

# **Oracle Financial Services Crime and Compliance Studio**

**Installation Guide**

**Release 8.0.8.3.0**

**August 2022**

**E91246-01**

**ORACLE**  
Financial Services

## OFS Crime and Compliance Studio Kubernetes Installation Guide

Copyright © 2022 Oracle and/or its affiliates. All rights reserved.

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 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 installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon 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, Opteron, the AMD logo, and the AMD Opteron 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.

For information on third party licenses, click [here](#).

# Document Control

Version Number	Revision Date	Change Log
1.4	August 2022	<p>Updated with note information for CDH in the following sections:</p> <ul style="list-style-type: none"> <li>• <a href="#">Hardware and Software Requirements</a> (Big Data)</li> <li>• <a href="#">Download the Big Data Files</a> (Additional Jars)</li> <li>• <a href="#">Appendix C – Additional Jars – PGX</a></li> <li>• <a href="#">Appendix D – Additional Jars – Batch Service</a></li> </ul> <p>Updated with correct reference topics in <a href="#">Configure the Extract Transfer and Load (ETL) Process</a> section.</p> <p>Updated SQL statements in <a href="#">Cleanup for Upgrade</a> section.</p>
1.3	May 2021	<p>Updated the following sections:</p> <ul style="list-style-type: none"> <li>• New parameters added to the <a href="#">Configure the studio-env.yml File</a> for Elastic Search.</li> <li>• <a href="#">Configuring Infer and Incremental Training API</a></li> <li>• New parameters added for PGX Service to the config.sh file in the <a href="#">Configure the Extract Transfer Load (ETL) Services</a>.</li> <li>• New section <a href="#">Create Credential Keystore for Elastic Search</a> is added.</li> <li>• New section <a href="#">Generate an Encrypted Password for Elastic Search</a> is added.</li> <li>• Updated the section <a href="#">Copying and Adding Files</a> for Elastic search.</li> <li>• Updated the section <a href="#">Place Files in the Installation Directories</a> for Elastic search.</li> <li>• Updated the Elastic Search version</li> <li>• Updated the section <a href="#">Frequently Asked Questions (FAQs)</a> to add 11 and 12 points.</li> </ul>

Version Number	Revision Date	Change Log
1.2	January 2021	<p>Added the following sections:</p> <ul style="list-style-type: none"> <li>• <a href="#">Loading Analysis Data to Elastic Search</a></li> <li>• <a href="#">Integration with MMG</a></li> <li>• <a href="#">Placing the Files for MMG</a></li> <li>• <a href="#">Integration with MMG (SAML)</a></li> <li>• <a href="#">SSO Token Generation</a></li> <li>• <a href="#">Add Synonyms and Keyword files in Elastic Search</a></li> <li>• <a href="#">Generate Studio Server SSL Configuration Mandatory File</a></li> <li>• <a href="#">Configuring Infer and Training API</a></li> <li>• <a href="#">Create the Hive Schema</a></li> </ul> <p>Updated the following sections:</p> <ul style="list-style-type: none"> <li>• <a href="#">Cleanup for Upgrade</a></li> <li>• <a href="#">Configure the Extract Transfer and Load (ETL) Process for the config.sh file.</a></li> <li>• <a href="#">Place Files in Wallet</a></li> <li>• <a href="#">Generate the Key Store File for Secure Batch Service</a></li> <li>• <a href="#">Spark Interpreter with remote spark cluster</a></li> <li>• <a href="#">Deploy FCC Studio on the K8s Cluster</a></li> </ul>
1.1	November 2020	<p>Added the following sections:</p> <ul style="list-style-type: none"> <li>• <a href="#">Appendix C – Additional Jars – PGX</a> for PGX.</li> <li>• <a href="#">Appendix D – Additional Jars – Batch Service</a> for Batch Services.</li> <li>• <a href="#">Cleanup for Upgrade</a> for upgrade scenarios.</li> </ul> <p>Updated the following sections:</p> <ul style="list-style-type: none"> <li>• <a href="#">Place Files in the Installation Directories</a></li> <li>• <a href="#">Configure the Spark Interpreter</a></li> <li>• <a href="#">Verify the Installation</a></li> </ul>
1.0	September 2020	Revamp

# Table of Contents

<b>1</b>	<b>Preface</b>	<b>9</b>
1.1	Audience	9
1.2	Access to Oracle Support	9
1.3	Related Documents	9
1.4	Conventions	9
1.5	Abbreviations	10
<b>2</b>	<b>Introduction</b>	<b>11</b>
2.1	Solutions offered by FCC Studio	11
2.2	Installation Check List when Studio is Deployed with OFSAA	12
2.3	Installation Check List when Studio is Deployed without OFSAA	14
<b>3</b>	<b>Pre-installation</b>	<b>16</b>
3.1	Prerequisites	16
3.2	Hardware and Software Requirements	17
3.3	Prerequisite Environmental Settings	18
3.4	Download the Big Data Files	19
3.5	Validation Checklist	20
3.6	Configure the Elastic Search Component	21
3.7	Configure the Interpreter Settings	23
3.7.1	Configure the fcc-jdbc Interpreter	24
3.7.2	Configuring the fcc-ore Interpreter	25
3.7.3	Configure the fcc-python Interpreter	25
3.7.4	Configure the Spark Interpreter	26
3.7.5	Configure the PySpark Interpreter	30
3.8	Create the Hive Schema	32
3.9	Create the Studio Schema	32
3.10	Assign Grants for the Database Schemas	33
3.11	Setup Password Stores with Oracle Wallet	33
3.11.1	Setup the Password Stores for Database User Accounts	33
3.11.2	Verify the Connectivity of the Wallet	36
3.11.3	Create the Credential Keystore	37

3.12	Download the Installer Kit .....	39
<b>4</b>	<b>Installation.....</b>	<b>40</b>
4.1	Extract the Installer Kit.....	40
4.2	Place Files in the Installation Directories .....	40
4.3	Add Synonyms and Keyword files in Elastic Search .....	42
4.4	Place Files in Wallet.....	42
4.5	Generate an Encrypted Password .....	43
4.5.1	<i>Generate an Encrypted Password for Elastic Search.....</i>	43
4.6	Generate the Public and Private Keys .....	43
4.7	Apply GDPR and Redaction Changes for FCC Studio.....	44
4.7.1	<i>Generate the Key Store File for Secure Batch Service.....</i>	44
4.7.2	<i>Generate Studio Server SSL Configuration Mandatory File.....</i>	45
4.8	Configure the Preferred Services .....	46
4.9	Configure the studio-env.yml File.....	46
4.10	Configure the Extract Transfer and Load (ETL) Process .....	74
4.10.1	<i>Configure the Extract Transfer Load (ETL) Services .....</i>	74
4.11	Verify the Resource Allocation for FCC Studio Services .....	80
4.12	Resource Limits .....	80
4.13	Resource Types .....	80
4.14	Resource Parameters in FCC Studio .....	80
4.15	Deploy FCC Studio on the K8s Cluster .....	88
<b>5</b>	<b>Post-installation Steps when OFSAA is Installed.....</b>	<b>91</b>
5.1	Verify the Installation .....	91
5.1.1	<i>Check Logs of FCC Studio Services.....</i>	91
5.2	Configure the Schema Creation .....	94
5.2.1	<i>Configure the Schema Creation from the FCC Studio Server.....</i>	94
5.3	Configure the ICIJ Data.....	95
5.4	Restart the PGX Service .....	96
5.5	Loading Analysis Data to Elastic Search.....	97
5.6	Integration with MMG .....	97
5.6.1	<i>Placing the Files for MMG.....</i>	98
5.6.2	<i>Integration with MMG (SAML).....</i>	99
5.7	Access the FCC Studio Application .....	99

5.8	Perform the OFSAA Configuration for Batch Execution .....	100
5.9	Configure and Run Published Notebooks.....	101
<b>6</b>	<b>Post-installation Steps when OFSAA is Not Installed.....</b>	<b>102</b>
6.1	Verify the Installation .....	102
6.1.1	<i>Checking Logs of FCC Studio Services</i> .....	102
6.2	Restart the PGX Service .....	105
6.3	Access the FCC Studio Application .....	105
6.4	Configuring Infer and Incremental Training API .....	107
6.4.1	<i>Prerequisite</i> .....	107
6.4.2	<i>Configurations</i> .....	107
<b>7</b>	<b>Upgrade .....</b>	<b>110</b>
7.1	Upgrade Scenarios.....	110
7.2	Upgrade Steps with OFSAA.....	110
7.3	Cleanup for Upgrade.....	111
7.3.1	<i>Start the FCC Studio Application</i> .....	112
7.4	Upgrade Steps without OFSAA.....	112
7.5	Configure the studio-env.yml File for Upgrade.....	113
7.6	Post-Installation Steps for Upgrade .....	113
<b>8</b>	<b>Reinstall FCC Studio.....</b>	<b>114</b>
8.1	Cleanup for Studio Schema .....	114
8.2	Cleanup for BD or ECM Atomic Schema .....	115
8.3	Cleanup for BD or ECM Config Schema .....	115
<b>9</b>	<b>Appendix - Starting and Stopping Services.....</b>	<b>116</b>
9.1	Start a Service .....	116
9.2	Stop a Service.....	116
<b>10</b>	<b>Appendix A - Tables and Sequences .....</b>	<b>117</b>
10.1	Studio Schema Tables.....	117
10.2	Studio Schema Sequences.....	118
<b>11</b>	<b>Appendix B - Using Python Virtual Environments with PySpark .....</b>	<b>119</b>

11.1	Create a Virtual Environment with Conda.....	119
11.2	Include a Virtual Environment in the Init Container .....	119
11.3	Update the Interpreter Properties .....	120
<b>12</b>	<b>Appendix C – Additional Jars – PGX .....</b>	<b>121</b>
<b>13</b>	<b>Appendix D – Additional Jars – Batch Service .....</b>	<b>123</b>
<b>14</b>	<b>Frequently Asked Questions (FAQs) .....</b>	<b>125</b>



# 1 Preface

This section provides information about the Oracle Financial Services (OFS) Crime and Compliance Studio (FCC Studio) Installation and Configuration Guide with kubernetes deployment.

