	Indicates the Graph Load Index UI resource path.	NO	



Table 5-1 (Cont.) config.sh file

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, 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.  NOTE: This is applicable for Compliance Studio.	NO	
##CANDIDATE_SELEC TION_SERVICE_URL# #		NO	
##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	
Changes for auth service	es + mmg keys		
##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
Properties to package Lo	oad to Graph (L2G) services ##	ce inside MMG	
##GRAPH_INSTALLATI ON_PATH##	The installation path of the Graph.	YES	<mmg installation<br="">Path&gt;/ OFS_MMG/ mmg-load-to-graph/ graph-service</mmg>



Table 5-1 (Cont.) config.sh file

raph Keystore assword. raph service port.	YES YES	Password  By default, it is set as
raph 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.
ns		
ame of the MMG atabase Server.	YES	
ne port of the MMG atabase server.	YES	
ame of the MMG atabase Service.	YES	
dicates the pgx server source path.	YES	http(s):// <hostname>.xyz.com:&lt;</hostname>
ote: Refer to the <u>PGX</u> stallation section for ore details.		pgx port>/ <pgx context<br="">name&gt;</pgx>
kip this if not installing gx.		
onfigurations		
aximum total shared ata memory size.	YES	Edit if required; default value is 20 GB.
aximum total private ata memory size.	YES	Edit if required; default value is 8 GB.
aximum per session ata memory size.	YES	Edit if required; default value is 700 MB.
its		
aximum data memory ze allowed for SUSRGRP.	YES	Edit if required; default value is 10 GB.
aximum data memory ze allowed for SBATCH.	YES	Edit if required; default value is 10 GB.
aximum data memory ze allowed for SINTER.	YES	Edit if required; default value is 5 GB.
ze allowed for	YES	Edit if required; default value is 5 GB.
· · · · · · · - · · · · · · · · ·		
	eximum total private ta memory size.  eximum per session ta memory size.  eximum data memory size allowed for SUSRGRP.  eximum data memory size allowed for SBATCH.  eximum data memory size allowed for SBATCH.  eximum data memory size allowed for SINTER.  eximum data memory size allowed for SINTER.	eximum total private YES sta memory size.  Eximum per session YES sta memory size.  Eximum data memory YES se allowed for SUSRGRP. Eximum data memory YES se allowed for SBATCH. Eximum data memory YES se allowed for SINTER. Eximum data memory YES se allowed for SINTER. Eximum data memory YES se allowed for SINTER. Eximum data memory YES se allowed for

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



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##GRAPH_SCHEMA_ WALLET_ALIAS##	Wallet alias created for the Graph Schema.	YES	
##GRAPH_SCHEMA_ DB_SCHEMA_NAME# #	Name of the Graph schema.	YES	

### **Additional MMG Features**

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

##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	
##EST_SERVICE_URL ##	YES	<pre><host_name>:<est _service_server_="" port="">/ <est_server_serv let_context_path=""></est_server_serv></est></host_name></pre>	

### **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.

##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##	•	NO	The default values are 7011, 7031.
##DATASTUDIO_PYTH ON_INTERPRETER_R EST_SERVER_PORT# #	Studio Python	NO	The default value is 6012
##DATASTUDIO_PGX_ PYTHON_INTERPRET ER_REST_SERVER_P ORT##	Studio PGX Python	NO	The default value is 6022



Table 5-1 (Cont.) config.sh file

Parameter	Description	Is Mandatory	Comments
##DATASTUDIO_THRI FT_EVENT_HANDLER _PORT##	•	NO	The default value is 8432
##DATASTUDIO_PGX_ INTERPRETER_PORT ##	•	NO	The default value is 7022.
##MMG_COHERENCE _CLUSTER_PORT##	This is the port where the Coherence Cluster for the Notification service runs. By default, this value is 7574.	YES	
##MMG Gateway Config	uration		
##MMG_GATEWAY_E NABLED##	Enables or disables the MMG Gateway. By default, this is set to 'self'.	YES	
##MMG_GATEWAY_P ORT##	Specifies the port on which the MMG Gateway will run.	YES	

## If Gateway is enabled, 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 commaseparated list of origins to allow embedding from those specified origins and from the same origin. # By default, this is set to 'self'.