## 1.1 Audience

OFS FCC Studio Installation and Configuration Guide is intended for System Engineers who are responsible for installing and maintaining the application.

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

- UNIX commands
- Database concepts
- Big Data concepts
- Kubernetes
- Docker

## 1.2 Access to Oracle Support

OFS FCC Studio Installation and Configuration Guide is intended for System Engineers who are responsible for installing and maintaining the application.

Oracle customers have access to electronic support through [My Oracle Support](#). For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>.

Or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing-impaired.

## 1.3 Related Documents

We strive to keep this and all other related documents updated regularly; visit the [OHC Documentation Library](#) to download the latest version available there. The list of related documents is provided here.

- *Oracle Financial Services Crime and Compliance Studio Installation Guide*
- *Oracle Financial Services Crime and Compliance Studio Administration Guide*
- *Oracle Financial Services Crime and Compliance Studio User Guide*
- *Oracle Financial Services Crime and Compliance Studio Data Model Guides*
- *Oracle Financial Services Crime and Compliance Studio Release Notes and Readme*

## 1.4 Conventions

The following text conventions are used in this document.

**Table 1: Document Conventions**

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
<i>italic</i>	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.

## 1.5 Abbreviations

The following table lists the abbreviations used in this document.

**Table 2: Abbreviations**

Abbreviation	Meaning
OFS	Oracle Financial Services
FCC Studio	Financial Crime and Compliance Studio
OFSAA	Oracle Financial Services Analytical Application
K8s	Kubernetes
BD	Behavior Detection
FCDM	Financial Crime Data Model
ICIJ	International Consortium of Investigative Journalists
IDCS	Oracle Identity Cloud Service
MMG	Model Management and Governance
SSO	Single Sign-On
SSH	Secure Shell

## 2 Introduction

The Financial Crime and Compliance (FCC) Studio uses the latest technology to harness the power of Graph Analytics to give Financial Institutions the ability to effectively monitor anti-money laundering and anti-fraud programs in financial institutions.

By using the FCC Studio, financial institutions can deploy advanced machine learning algorithms and artificial intelligence to discover unknown criminals, streamline compliance operations and have unprecedented control over their Financial Crime program.

This chapter provides the information required to understand the deployment of the Oracle Financial Services (OFS) Crime and Compliance Studio (FCC Studio) application on the Kubernetes (K8s) cluster.

This release (v8.0.8.3.0) of FCC Studio can be used for the following:

- To install a new instance of FCC Studio as follows:
- FCC Studio with OFSAA (Oracle Financial Services Analytical Application)
  - Without MMG (Model Management and Governance)
  - With MMG (Model Management and Governance): **MMG pack (8.0.8.0.0), with 80801 patch** can be used on top of existing OFSAA pack. To know more about installing MMG see, [MMG Installation Guide](#).

### NOTE

- Here, with OFSAA is with Behavior Detection (BD) or Enterprise Case Management (ECM).
- There is no recommended order when installing Studio and MMG.
- MMG can also be a standalone installation or sharing same atomic schema of OFSAA (pack-on-pack with OFSAA).
- For integrating MMG with Studio, see [Integration with MMG](#).

- FCC Studio without OFSAA
- To upgrade an existing instance of FCC Studio on the K8s cluster as follows:
- Upgrade FCC Studio from v8.0.8.0.0 onwards to v8.0.8.3.0 with and without OFSAA.

### Topics:

- [Installation Check List when Studio is Deployed with OFSAA](#)
- [Installation Check List when Studio is Deployed without OFSAA](#)

## 2.1 Solutions offered by FCC Studio

The FCC Studio is a platform that can be used by Data Scientists and Data Analysts for perform the FCC analysis.

The following are the solutions offered by FCC Studio:

- **Machine Learning**  
Out-of-the-box notebooks which are language agnostics and supports SparkML, ORE, R, Python, Deep Learning, On Graph, Case Similarity, Predictive Behavioral Models, AAS, and so on for your Machine Learning requirements.
- **Graph Analytics**  
Provides graph analytics such as, Pattern Matching, Graph Algorithm, Case Creation, Graph Based Investigation, PGX, and Spark.
- **Data Discovery**  
Provides out of the box notebooks that can be used for discovering the data. Also, supports Graph Pattern Discovery, Data Visualization, and Graph Questions.
- **Rapid Development**  
Provides REST interface, widgets notebooks in batch, and consumerization data science.
- **Modeling**  
Provides you the flexibility to create and import graph models and visualize the outputs.
- **Explainability**  
The intuitive Interface of the FCC Studio helps you to learn and adapt to the Interface with less training and explanation.
- **Leverage fragmented data**  
Replicates the Oracle Financial Crime Data Model (Graph and Relational) which are on the data lake and the ability to connect with any of the data set.
- **Collaboration**  
Provides an ability to collaborate and share notebooks to improve the productivity.

## 2.2 Installation Check List when Studio is Deployed with OFSAA

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

**Table 3: Installation Check List**

Sl. No.	Activity	Mandatory	Comments
<b>Pre-installation Steps</b>			
1	<a href="#">Install</a> all the prerequisite hardware and software.	Yes	-
2	<a href="#">Setup</a> the environmental settings.	Yes	-
3	<a href="#">Download</a> the files for Big Data	No	Required for graph analytics and leverage fragmented data

4	<a href="#">Configure</a> the Elastic Search Component	No	Required for graph analytics and leverage fragmented data
5	<a href="#">Configure</a> the Interpreter Settings	Yes	-
6	<a href="#">Create</a> the Studio Schema	Yes	-
7	<a href="#">Assign</a> Grants for the Database Schemas	Yes	-
8	<a href="#">Setup</a> Password Stores with Oracle Wallet.	Yes	-
9	<a href="#">Create</a> the Credential Keystore	Yes	-
10	<a href="#">Download</a> the installer kit.	Yes	-
<b>Installation Steps</b>			
1	<a href="#">Extract</a> the installer kit.	Yes	-
2	<a href="#">Place</a> files in the Installation Directories and <a href="#">Place</a> files in wallet.	Yes	-
3	<a href="#">Generate</a> an Encrypted Password	Yes	-
4	<a href="#">Generate</a> the Public Key and Private Key	Yes	-
5	<a href="#">Generate</a> the Key Store File for Secure Batch Service	Yes	-
6	<a href="#">Add</a> the Batch Service (SSL) to PGX Trust Store	Yes	-
7	<a href="#">Configure</a> the Preferred Services	Yes	-
8	<a href="#">Configure</a> the studio-env.yml File	Yes	-
9	<a href="#">Configure</a> the Extract Transfer and Load (ETL) Process	No	Required for graph analytics and leverage fragmented data
10	<a href="#">Verify</a> the Resource Allocation for FCC Studio Services	Yes	-
11	<a href="#">Deploy</a> FCC Studio on the K8s Cluster	Yes	-
<b>Post-Installation Steps</b>			
1	<a href="#">Verify</a> the Installation	Yes	-
2	<a href="#">Configure</a> the Schema Creation	No	Required for graph analytics and leverage fragmented data
3	<a href="#">Configure</a> the ICIJ Data	No	Required for graph analytics and leverage fragmented data
4	<a href="#">Access</a> the FCC Studio Application	Yes	-
5	<a href="#">Perform</a> the OFSAA Configuration for Batch Execution	No	Required if leveraging OFSAA's scheduling and executing capability.
6	<a href="#">Configure</a> and Run Published Notebooks	No	Required if leveraging OFSAA's batch execution.
7	<a href="#">Integrate</a> with MMG	No	Required for leveraging MMG's capabilities.

## 2.3 Installation Check List when Studio is Deployed without OFSAA

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

**Table 4: Installation Check List**

Sl. No.	Activity		
<b>Pre-installation Steps</b>			
1	<a href="#">Install</a> all the prerequisite hardware and software.		
2	<a href="#">Setup</a> the environmental settings.		
3	<a href="#">Download</a> the files for Big Data		
4	<a href="#">Configure</a> the Elastic Search Component		
5	<a href="#">Configure</a> the Interpreter Settings		
6	<a href="#">Create</a> the Studio Schema		
7	<a href="#">Assign</a> Grants for the Database Schemas		
8	<a href="#">Setup</a> Password Stores with Oracle Wallet.		
9	<a href="#">Create</a> the Credential Keystore		
10	<a href="#">Download</a> the installer kit.		
<b>Installation Steps</b>			
1	<a href="#">Extract</a> the installer kit.		
2	<a href="#">Place</a> files in the Installation Directories and <a href="#">Place</a> files in wallet.		
3	<a href="#">Generate</a> an Encrypted Password		
4	<a href="#">Generate</a> the Public Key and Private Key		
5	<a href="#">Generate</a> the Key Store File for Secure Batch Service		
6	<a href="#">Add</a> the Batch Service (SSL) to PGX Trust Store		
7	Configure the Preferred Services		
8	<a href="#">Configure</a> the studio-env.yml File		

---

9	<a href="#">Configure</a> the Extract Transfer and Load (ETL) Process		
10	<a href="#">Verify</a> the Resource Allocation for FCC Studio Services		
11	<a href="#">Deploy</a> FCC Studio on the K8s Cluster		
<b>Post-Installation Steps</b>			
1	<a href="#">Verify</a> the Installation		
2	<a href="#">Access</a> the FCC Studio Application		

## 3 Pre-installation

This chapter provides information about the tasks that must be performed before installing FCC Studio.

### Topics:

- [Prerequisites](#)
- [Hardware and Software Requirements](#)
- [Prerequisite Environmental Settings](#)
- [Download the Big Data Files](#)
- [Validation Checklist](#)
- [Configure the Elastic Search Component](#)
- [Configure the Interpreter Settings](#)
- [Create the Hive Schema](#)
- [Create the Studio Schema](#)
- [Assign Grants for the Database Schemas](#)
- [Setup Password Stores with Oracle Wallet](#)
- [Download the Installer Kit](#)

### 3.1 Prerequisites

The Linux machine must satisfy the following conditions:

- BD (Behavior Detection) or the ECM (Enterprise Case Management) application pack must be installed.
- Kubernetes (k8s) cluster must be installed with the following:
  - Registry to store docker images.
  - Minimum of 8 GB memory (inclusive of all nodes) available for the installation. The actual memory requirement depends on the workload/container size configuration.
  - Docker and kubectl must be installed.
  - kubectl is configured (that is, connected to cluster where you want to install FCC Studio).
  - Docker has push access to a private registry.
  - 25GB free space is available to store the FCC Studio Installer zip file in some directory.
  - 45GB free space is available in the docker root directory. You can find the docker root directory using the `docker info` command.



## 3.2 Hardware and Software Requirements

The following hardware and software are required for this version of FCC Studio. The installation environment or setup must have these requirements for an application to run smoothly and efficiently.

**Table 5: Hardware and Software Requirements**

Hardware or Software Category	Component Version
Browser	<ul style="list-style-type: none"> <li>• Chrome 57.x</li> <li>• Firefox 52.x</li> </ul>
Java Version	Java 8
Docker Registry	<ul style="list-style-type: none"> <li>• Docker registry must be present to store docker images</li> <li>• Min of 45GBspace is required to save docker images</li> </ul>
Database Server	<ul style="list-style-type: none"> <li>• Oracle Database Server 12c Release 2 (12.2.0.1+) Enterprise Edition</li> <li>• Oracle R Enterprise 1.5.1 with Oracle R Distribution 3.3.0</li> </ul>
Elastic Search	elasticsearch-7.9.2
Elastic Search Hadoop Jars	ES - Hadoop v7.9.2 You must download the ZIP file from <a href="https://www.elastic.co/downloads/past-releases/elasticsearch-7-9-2">https://www.elastic.co/downloads/past-releases/elasticsearch-7-9-2</a>
Kubernetes Cluster	
Processing Server	<ul style="list-style-type: none"> <li>• RHEL 7.4+</li> <li>• SFTP</li> <li>• Oracle JRE Standard Edition 1.8.x(with JCE)</li> <li>• Kubernetes(K8s) cluster</li> </ul>
PGX (Graph) Server	<ul style="list-style-type: none"> <li>• RHEL 7.4+</li> <li>• Kubernetes(K8s) cluster</li> <li>• Minimum gcc library version 4.8.2</li> </ul>
Big Data	

Hadoop and Spark	<p><b>NOTE:</b> Kerberos authentication must be enabled for Big Data.</p> <ul style="list-style-type: none"> <li>• Apache Hadoop Version 3.0.0</li> <li>• Apache Hive Version 2.1.1</li> <li>• Apache Spark Version 2.4.0</li> <li>• Apache Sqoop Version 1.4.7</li> </ul> <p>The <code>.profile</code> file must be present with the <code>SPARK_HOME</code> and <code>PYTHON_HOME</code> parameters already set.</p> <p><b>NOTE:</b> The product is certified for Apache-Hadoop, and any vendor-specific Hadoop distributions have to confirm compliance with Apache-Hadoop standards, and if not, the vendor the customer chooses to work with for Hadoop should ensure compliance to Apache-Hadoop standards. Any issue raised on vendor-specific distributions will be fixed only if the issue is reproducible on Apache-Hadoop, Apache-Hive, and Apache-Spark.</p>
Hive Connectors	Hive JDBC Connectors V 2.5.15
Hadoop Security Protocol	<ul style="list-style-type: none"> <li>• Kerberos R release 1.6.1</li> <li>• Sentry-1.4.0</li> </ul>

**NOTE** If Cloudera is not installed, the following versions are required:

- Hadoop Version 3.0.0
- Hive Version 2.1.1
- Spark Version 2.4.0
- Sqoop Version 1.4.7

Kerberos authentication must be enabled for Big Data.

### 3.3 Prerequisite Environmental Settings

The following prerequisite environmental settings must be set before beginning the installation of FCC Studio. These settings are the configuration that a system must have for an application to run smoothly and efficiently.

**Table 6: Prerequisite Environmental Settings**

Category	Expected Value
Java Settings	<p>PATH in the <code>.profile</code> file must be set to include <code>kubectl</code> and the Java Runtime Environment (Java 8) absolute path.</p> <p><b>NOTE:</b></p>

	<p>Ensure the absolute path to <code>JRE/bin</code> is set at the beginning of the <code>PATH</code> variable.</p> <p><b>For example:</b> <code>PATH=/usr/java/jre1.8/bin:\$PATH</code></p> <p>Ensure no <code>SYMBOLIC</code> links to Java installation are set in the <code>PATH</code> variable.</p>
Oracle Database Settings	<p><b>NOTE:</b></p> <p>This setting is required only if the Wallet has to be created on the same server as that of the FCC Studio server.</p> <p>Oracle Processing Server</p> <ul style="list-style-type: none"> <li>• <code>ORACLE_HOME</code> must be set in the <code>.profile</code> file pointing to the appropriate Oracle DB Client installation.</li> <li>• <code>PATH</code> in the <code>.profile</code> file must be set to include the appropriate <code>\$ORACLE_HOME/bin</code> directory.</li> </ul>
Download Directory	Indicates the directory where the product installer zip file is downloaded or copied. The user permission must be set to 755 for this directory.
Installation Directory	<p>Indicates the directory where the product installer zip file is extracted and the installation files are placed. The user permission must be set to 755 for this directory.</p> <p><b>NOTE:</b></p> <p>The Installation and the Download Directory can be the same if the product installer zip file is not copied separately to another directory.</p>
OS Locale	<p>Linux: <code>en_US.utf8</code></p> <p>Execute the following command to check the locale:</p> <pre>locale -a   grep -i 'en_US.utf'</pre> <p>The locale is displayed.</p>

## 3.4 Download the Big Data Files

Download the following configuration files from the Cloudera installation setup:

**NOTE** These files must be kept ready and provided in the following file structure that is used during FCC Studio installation.

**Table 7: Required File Structure**

File Category	File Names
---------------	------------

Hadoop Cluster	<ul style="list-style-type: none"> <li>• hive-site.xml</li> <li>• core-site.xml</li> <li>• hive-env.sh</li> <li>• spark-env.sh</li> <li>• redaction-rules.json</li> <li>• mapred-site.xml</li> <li>• yarn-site.ml</li> </ul>
Kerberos Files	<ul style="list-style-type: none"> <li>• krb5.conf</li> <li>• keytab file name as mentioned in the <code>studio-env.yml</code> file.</li> </ul> <p><b>NOTE:</b> Ensure to rename your <code>.keytab</code> file to <code>fccstudio.keytab</code>.</p>
Additional Jars (PGX)	<p>For information about jars, see <a href="#">Appendix C – Additional Jars – PGX</a>.</p> <p><b>NOTE:</b></p> <ul style="list-style-type: none"> <li>• The version of the jars is client or user-specific. These jars can be obtained from the existing jars of the Cloudera installation.</li> <li>• The <code>HiveJDBC4.jar</code> file is not available in the Cloudera installation setup. You must download the same from the <a href="#">Cloudera</a> website. This is applicable only for Cloudera Cluster.</li> </ul>
Additional Jars (Batch Service)	<p>For information about jars, see <a href="#">Appendix D – Additional Jars – Batch Service</a>.</p>
ES-Hadoop Jars	<p><code>elasticsearch-spark-20_2.11-7.9.2.jar</code></p> <p>To download the <code>elasticsearch-spark-20_2.11-7.2.2.jar</code> file, follow these steps:</p> <ol style="list-style-type: none"> <li>1. Download the ZIP file from <a href="https://www.elastic.co/downloads/past-releases/elasticsearch-7-9-2">https://www.elastic.co/downloads/past-releases/elasticsearch-7-9-2</a>.</li> <li>2. Extract the downloaded file.</li> <li>3. Navigate to the <code>dist</code> directory and download the <code>elasticsearch-spark-20_2.11-7.9.2.jar</code> file.</li> </ol>

## 3.5 Validation Checklist

The Validation Checklist section provide you the information about the parameters that you can validate to avoid installation issues. This section explains the validation and actions that can be taken for some of the common parameters that are used in the `studio-env.yml` file for the installation. The parameters that can be validated are as follows:

**Table 8: Required File Structure**

Parameters	Validation
External Service (OFSAA_SERVICE URL)	The <code>OFSAA_Service</code> URL can be validated by clicking the URL for verification.