##MMG\_CSP\_FRAME Configure this attribute YES
\_ANCESTORS## 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.

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



Table 5-1 (Cont.) config.sh file

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 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.	YES	
RESTRICT_UNMAPPE D_DATASTORES_ACC ESS		NA	The default value is False.



### Note

- In case of ##OFSAA\_URL## and ##MMG\_SVC\_URL##, don't add any ending '/' in the URLs
- If pool size, connection timeout and idle timeout are not configured, then it will proceed with default Hikari Configurations.
- The default session timeout is 3600 seconds (60 mins). You can configure timeout using 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 minimally to be present for a successful subsequent logins.
- The name for MMG Studio cookie is ORA\_OLDS\_SESSION.
- If the ##SSL\_ENABLED## is set to false, 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 same for all the MMG services including MMG Studio. So, if you
  want to use the MMG Studio with wallet configurations, then configure in the
  same wallet.
- If the 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 MMG application is Non SSL, set the below property to "false" in the application.yml file inside the MMG Studio and restart the services. security:

cookies:
secure: false

## 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"





### (i) Note

The above step should be performed by the Root user.

### Run the MMG Installer

To run the MMG Installer, follow these steps:

Navigate to following path:

Go to <MMG\_INSTALLATION\_PATH>/bin directory.

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=jdbc: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.

# 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:

### (i) Note

The Keystore generation fails if graph service is down.

- Execute graph-keystore-generator.sh using PUTTY.
- 2. 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 Python Library**

This section provides detailed steps to install the Python Library.

### Prerequisites

### Python 3.9.x



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

### For Example:

- Installation
- Install MMG Python Library
- Model Management and Governance



### (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 sqlite-devel
- ncurses-devel: Install as a root user using the command yum install ncursesdevel
- 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 libffi-devel

### 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

### Procedure

- 1. Set system python3 to the one that is to be used. Navigate to bin folder.
- 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.



3. 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.

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

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

### (i) 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 all the

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 lib 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##
```

### (i) 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:

 Import MMG studio server certificate to MMG application server java keystore and vice versa.

For more details, see Import Server Certificate to Java Keystore section.

### (i) Note

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:

- 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-23.4.6
- 3. Give 0755 permission to mmg-pgx folder.
- 4. Configure the config.sh of pgx. For more details, see Configure the config.sh File of PGX section.
- 5. Copy the graph-keystore.p12 from MMG Installation server to <pgx installation path>/mmg-pgx/conf. For more details, see <a href="Generate GRAPH-KEYSTORE.P12">Generate GRAPH-KEYSTORE.P12</a> 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
- 7. 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 <a href="Configure the config.sh File of PGX">Configure the config.sh File of PGX</a> 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.
- 2. 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.



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

Parameter	Description	Is Mandatory	Comments
##PGX_SSL_ENABLE D##	The values can be true /false.  If true, follow the below steps if Self Signed is being used:  Import PGX server.cer file to	YES	
	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 t	rue.	
##PGX_SSL_KEYSTO RE##		YES	/conf/server.keystore. Include the file name in
	This is applicable only if ##PGX_SSL_ENABLE D## is set to true.	the	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.		
##PGX_SSL_KS_ALIA S##	The Alias of the PGX keystore.  This is applicable only if ##PGX_SSL_ENABLE D## is set to true.	YES	password123
Properties for graph service			



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

Parameter	Description	Is Mandatory	Comments
##	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 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.

### ORD-3.6.1 Installation

To install ORD-3.6.1, follow the steps mentioned in the below guides:

- https://www.oracle.com/database/technologies/r-distribution.html
- https://docs.oracle.com/en/database/oracle/machine-learning/oml4r/1.5.1/oread/installingoracle-R-distribution-on-linux.html#GUID-A73BA0EB-507C-4678-9AD7-CE2CB6CE0251



- Check installation:
  - a. R-version
- 2. Installing other packages:

Set proxy:

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

### R 4.1.2 Installation



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

To install R 4.1.2, follow these steps:

- 1. Set Proxy, (pseudo user):
  - a. curl- O https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
  - b. yum install epel-release-latest-7.noarch.rpm
  - c. curl- O https://cdn.rstudio.com/r/centos-7/pkgs/R-\${R VERSION}-1-1.x86 64.rpm
  - d. sudo yum install R-\${R\_VERSION}-1-1.x86\_64.rpm
  - e. sudo In -s /opt/R/\${R VERSION}/bin/R /usr/bin/R
- 2. Check installation:
  - a. R-version
- Installing other packages:
  - a. R-e "install.packages('Rserve', repos='https://www.rforge.net/')"
  - b. R-e "install.packages(c('knitr', 'ggplot2', 'backports'), repos='https://mirror.las.iastate.edu/CRAN/')"

### 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.





### Steps to enable TLS during R Interpreter Installation

Steps to enable TLS during R Interpreter Installation:

- Download Rserve\_1.8-10.tar.gz from this link: <a href="https://cran.r-project.org/src/contrib/">https://cran.r-project.org/src/contrib/</a>
   Archive/Rserve/
- 2. Install openssl-devel [yum install openssl-devel]
- 3. 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> --no-save")

### MMG Connection Objects Library Setup

This section describes the MMG Connection Objects Library Setup.

### Installing ROracle Library

### **Prerequisites**

DBI is one of the dependencies for using this library.

- Installing DBI
  - 1. curl- O <a href="https://cran.r-project.org/src/contrib/DBI\_1.1.1.tar.gz">https://cran.r-project.org/src/contrib/DBI\_1.1.1.tar.gz</a>
  - 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
  - 1. curl- O <a href="https://cran.r-project.org/src/contrib/ROracle">https://cran.r-project.org/src/contrib/ROracle</a> 1.3-1.1.tar.gz
  - 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 <code>\$ORACLE\_HOME/bin path</code>.

### 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 '/usr/lib64/R/library' or similar root directory for system installation.

### For R 4.1.2

- 1. curl- O https://cran.r-project.org/src/contrib/RODBC 1.3-19.tar.gz
- 2. R CMD INSTALL RODBC 1.3-19.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)



### Using MMG Studio to Oracle Connection Objects

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

### Workspaces

- 1. mmg.list\_workspaces(): Used to fetch a vector of all workspaces.
  For example: vec <- mmg.list\_workspaces() vec will be vector object</pre>
- 2. 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:

mmg.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

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");
- 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");
- 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



# **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.

# 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
```

2. 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
```

# **Interpreter Configuration**

#### For JDBC

"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:

- Copy the configured Spark directory from hadoop cluster to <MMG Studio>/interpreterserver/spark-interpreter/extralibs. For example: spark-2.4.8-bin-hadoop2.7
- 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

### 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).

- 4. Update file spark-defaults.conf keytab location to the location where <keytabfile>.keytab file is copied
- 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".

# Network Configuration for Load Balancer

This section provides information on the network configurations for the load balancer.

The \*load balancer IP\* and \*application hostnames\* must be added to the {{/etc/hosts}} file on all relevant servers.

# Post-installation Steps

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

# Access the Application

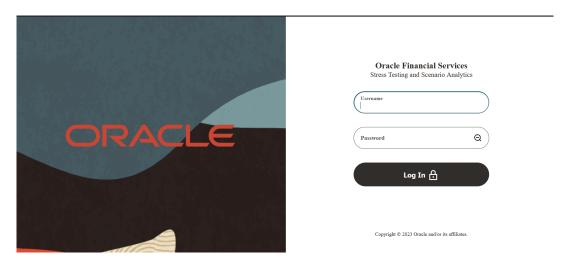
To access the application, follow these steps:

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

http(s)://est service host name: <MMG\_GATEWAY\_PORT>/<context\_path>/home
For example:

https://xyz.com:4155/mmgservice/

Figure 6-1 Login window – AAI Authentication



For more information, see the <u>User Access and Permissioning Management</u> 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  $\Delta AI$ :

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.

### (i) Note

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

- **IDNTYADMN**
- **IDNTYAUTH**

# 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.



### Note

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
```

# **Access and Permissioning Management**

STSA 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 STSA Application is accessed using the following realms that you have selected during the installation of the STSA 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 STSA applications 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
  STSA Applications using the Permission feature.