DB Details for Studio Schema	You can login to SQL developer and verify the DB Details for Studio Schema.
BD Config Schema	You can login to SQL developer and verify the BD Config Schema.
Studio BE Wallet Details	You can verify the Wallet details by reviewing the steps in <a href="#">Verify the Connectivity of the Wallet</a> .
Atomic Wallet Details	You can verify the Wallet details by reviewing the steps in <a href="#">Verify the Connectivity of the Wallet</a> .
SQL Scripts	You can login to FCC Studio using SQL developer and validate the Studio_DBLINK_BD, if link type is DBLINK, if Schema is not DBLINK, there is no validation required.
Cloudera	You can verify the Cloudera details and validate it by reviewing the steps in <a href="#">Create the Credential Keystore</a> .
Cloudera (Keytab)	Run the command <code>kinit -V &lt;KERBEROS_PRINCIPAL&gt; -k -t &lt;KEYTAB_FILEPATH&gt;</code> to verify the keytab.

## 3.6 Configure the Elastic Search Component

To configure the Elastic Search component, follow these steps:

### NOTE

- Ensure that a minimum of 4GB free RAM space is available for elastic search. If RAM is low, the shards of the elastic search fails and correct result is not fetched.
- You must manually clean the cache if facing performance issue.
- Make sure to have Elastic Search cluster with 3 nodes.

1. Navigate to the `elasticsearch-7.9.2/config` directory.
2. Configure the `elasticsearch.yml` with the following variables:

**Table 9: elasticsearch.yml File**

Interaction Variable Name	Significance
cluster.name	Indicates the name of the cluster.

node.name	Indicates the name given for the node.
node.master	Indicates whether the node is a master.
node.data	Indicates the node data.
path.data	Indicates the directory where you want to store the data.
path.logs	Indicates the directory where you want to store the logs.
network.host	Indicates the hostname of the machine where you want to install the elastic search service.
http.port	Indicates the port number where the elastic search service is installed.
discovery.seed_hosts	(Optional) Indicates the hostnames of the nodes of the cluster.
cluster.initial_master_nodes	(Optional) Indicates the number given to the nodes of the cluster.
indices.breaker.total.use_real_memory	<ul style="list-style-type: none"> <li>Indicates the static setting to determine whether the parent breaker must consider the real memory usage into account or only consider the amount that is reserved by the child circuit breakers.</li> <li>This setting is used to prevent the <code>OutOfMemory</code> error.</li> </ul>

### 3. Configure the `jvm.options` file as follows:

**Table 10: elasticsearch.yml File**

Interaction Variable Name	Significance
-Xms1g	<ul style="list-style-type: none"> <li>Set the value for these parameters. The maximum value set can be up to 50% of the RAM size of the machine.</li> <li>Recommended value: Less than 32GB.</li> </ul>
-Xmx1g	

4. Enter the URL in the following format into the browser:

```
http://<network.host>:<http.port>
```

The following output is displayed to indicate successful installation of the Elastic Search service.

```
{
  "name" : "node-1",
  "cluster_name" : "my-application",
  "cluster_uuid" : "_D-2qEGGSgKQPd3i-UtgWQ",
  "version" : {
    "number" : "7.9.2",
    "build_flavor" : "oss",
    "build_type" : "tar",
    "build_hash" : "1c1faf1",
    "build_date" : "2019-09-06T14:40:30.409026Z",
    "build_snapshot" : false,
    "lucene_version" : "8.1.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

## 3.7 Configure the Interpreter Settings

You must configure the interpreter settings for the following interpreters before installing FCC Studio.

**NOTE** Ensure to configure the settings only for the interpreters that you require.

**Table 11: Pre-installation Interpreter Settings**

Interpreter	Prerequisite Settings
fcc-jdbc	No additional configuration is required. <b>NOTE:</b> The FCC Studio application installed without OFSAA can use the Vanilla jdbc interpreter instead of the fcc-jdbc interpreter to connect to the Studio schema.
fcc-ore	For the required configuration, see <a href="#">Configure the fcc-ore Interpreter</a> .

fcc-pyspark	<ul style="list-style-type: none"> <li>• Install the py4j package in the Spark cluster.</li> <li>• Install the Livy server (0.5.0) on the master node of the Big Data cluster.</li> </ul>
fcc-python	For the required configuration, see <a href="#">Configure the fcc-python Interpreter</a> .
fcc-spark-scala	Install the Livy server (0.5.0) on the master node of the Big Data cluster.
fcc-spark-sql	Install the Livy server (0.5.0) on the master node of the Big Data cluster.
jdbc	No additional configuration is required.
md	No additional configuration is required.
pgql	No additional configuration is required.
pgx-algorithm	No additional configuration is required.
pgx-java	No additional configuration is required.
pyspark	For the required configuration, see <a href="#">Configure the PySpark Interpreter</a> .
spark	For the required configuration, see <a href="#">Configure the Spark Interpreter</a> .

**Topics:**

- [Configure the fcc-jdbc Interpreter](#)
- [Configure the fcc-ore Interpreter](#)
- [Configure the fcc-python Interpreter](#)
- [Configure the Spark Interpreter](#)
- [Configure the PySpark Interpreter](#)

**3.7.1 Configure the fcc-jdbc Interpreter**

To create context for the fcc-jdbc interpreter, follow these steps:



1. Log in to Oracle Database as a **SYSDBA** user.
2. Grant Execute permission to the user using the following command:  

```
grant execute dbms_ols to <Studio_DB_Username>;
```

The Execute permission is granted to user.
3. Grant Create permission to the context using the following command:  

```
grant create any context to <Studio_DB_Username>;
```

The Create permission is granted to context.

## 3.7.2 Configuring the fcc-ore Interpreter

To configure the fcc-ore interpreter you must install the [Oracle R Distribution](#).

### 3.7.2.1 Installing the Oracle R Distribution

To install the Oracle R Distribution (ORD), enable the addons and optional\_latest channels in yum as shown in the following code:

```
```bash
(root)# yum-config-manager --enable ol7_addons
(root)# yum-config-manager --enable ol7_optional_latest
```
```

After completing the previous step, pull ORD from the yum repository using the following command:

```
```bash
(root)# yum install R.x86_64 R-core-extra
```
```

To install ORD, see [Install Oracle R Distribution on Linux](#).

## 3.7.3 Configure the fcc-python Interpreter

The fcc-python interpreter image includes the following:

- The support for python2 has been deprecated.
- By default, the Python interpreter points to the Python 3.6 version.
- The following Python libraries are part of the fcc-python interpreter images for Python 3.6 version:
  - pandas 0.25.3
  - numpy 1.17.4
  - scipy 1.3.2
  - scikit-learn 0.23.2
  - matplotlib-lib 3.3.3
  - seaborn 0.9.1
  - cx-oracle 7.3.0

- `sqlalchemy 1.3.11`
- `xgboost==1.0.1`
- `imbalanced-learn==0.6.2`

To modify the Python packages in the Python 3.6 or to add a different version of Python to the Python interpreter, you must modify the Python image. As an example, follow the steps provided in the *Modifying the Python Images for the Python Interpreter* section in the *OFS Crime and Compliance Studio Administration Guide*.

### 3.7.3.1 Prerequisites for Installing Python Libraries

Following are the prerequisites to install Python libraries:

- FCC Studio supports any-version of Python with the following required conditions:
  - The preferred Python version is installed on the Processing Server (Studio Notebook Server) and is accessible to the Linux user of FCC Studio.
  - The Python Library (py4j) package is present for the preferred Python version.
  - The prerequisites to install the Python 3.6 libraries that are packaged with the FCC Studio Installer are as follows:
    - The Python3.6 version is installed on the Processing Server (Studio Notebook Server) and is accessible to the Linux user of FCC Studio.
    - The Python Library (py4j) package is installed and is accessible to the Linux user of FCC Studio.

## 3.7.4 Configure the Spark Interpreter

### 3.7.4.1 Prerequisites for using the Spark Interpreter

To configure Spark Interpreter, you must download the desired spark distribution from [Spark's official website](#).

For example: `spark-2.4.0-bin-hadoop2.7.tgz` from the [website](#).

[Spark interpreter](#) can be used in several situations as follows:

- Connecting to remote spark cluster
  - With/without Kerberos
  - Custom Hadoop client configuration
  - Custom libraries
- Spark in local mode.

In case you want to connect to remote spark cluster, then obtain following files:

- Hadoop or Hive client configuration as per your use case
- Kerberos files (if applicable)
  - `krb5.conf`
  - `keytab` file

### 3.7.4.1.1 Setting up spark-interpreter

The spark interpreter requires spark distribution to start.

The default configuration of Spark is configured for yarn deployment mode with Kerberos enabled remote spark cluster. For local mode skip this section and follow below.

### 3.7.4.1.2 Spark Interpreter with remote spark cluster

The Spark Interpreter with remote spark cluster can be performed for the following:

- [Configuration with Kerberos enabled remote spark cluster](#)
- [Configuration with Kerberos disabled remote spark cluster](#)

#### Configuration with Kerberos enabled remote spark cluster

1. Move the obtained Kerberos files to `<Studio Installation path>/configmaps/batchservice/user/conf` directory.

**NOTE** These are the same Kerberos files used for ETL. For example, `krb5.conf` and `keytab` files.

2. In case, spark-interpreter should connect with different remote spark cluster or have different Kerberos files, perform the following additional steps:

- a. Create a folder inside `<Studio Installation path>/configmaps/` directory. For example: `spark-kerberos-conf`.
- b. Navigate to `<Studio Installation path>` and open `deploy.sh` and add following command inside function `deployServices()`, case `spark` to create configmap from above Kerberos files.