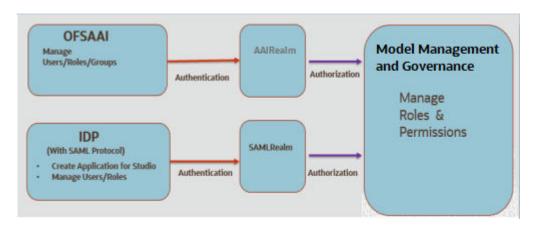


Figure 6-2 Authentication and Authorization process in STSA

### Access MMG Using AAI Realm

This section provides information on creating users who can access MMG using the AAIRealm 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. 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>.
- If MMG is SSL enabled, then the SSL certificate for MMG application should be imported in AAI.
- In OFSAA Application, Allow user to log in from multiple machines option should be enabled.



# 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



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

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.



It is recommended to use AAIRealm or SAMLRealm.

# 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

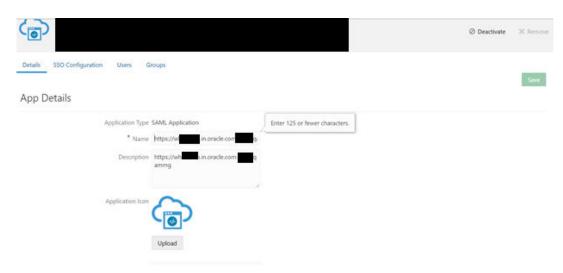
OFSAAAIINFO EMFLD

# **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:

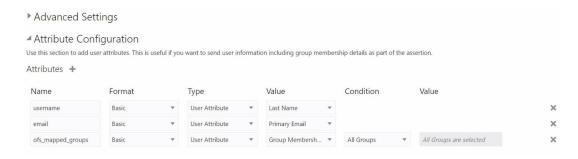


Figure 6-4 SSO Configuration section





Figure 6-5 IDCS Server

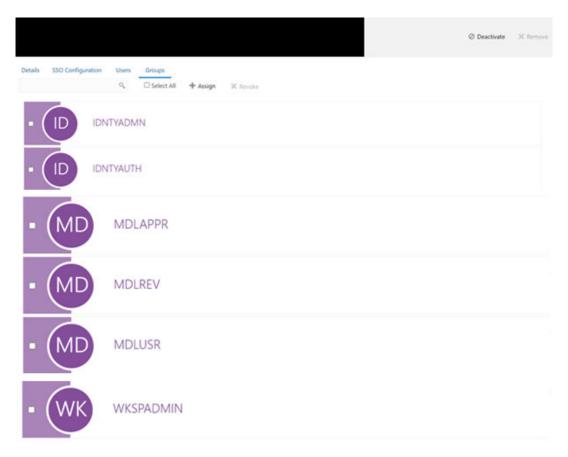




The following attributes such as username, email, and ofs\_mapped\_groups needs to configure as shown in the above image.

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:

- 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-missing-
files
```



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:

mkdir ml4aml\_8.1.2.6.0

- d. Rename the existing tar files if present, to avoid overwrites.
- 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>
```

- 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
```



#### Note

Your prompt must now reflect the activated environment.

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
```

10. 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 in the {{AAICL\_SC\_COMPONENT\_DETAILS}}} table to enable email notifications for the scheduler batch



execution in OFS MMG. These instructions apply to environments running versions released prior to the automatic installer enhancement.

#### **Prerequisites**

- Database access privileges to update the configuration schema.
- Knowledge of your SMTP server's hostname 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 that are running versions of OFS MMG that were released prior to the automated installer enhancement.