```
kubectl -n ${NAMESPACE} create configmap spark-kerberos-conf from-
file=configmaps/spark-kerberos-conf.
```

**NOTE** Provide proper names for configmap and folder.

- c. Navigate to `<Studio Installation path>/deployments` directory
- d. Open `spark.yml` and update the configmap name for volumes name: `kerberos` to `spark-kerberos-conf`

```
name: kerberos
configMap:
name: spark-kerberos-conf
```

3. Place the **Hadoop** or **Hive client** configuration files to `<Studio Installation path>/configmaps/spark-interpreter-conf` directory.

**NOTE** Do not remove the `spark-default.conf` and `spark-env.sh` files.

4. Open `spark-default.conf` and update the spark configurations accordingly.
5. Uncomment and update `spark.yarn.dist.files` and `spark.executorEnv.PYTHONPATH`.

**NOTE** The path must be same as the path given for for the downloaded spark distribution. For example: path for `spark-2.4.0-bin-hadoop2.7` spark distribution.

- a. Update the `spark.driver.host`.
- b. Update the `spark.yarn.keytab`.
- c. Update the `spark.yarn.principal`.
- d. If required, you can add additional spark configuration.

#### Configuration with Kerberos disabled remote spark cluster:

1. Place the **Hadoop** or **Hive client** configuration files to `<Studio Installation path>/configmaps/spark-interpreter-conf` directory.

**NOTE** Do not remove the `spark-default.conf` and `spark-env.sh` files.

2. Open `spark-default.conf` and update the spark configurations accordingly.
  - a. Uncomment and update `spark.yarn.dist.files` and `spark.executorEnv.PYTHONPATH`.

**NOTE** The path must be same as the path given for the downloaded spark distribution. For example: path for `spark-2.4.0-bin-hadoop2.7` spark distribution.

- b. Update the `spark.driver.host`.
3. Navigate to `<Studio Installation path>/deployments` and open `spark.yml` and perform the following:
    - `set KERBEROS_ENABLED as FALSE`  
`name: KERBEROS_ENABLED`

```
value: FALSE
```

- Update JAVA\_OPTS by removing:

```
“-Dsun.security.krb5.debug=false -
Djavax.security.auth.useSubjectCredsOnly=false -
Djava.security.krb5.conf=/var/olds-spark-
interpreter/extralibs/kerberos/krb5.conf”
```

### 3.7.4.2 Spark Interpreter in local mode

1. Navigate to `<Studio Installation path>/deployments` and open `spark.yml` and perform the following:

- set `KERBEROS_ENABLED` as `FALSE`

```
name: KERBEROS_ENABLED
```

```
value: FALSE
```

- Update JAVA\_OPTS by removing:

```
“-Dsun.security.krb5.debug=false -
Djavax.security.auth.useSubjectCredsOnly=false -
Djava.security.krb5.conf=/var/olds-spark-
interpreter/extralibs/kerberos/krb5.conf”
```

2. Set `spark.master` as `local[*]` in interpreter configuration file.

### 3.7.4.3 Configuration

The Spark interpreter configuration can be divided into the following categories:

- Configuration related to deployment

These properties can be set either in the Spark libraries, for example, the `spark-defaults.conf` file, or through the system environment variable, `SPARK_CONF`, for example, `SPARK_CONF="--conf spark.driver.memory=2g"`.

#### NOTE

- These properties cannot be changed when the Spark interpreter is running.
- Configuration from interpreter UI screen from Studio has higher precedence over the `spark-default.conf`.

- Configuration related to Spark runtime control

These properties can be set from the Interpreters page of the FCC Studio application UI. This includes properties such as `spark.executor.memory`.

#### NOTE

- The properties related to the driver cannot be set during runtime and are considered deployment configuration. The properties related to the executors can be set during runtime. Hence, the latter option of runtime control configuration is preferred.

- Restart the Studio after changing driver configurations.

A list of possible properties are available in the [Spark Official Documentation](#). All the properties prefixed with the term "zeppelin", that are listed in the [Zeppelin Spark Configuration Document](#) can also be set from the Interpreters page of the FCC Studio application user interface.

## 3.7.5 Configure the PySpark Interpreter

### 3.7.5.1 Prerequisites

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

### 3.7.5.2 Configuration

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

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

### 3.7.5.3 Use the Python Virtual Environments with PySpark

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

To use Python Virtual Environments with PySpark, follow these steps:

- [Create a Virtual Environment with Conda](#)
- [Include Virtual Environment in the Init Container](#)
- [Update the Interpreter Properties](#)

#### 3.7.5.3.1 Create a Virtual Environment with Conda

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

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

1. Ensure that you have `conda` and `conda-pack` installed.

2. Create your virtual environment using the following command:

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

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

3. Activate your virtual environment using the following command:

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

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

```
which python
```

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

5. Compress your virtual environment using the following command:

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

### 3.7.5.3.2 Include Virtual Environment in the Init Container

To include the virtual environment in the Init container, you must place the Virtual Environment in the same path as the Spark libraries. To provide your own Spark libraries and (or) Hadoop client-configuration files for the Spark interpreter to connect to a Spark Standalone or Yarn cluster, follow these steps:

- a. Download the Spark libraries from the [Spark's Official Release Archive](#).
- b. Prepare Spark libraries by following these steps:
  - i. Place your libraries in the `<Studio_Installation_Path>/docker/user/spark-interpreter-libraries/` directory.

**NOTE** Ensure that the directory name of the Spark library is prefixed with the term 'spark'

- ii. To use separate Hadoop libraries, download them as well and place the Hadoop libraries directory in the same directory.
- iii. Be cautious when linking the two libraries, since the path where they are located is in the K8s pod, and the location is as follows:

```
/var/olds-spark-interpreter/interpreter/spark/libs/
```

- iv. Change the image of the Spark interpreter Init Container in the `<Studio_Installation_Path>/deployments/spark.yml` file to `{{imageRepository}}/fcc-studio/3rdparty:init`.
- v. (Optional) Place your Hadoop Client Configuration files in the `<Studio_Installation_Path>/configmaps/spark-interpreter-conf/` directory.

The cluster's Hadoop client-side configuration files that include XML files such as `yarn-site.xml` are required and must be supplied with the Spark libraries. These files are available in the Hadoop configuration directory (`HADOOP_CONF_DIR`) of the cluster or can be downloaded from the cluster manager's user interface if you are using a Cloudera cluster.

### 3.7.5.3.3 Update the Interpreter Properties

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

- In the Spark Interpreter Settings page of the FCC Studio application UI (or `spark.json`), change the following:
  - Change the value of the `spark.yarn.dist.archives` property to `<environment-absolute-path>/<environment-name>.tar.gz#<environment-name>`
  - Change the value of the `spark.pyspark.python` property to `./<environment-name>/bin/python`
- In the PySpark Interpreter Settings page of the FCC Studio application UI (or `pyspark.json`), change the value of the `zeppelin.pyspark.python` parameter to `<environment-absolute-path>/bin/python`.

## 3.8 Create the Hive Schema

To create hive schema, perform the following steps:

1. Login to the server where **bigdata/hive** is installed.
2. Open a hive session in the command prompt.
3. Create a new hive schema using the following script:

```
create database <hive schema name>;
```

A new hive schema is created.

#### NOTE

Ensure that the hive schema that is created is provided as the parameter value `HIVE_SCHEMA` in the `studio-env.yml` file.

## 3.9 Create the Studio Schema

To create studio schema, create a new Oracle Database schema user using the following script:

```
CREATE USER <Studio Schema User Name> IDENTIFIED BY <Password>;
```



A new Oracle Database schema (Studio schema) is created.

## 3.10 Assign Grants for the Database Schemas

Grant the following permissions to the newly created Oracle Database Schema:

- `GRANT create session TO <Studio Schema User>;`
- `GRANT create table TO <Studio Schema User>;`
- `GRANT create view TO <Studio Schema User>;`
- `GRANT create any trigger TO <Studio Schema User>;`
- `GRANT create any procedure TO <Studio Schema User>;`
- `GRANT create sequence TO <Studio Schema User>;`
- `GRANT execute on dbms_ols TO <Studio Schema User>;`
- `GRANT execute on sys.dbms_session TO <Studio Schema User>;`
- `ALTER USER <Studio Schema User> QUOTA 100M ON users;`
- `GRANT create sequence TO <Studio Schema User>;`
- `GRANT create SYNONYM TO <Studio Schema User>;`
- `GRANT create any context TO <BD or ECM Schema User>;`
- `GRANT execute on dbms_ols TO <BD or ECM Schema User>;`

## 3.11 Setup Password Stores with Oracle Wallet

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.

### Topics:

- [Setup the Password Stores for Database User Accounts](#)
- [Verify the Connectivity of the Wallet](#)

### 3.11.1 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 user name 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.

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 [Oracle Database Security Guide](#).

**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.

To create a wallet, follow these steps:

1. 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.

**Figure 1:Wallet Creation**

```

-bash-4.1$ pwd
/
-bash-4.1$ mkstore -wrl                               -create
Oracle Secret Store Tool : Version 12.1.0.2
Copyright (c) 2004, 2014, Oracle and/or its affiliates. All rights reserved.

Enter password:
Enter password again:
-bash-4.1$ mkstore -wrl                               -createCredential Studio_808
Oracle Secret Store Tool : Version 12.1.0.2
Copyright (c) 2004, 2014, 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:
Create credential oracle.security.client.connect_string1
-bash-4.1$ mkstore -wrl                               -createCredential aif_
Oracle Secret Store Tool : Version 12.1.0.2
Copyright (c) 2004, 2014, 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:
Create credential oracle.security.client.connect_string2
-bash-4.1$ mkstore -wrl                               -createCredential aif_
Oracle Secret Store Tool : Version 12.1.0.2
Copyright (c) 2004, 2014, 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:

```

3. Create the database connection credentials for the studio schema alias using the following command:

```
mkstore -wrl <wallet_location> -createCredential <alias-name>
<database-user-name>
```

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.

4. Create the database connection credentials for the atomic schema alias using the following command:

```
mkstore -wrl <wallet_location> -createCredential <alias-name>
<database-user-name>
```

**NOTE** Creating an atomic schema is not required when you install FCC Studio without OFSAA.


5. Create the database connection credentials for the config schema alias using the following command:

```
mkstore -wrl <wallet_location> -createCredential <alias-name>
<database-user-name>
```

**NOTE** Creating a config schema is not required when you install FCC Studio without OFSAA.

In this manner, create a wallet and associated database connection credentials for all the database user accounts.

The wallet is created in the `secrets/<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](#).

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

**Figure 2: Location of the Created Wallet Folder**

| Name        | Size | Changed             | Rights   | Owner |
|-------------|------|---------------------|----------|-------|
| wallet_808_ |      | 12-08-2020 14:52:49 | rwX----- |       |

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

6. Move the wallet folder to the `<Studio_Installed_Path>/<alias-name>` directory.
7. In the `<wallet_location>` directory, configure the `tnsnames.ora` file to include the entry for each alias name to be set up.

- NOTE**
- You can either update the existing `tnsnames.ora` file with the above details or create a new `tnsnames.ora` file and add the required entries.
  - `<alias-name>` is a user-defined value.

### 3.11.2 Verify the Connectivity of the Wallet

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

1. 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
```

2. Test the connectivity using the following command:

**NOTE** The `ORACLE_HOME` used with the wallet must be the same version or higher than what the wallet was created.

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

The output is similar to:

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

To verify if you are connected to the correct user:

```
SQL> show user
```

The output is similar to:

```
USER is "<database-user-name>"
```

### 3.11.3 Create the Credential Keystore

Credential keystore must be created for the Behavior Detection (BD) or Enterprise Case Management (ECM) Atomic schema and FCC Studio Schema. To create a credential keystore, follow these steps:

1. Login as HDFS Superuser.
2. Create a credential keystore on HDFS using the following command:

```
hadoop credential create mydb.password.alias -provider
jceks://hdfs/user/root/oracle.password.jceks
```

Where,

- `mydb.password.alias` is the alias name
  - `hdfs/user/root/oracle.password.jceks` is the file path of the credential keystore
3. Verify the credential keystore file using the following command:

```
hadoop credential list -provider
jceks://hdfs/user/root/oracle.password.jceks
```

4. Grant Read permission to the keystore file using the following command:

```
hadoop fs -chmod 744 /user/root/oracle.password.jceks
```

**NOTE** Ensure the credential keystore file path and the alias are correctly mentioned in the `config.sh` file.

### 3.11.3.1 Copying and Adding Files

To copy the jar files, follow these steps:

1. Create the folder in the GRAPH\_FILES\_PATH parameter in a node of the big data cluster.
2. Create a folder called jars.
3. Navigate to the <STUDIO\_INSTALLED\_PATH>/ficdb/etlJars directory and copy all the jars present in this directory into the big data node inside the folder jars.
4. Perform this step if https is enabled for Elastic Serach:
  - a. Create a new folder with the name as **conf** in the GRAPH\_FILES\_PATH parameter in a node of the big data cluster.
  - b. Place the **es\_truststore.jks** file in the **conf** folder.

#### NOTE

To use the ES-Hadoop connector, download the commons-httpclient-3.0.1.jar and elasticsearch-spark-20\_2.11-7.9.2.jar files and place them in the jars folder.

### 3.11.3.2 Create Credential Keystore for Elastic Search

You must create a credential Keystore for storing Elastic Search Keystore password and elastic search password if HTTPS is enabled for Elastic Search.

To create a credential keystore, follow these steps:

1. Login as HDFS Superuser.
2. Create a credential keystore on HDFS using the following command:

```
hadoop credential create elastic.password.alias -value <elastic search password> \
-provider jceks://<hdfs path>/elastic.password.jceks
hadoop credential create elastic.keystore.password.alias -value password \
-provider jceks://<hdfs path>//elastic/elastic.password.jceks
```

Where,

- elastic.password.alias is the elastic search password alias name
- elastic.keystore.password.alias is the elastic search keystore password alias name
- password is elastic search keystore password
- <hdfs path>/elastic.password.jceks is the file path of the credential keystore

- `<hdfs path>` for example, `hdfs/testuser/fccstudio/test`
3. Verify the credential keystore file using the following command:  

```
hadoop credential list -provider jceks://<hdfs path>/elastic.password.jceks
```
  4. Grant Read permission to the keystore file using the following command:  

```
hadoop fs -chmod 744 /<hdfs path>/elastic.password.jceks
```

**NOTE** Ensure the credential keystore file path and the alias are correctly mentioned in the `config.sh` file.

## 3.12 Download the Installer Kit

To download the software as a `.zip` folder, download the following mandatory one-off patch **31761460** from [My Oracle Support \(MOS\)](#).

## 4 Installation

After you download the installer kit, you can begin the installation, or deployment of the images and containers in the kubernetes environment. Perform the following steps to complete the installation:

- [Extract the Installer Kit](#)
- [Place Files in the Installation Directories](#)
- [Generate an Encrypted Password](#)
- [Generate the Public and Private Keys](#)
- [Generate the Key Store File for Secure Batch Service](#)
- [Configure the Preferred Services](#)
- [Configure the studio-env.yml File](#)
- [Configure the Extract Transfer and Load ETL Process](#)
- [Verifying the Resource Allocation for FCC Studio Services](#)
- [Deploying FCC Studio on the K8s Cluster](#)

### 4.1 Extract the Installer Kit

After downloading the **.zip** folder, follow these steps to extract the folder contents:

1. Extract the contents of the installer archive file in the download directory using the following commands:

```
<FCC_Studio_Installer_Archive_File>_1of2.zip
```

```
<FCC_Studio_Installer_Archive_File>_2of2.zip
```

Both FCC Studio installer files are extracted to the same directory and the `OFS_FCCM_STUDIO` directory is obtained and is referred to as `<Studio_Installation_Path>`.

**WARNING** Do not rename the application installer directory name after extraction from the archive.

2. Navigate to the download directory where the installer archive is extracted and assign execute permission to the installer directory using the following command:

```
chmod 0755 OFS_FCCM_STUDIO -R
```

### 4.2 Place Files in the Installation Directories

To place the required jars and Kerberos files in the required locations, follow these steps:

1. To place the additional jar files, follow these steps:
  - a. Navigate to the `<Studio_Installation_Path>/docker/user/batchservice/lib` directory.



- b. Move the obtained *hdfs-libs* libraries to `<Studio Installation path>/docker/user/hdfs-libs/` directory. For information about jars, see [Appendix C – Additional Jars – PGX](#).

**NOTE**

- The version of the jars is client or user-specific. These jars can be obtained from the existing jars of the Cloudera installation.
- The `HiveJDBC4.jar` file is not available in the Cloudera setup. You must download the same from the [Cloudera](#) website.

2. To place the Kerberos files, follow these steps:

- a. Navigate to the

`<Studio_Installation_Path>/configmap/batchservice/user/conf` directory.

- b. Place the following Kerberos files:

— `krb5.conf`

— `keytab` file name as mentioned in the `studio-env.yml` file.

3. For synonyms and stopwords, navigate to

`<Studio_Installation_Path>/docker/user/analysis` directory.