#### **Execute the Insert Statements**

Replace  $\{\{<smtphost>\}\}$  and  $\{\{<smtpport>\}\}$  with your actual SMTP server hostname 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);
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', 'port', '<smtpport>', NULL, NULL, NULL, NULL);
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);
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', 'port', '<smtpport>', NULL, NULL, NULL, NULL);
```

## Validate Configuration

Verify the entries using a SELECT statement:

```
SELECT * FROM AAICL_SC_COMPONENT_DETAILS WHERE V_COMPONENT_ID = 'AAICL_EMAIL';
```

# **Upgrade Installation**

This chapter describes the Upgrade Installation.

# Upgrading to 8.1.2.6.0

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

### Prerequisite:

- A valid working setup should be available before performing the upgrade.
- Use the MMG Config and MMG Datastudio Schema from the existing version along with the wallet configurations.
- Create a new Graph Schema. For more details, see Create the Graph Schema section.



If upgrading the MMG Application from 8.1.2.3.0 and above versions, skip the above step.

- 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.

(i) Note

Compare and copy the placeholder values from the existing installed MMG.config.sh to the new MMG.config.sh 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**



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:

- 1. 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**

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**.



## (i) 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

# 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 <a href="Setup Password Stores with Oracle Wallet">Setup Password Stores with Oracle Wallet</a>.
  - **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.

- 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

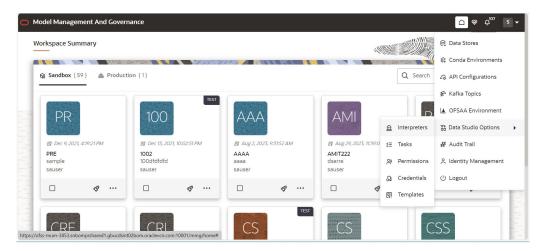
```
[ {
    "name" : "manual Decision",
    "description" : null,
    "link": "manualDecision",
    "tags" : null,
    "version" : "5",
    "layout" : "zeppelin",
    "type" : "Default",
    "readOnly" : false,
```

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
  - a. From UI: Using Wizard screen
     Go to Interpreters screen in MMG-Studio from Datastudio Options tab.

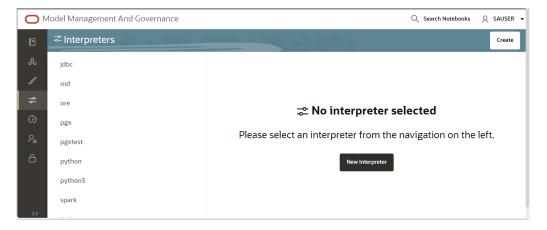


Figure 10-2 Datastudio Options tab



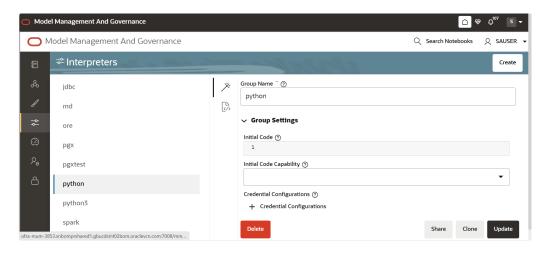
**b.** Once on the interpreters option screen select the Python Interpreter for which we want to configure the zeppelin.limit.interpreter.output.

Figure 10-3 Interpreter screen



Select python from the LHS options.

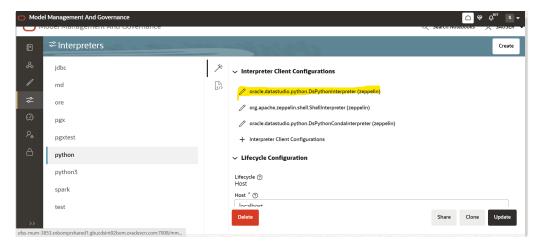
Figure 10-4 Python Interpreter





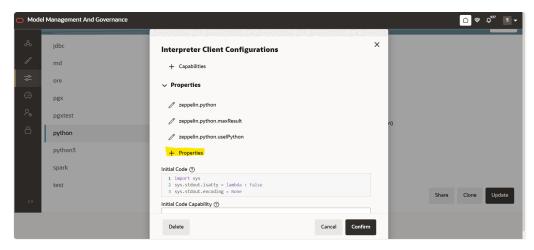
d. Now scroll down in the RHS side and click on the oracle.datastudio.python.DsPythonInterpreter under Interpreter Client Configurations it will open a popup.

Figure 10-5 Interpreter Client Configurations