There are 8 text files in this directory as follows:

- `Country`
- `Gender`
- `Organisation_strip`
- `Organisation_suffix`
- `Name_synonym`
- `Title`
- `Namestop`
- `Cardinal_ordinal`

You can add contents to these files.

**NOTE**

- User can decide to provide any data in the Stopword or Synonym files.
- Each stopword must be provided in a separate line.
- All related synonyms must be provided in the same line separated by a comma.
- All the synonyms must be provided in the same line and ensure that there are no repetitions of the synonym. For Example: `rob, robi, robie, roby, robbi`.
- **Title.txt** file should contain Name Tiles and organization Stopwords.

4. If you want to perform MMG integration steps with Studio, see the [Placing the Files for MMG](#).
5. Perform this step if https is enabled for Elastic Search:
  - a. Copy **es\_truststore.jks** file from `<Elastic_Search_Installation_Path>`.
  - b. Place the **es\_truststore.jks** file in `<FCC_Studio_Installation_Path>/OFS_FCCM_STUDIO/configmaps/matching-service/conf` directory.

**NOTE** Generate the **es\_truststore.jks** file in the `<Elastic_Search_Installation_Path>` before performing this step. This file contains Keystore certificates.

## 4.3 Add Synonyms and Keyword files in Elastic Search

To consider the similarity when performing the elastic search you can add the synonyms and keyword files in Elastic search.

To add synonymns and keyword files in Elastic search, perform the following steps:

1. Create a folder in the name analysis in this `<Elastic Search Installation path>/config` directory.
2. You can add your synonyms and stop words to these files and place the files in the analysis folder:
  - Title.txt
  - Synonym.txt
  - Organisation\_suffix.txt
  - Organisation\_strip.txt
  - Gender\_OLD.txt
  - Gender.txt
  - Country.txt
  - Cardinal\_ordinal.txt

## 4.4 Place Files in Wallet

To place the files in wallet in the required locations, follow these steps:

1. To place the files in wallet, follow these steps:
  - a. Navigate to the `<Studio_Installation_Path>/secrets/wallets` directory.
  - b. Place the following files:
    - tnsnames.ora
    - ewallet.p12
    - cwallet.sso

```

— ewallet.p12.lck
— cwallet.sso.lck

```

2. Place the **sqlnet.ora** file into wallet and update the path for the current wallet location.

```

WALLET_LOCATION =
    (SOURCE =
        (METHOD = FILE)
        (METHOD_DATA =
            (DIRECTORY = <wallet location>)
        )
    )
SQLNET.WALLET_OVERRIDE = TRUE

```

## 4.5 Generate an Encrypted Password

To generate an encrypted password, follow these steps:

1. Set the `export FIC_DB_HOME` path in the `<Studio Installed Directory>/ficdb` directory.
2. Run the `echo $FIC_DB_HOME` command.
3. Go to the `<Studio_Installation_Path>/ficdb/bin` directory and run the `./FCCM_Studio_Base64Encoder.sh <password to be encrypted>` command.

### 4.5.1 Generate an Encrypted Password for Elastic Search

If Elastic Search provides authentication then generate encrypted password for Elastic Search password.

To generate an encrypted password, follow these steps:

1. Set the `export FIC_DB_HOME` path in the `<FCC_Studio_Installation_Path>/OFS_FCCM_STUDIO/ficdb` directory.
2. Run the `echo $FIC_DB_HOME` command.
3. Go to the `<FCC_Studio_Installation_Path>/OFS_FCCM_STUDIO/ficdb/bin` directory.
4. Run the `./FCCM_Studio_Base64Encoder.sh <Elastic search password to be encrypted>` command.

## 4.6 Generate the Public and Private Keys

The Public and Private keys are JSON Web Tokens (JWT) that are generated for PGX Authentication from FCC Studio.

To generate the keys, follow these steps:

**NOTE** The following steps are mandatory for the first time FCC Studio installation.

1. Navigate to the `<Studio_Installation_Path>/ficdb/bin` directory.
2. Run the Shell Script `FCCM_Studio_JWT_Keygen.sh` from the directory.  
The Public and Private Keys are generated and available in the `<Studio_Installation_Path>/ficdb/conf` directory.
3. Copy the `private.key` and `public.key` files to the `<Studio_Installation_Path>/secrets/keys` directory.

After generating the key store file and adding the batch service to the PGX trust store, configure the user mapping for GDPR and Redaction changes in the database.

## 4.7 Apply GDPR and Redaction Changes for FCC Studio

The General Data Protection Regulation (GDPR) is a regulation in EU law on data protection and privacy in the European Union and the European Economic Area. You can apply the GDPR changes that is required for FCC Studio.

To apply GDPR and Redaction, you must configure the following:

- [Generate the Key Store File for Secure Batch Service](#)
- [Add the Batch Service \(SSL\) to PGX Trust Store](#)

### 4.7.1 Generate the Key Store File for Secure Batch Service

Generating the Key Store file for Secure Batch Service is a process of generating the key store parameters and changing the key store parameters from HTTP to HTTPS protocol.

**NOTE** Proceed with this section only if you want to create a self-signed certificate.

To configure the Key Store file for Secure Batch Service, follow these steps:

1. Run the `keytool -genkey -alias batchservice -keyalg RSA -keysize 2048 -keystore <Studio_Installation_Path>/OFS_FCCM_STUDIO/secrets/keys/<Keystore file name>.jks` command in the Studio Server.

**NOTE** For Kubernetes, this keystore file name should be `batchservice_keystore.jks`.

When generating the keytool ensure to provide the hostname in first name. For example:

**Question:** What is your first and last name?

**Answer:** Provide the value batchservice name. For example: **batchservice**.

2. Specify the keystore password. The <Keystore file name>.jks file is created in the path <Studio\_Installation\_Path>/OFS\_FCCM\_STUDIO/secrets/keys directory.
3. Specify the following parameters in the *Studio env.yml* file.
  - KEYSTORE\_FILE\_NAME=<Keystore file name>.jks
  - KEYSTORE\_PASS=<password>

## 4.7.2 Generate Studio Server SSL Configuration Mandatory File

To generate the Studio Server SSL Configuration Mandatory File, perform the following steps:

**NOTE** Proceed with this section only if you want to create a self-signed certificate.

1. Run the following jks command in the Studio Server:

```
keytool -genkey -alias <alias> -keyalg RSA -keystore <alias>.jks
```

**NOTE** The keystore file name must be `studio_server.jks`.

2. Specify the keystore password.
3. When generating the keytool ensure to provide the hostname in first name. For example:

**Question:** What is your first and last name?

**Answer:** Provide the fully qualified studio server hostname.

For example, <hostname>.<domain name>

4. Specify the any name for the other questions.
5. Specify the keystore password. The jks file is created in the Studio Server.
6. Run the following jks command in the Studio Server;

**NOTE**

- Use the same keystore name that is used for jks for importing to P12 file (`studio_server`).
- You must use the same **password** and **alias** that is provided in the `studio-env.yml` file.

```
keytool -importkeystore -srckeystore <alias>.jks -destkeystore <alias_name>.p12 -srcalias <alias> -srcstoretype jks -deststoretype pkcs12
```

7. Specify the keystore password. The .p12 file is created in the Studio Server.
8. Copy the .jks and .p12 files and place in the <Studio Installation path>/secrets/keys directory.

## 4.8 Configure the Preferred Services

To configure the preferred services to be deployed during deployment of FCC Studio, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/bin/` directory.
2. Set the `deployment` parameter depending on the services you want to deploy in the `serviceMapping.sh` file. Set the `deployment` parameter to `All` if you want to deploy all services, or set the `deployment` parameter to `Custom` if you want to choose specific services to deploy.

If the `deployment` parameter is set to `Custom`, then set the values of the desired services to `true` and the values of undesired services to `false`.

A sample `serviceMapping.sh` file is as follows.

**Figure 4: Sample `serviceMapping.sh` File**

```
#!/usr/bin/env bash
## This shell-script is used as configuration for starting only desired services
## Set the value of parameter, deployment, as :
## "All" for starting all services
## "Custom" for starting selected services and then set "true"/"false", where "true" means service is desired to be installed.
deployment=All
## For custom:
## FCC Studio independent of OFSAA: set "false" for service(s): authservice, entity-resolution, and matching-service
## FCC Studio lite: set "false" for service(s): fcc-pgql, fcc-pgx-algorithm, fcc-pgx-java and pgx-server.
##Services:
server=true
```

### NOTE

- Do not set `false` for the `server` service.
- Set `spark` as `false` if you don't intend to use spark.

## 4.9 Configure the `studio-env.yml` File

To configure the `studio-env.yml` file for installing FCC Studio, follow these steps:

1. Login to the server as a non-root user.
2. Navigate to the `<Studio_Installation_Path>/secrets/` directory.
3. Configure the `studio-env.yml` file as shown in the following table.

A sample `studio-env.yml` file is as follows.