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

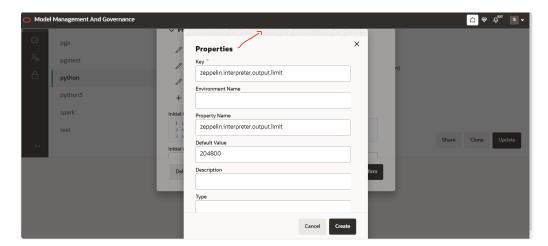
Figure 10-6 Properties screen



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

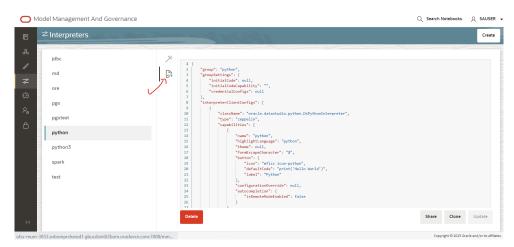


### (i) Note

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

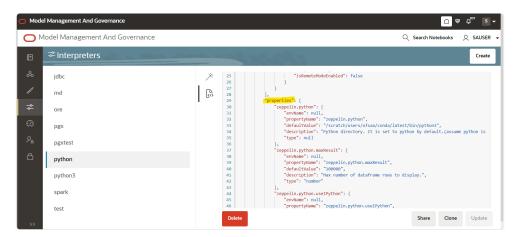
- g. Once filled click on Create (you will see zeppelin.interpreter.output.limit under the Properties section), then click on **Confirm** (if you are not able to see the Confirm button on the UI, either click on the same shaded area on popup as highlighted in above image or zoom out in UI of browser) 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



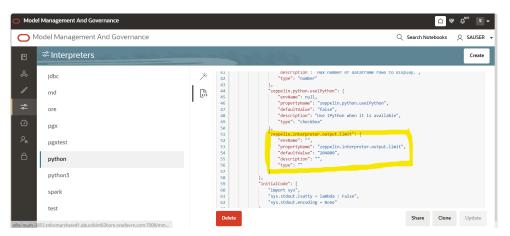
ii. Scroll down under interpreterClientConfigs with className oracle.datastudio.python.DsPythonInterpreter you will find following properties section with bunch of zeppelin configurations.

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)]

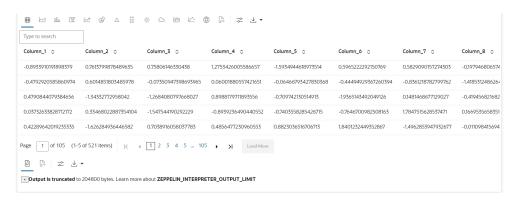
# Create DataFrame df = pd.DataFrame(data, columns=columns)

# 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 <u>My Oracle Support (MOS)</u> 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]
```

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 '\_lzma'"? You must install xz-devel library before installing the Python. For more details, see <u>Install MMG Python Library</u> 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.
- b. 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/pgx-interpreter-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/.



i. 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/pgx-interpreter-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 the

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).

- ii. 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 <a href="http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple">http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple</a> --extra-index-url <a href="http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple">http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple</a> --trusted-host <a href="http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple">http://artifactory.XYZ.com/artifactory/api/pypi/XYZ-py-local/simple</a> --trusted-host
- 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:

- Copy the configured Spark directory from hadoop cluster to <MMG Studio>/interpreterserver/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
- 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

### 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.

### 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
- 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".



# Application Pack 8.1.2.0.0 FAQs

- 1. If the cx\_Oracle connection is failing in DS with below 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. Incase of Python Interpreter fails With `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.