**Figure 4: Sample `studio-env.yml` file**

```

apiVersion: v1
kind: Secret
metadata:
  name: studio-env
stringData:
  NON_OFSSA: 'false'
  REALM: com.oracle.ofss.fccm.studio.saml.FccSamlRealm
  OFSAA_SERVICE_URL: http://[REDACTED]
  LIVY_HOST_URL: ##LIVY_HOST_URL##
  STUDIO_DB_HOSTNAME: [REDACTED]
  STUDIO_DB_PORT: '1521'
  STUDIO_DB_SERVICE_NAME: [REDACTED]
  STUDIO_DB_SID: [REDACTED]
  STUDIO_DB_USERNAME: [REDACTED]
  STUDIO_DB_PASSWORD: [REDACTED]
  STUDIO_DB_ENCRYPTED_PASSWORD: [REDACTED]
  STUDIO_ALIAS_NAME: [REDACTED]
  STUDIO_WALLET_LOCATION: [REDACTED]

```

**NOTE**

- Do not alter the parameter values that are already set in the `studio-env.yml` file
- Retain the existing placeholder values for the parameters which are not shown as mandatory in the following table.
- You must manually set the parameter value in the `studio-env.yml` file. If a value is not applicable, enter `NA` and ensure that the value is not entered as `NULL`.
- Depending on the installation architecture, ensure to provide the correct hostname for URL of PGX service in the `studio-env.yml` file.
- When upgrading FCC Studio with OFSAA, ensure to provide the same BD database, Studio schema, Hive schema, wallet related information that you used during the installation of the existing instance of FCC Studio.
- When upgrading FCC Studio with non-OFSAA, ensure to provide the same Studio schema and wallet related information that you used during the installation of the existing instance of FCC Studio.

**Table 12: studio-env.yml Parameters**

| Parameter | Significance  | Installing with OFSAA (Mandatory) | Upgrading with OFSAA (Mandatory) | Installing without OFSAA (Mandatory) |
|-----------|---|-----------------------------------|----------------------------------|--------------------------------------|
| NON_OFSAA | Indicates the type of installation. <ul style="list-style-type: none"><li>To install FCC Studio with OFSAA on the Kubernetes cluster, enter <i>false</i>.</li><li>To install FCC Studio without OFSAA on the Kubernetes cluster, enter <i>true</i>.</li></ul>   | Enter false                       | Enter false                      | Enter true                           |
| REALM     | Realm indicates functional grouping of database schemas and roles that must be secured for an application. Realms protect data from access through system privileges; realms do not give additional privileges to its owner or participants.<br>FCC Studio uses realm based authorization and authentication for its users. For more information, see the Realm Based Authorization for FCC Studio section in the <a href="#">OFS Crime and Compliance Studio Administration Guide</a> .<br>The FCC Studio application can be accessed using the following realms:<br><b>FCCMRealm</b><br><b>Value</b> =com.oracle.ofss.fccm.studio.datastudio.auth.FCCMRealm | Yes                               | Yes                              | Yes                                  |



|                   |  |     |     |    |
|-------------------|--|-----|-----|----|
|                   | <p><b>IdcsRealm</b><br/><b>Value</b>=oracle.datastudio.realm.idcs.IdcsRealm</p> <p><b>DemoRealm</b><br/><b>Value</b>=com.oracle.ofss.fccm.studio.datastudio.auth.DemoRealm</p> <p><b>NOTE:</b><br/>The DemoRealm is used only for demo purpose and is not recommended for usage.</p> |     |     |    |
| OFSAA_SERVICE_URL | <p>Indicates the URL of the main database being used for installation.<br/>For example:<br/>https://&lt;HostName&gt;:&lt;PortNo&gt;/&lt;ContextName&gt;/rest-api</p>   | Yes | Yes | No |
| LIVY_HOST_URL     | <p>Indicates the URL of the Livy application.<br/><b>Example:</b><br/>http://&lt;HostName&gt;:&lt;PortNo&gt;</p> <p><b>NOTE:</b><br/>This parameter is applicable only if fcc-spark-sql, fcc-spark-scala, and (or) fcc-pyspark interpreters are to be used.</p>                      | No  | No  | No |

| DB Details for Studio Schema |  |     |     |     |
|------------------------------|--|-----|-----|-----|
| STUDIO_DB_HOSTNAME           | Indicates the hostname of the database where Studio schema is created.             | Yes | Yes | Yes |
| STUDIO_DB_PORT               | Indicates the port number where Studio schema is created.                          | Yes | Yes | Yes |
| STUDIO_DB_SERVICE_NAME       | Indicates the service name of the database where Studio schema is created.         | Yes | Yes | Yes |
| STUDIO_DB_SID                | Indicates the SID of the database where Studio schema is created.                  | Yes | Yes | Yes |
| STUDIO_DB_USERNAME           | Indicates the username of the Studio Schema (newly created Oracle Schema).         | Yes | Yes | Yes |
| STUDIO_DB_PASSWORD           | Indicates the password for the newly created schema. This value must not be blank. | Yes | Yes | Yes |

|   |  |     |     |     |
|---|--|-----|-----|-----|
| STUDIO_DB_ENCRYPTED_PASSWORD  | Indicates the encrypted password that is provided for the Studio schema.<br>For example, cGFzc3dvcmQ.                      | Yes | Yes | Yes |
| Studio DB Wallet Details<br>For more information on creating wallet, see <a href="#">Appendix - Setting Up Password Stores with Oracle Wallet</a> . |  |     |     |     |
| STUDIO_ALIAS_NAME   | Indicates the Studio alias name.<br><b>NOTE:</b><br>Enter the alias name that was created during wallet creation.          | Yes | Yes | Yes |
| STUDIO_WALLET_LOCATION  | Indicates the Studio wallet location.  | Yes | Yes | Yes |
| STUDIO_TNS_ADMIN_PATH   | Indicates the path of the <code>tnsnames.ora</code> file where an entry for the <code>STUDIO_ALIAS_NAME</code> is present. | Yes | Yes | Yes |
| Hadoop Connection Details   |  |     |     |     |

|                                 |  |     |     |          |
|---------------------------------|--|-----|-----|----------|
| STUDIO_HADOOP_CREDENTIAL_ALIAS  | Indicated the alias password saved in Hadoop.<br>For example, studio.password.alias          | Yes | Yes | Yes      |
| STUDIO_HADOOP_CREDENTIAL_PATH   | Indicates the credentials path.<br>For example, <Studio Installed Path>oracle.password.jceks | Yes | Yes | Yes      |
| DB Details for BD Config Schema |  |     |     |          |
| BD_CONFIG_HOSTNAME              | Indicates the hostname of the database where BD or ECM config schema is installed.           | Yes | Yes | Enter NA |
| BD_CONFIG_PORT                  | Indicates the port of the database where BD or ECM config schema is installed.               | Yes | Yes | Enter NA |
| BD_CONFIG_SERVICE_NAME          | Indicates the service name of the database where BD or ECM config schema is installed.       | Yes | Yes | Enter NA |
| BD_CONFIG_SID                   | Indicates the SID of the database where BD or ECM config schema is installed.                | Yes | Yes | Enter NA |

|   |   |     |     |          |
|---|---|-----|-----|----------|
| BD_CONFIG_USERNAME  | Indicates the username for the BD, ECM, or MMG config schema.   | Yes | Yes | Enter NA |
| BD_CONFIG_PASSWORD  | Indicates the password for the BD, ECM, or MMG config schema. This value must not be blank.                                   | Yes | Yes | Enter NA |
| BD Config Wallet Details<br>For more information on creating wallet, see <a href="#">Appendix - Setting Up Password Stores with Oracle Wallet</a> . |   |     |     |          |
| BD_CONFIG_ALIAS_NAME  | Indicates the BD or ECM config alias name.<br><b>NOTE:</b><br>Enter the alias name that was created during wallet creation.   | Yes | Yes | Enter NA |
| BD_CONFIG_WALLET_LOCATION   | Indicates the BD or ECM config wallet location.   | Yes | Yes | Enter NA |
| BD_CONFIG_TNS_ADMIN_PATH  | Indicates the path of the <code>tnsnames.ora</code> file where an entry for the <code>BD_CONFIG_ALIAS_NAME</code> is present. | Yes | Yes | Enter NA |

| DB Details for BD Atomic Schema |  |     |     |          |
|---------------------------------|--|-----|-----|----------|
| BD_ATOMIC_HOSTNAME              | Indicates the BD or ECM atomic schema hostname.                                      | Yes | Yes | Enter NA |
| BD_ATOMIC_PORT                  | Indicates the BD or ECM atomic schema port number.                                   | Yes | Yes | Enter NA |
| BD_ATOMIC_SERVICE_NAME          | Indicates the BD or ECM atomic schema service name.                                  | Yes | Yes | Enter NA |
| BD_ATOMIC_SID                   | Indicates the BD or ECM atomic schema SID.   | Yes | Yes | Enter NA |
| BD_ATOMIC_USERNAME              | Indicates the username of the BD or ECM atomic schema.                               | Yes | Yes | Enter NA |
| BD_ATOMIC_PASSWORD              | Indicates the password of the BD or ECM atomic schema. This value must not be blank. | Yes | Yes | Enter NA |

|  |   |     |     |          |
|--|---|-----|-----|----------|
| BD Atomic Wallet Details.<br>For more information on creating wallet, see <a href="#">Appendix - Setting Up Password Stores with Oracle Wallet</a> . |   |     |     |          |
| BD_ATOMIC_ALIAS_NAME   | Indicates the BD or ECM atomic alias name.<br><b>NOTE:</b><br>Enter the alias name that was created during wallet creation. | Yes | Yes | Enter NA |
| BD_ATOMIC_WALLET_LOCATION  | Indicates the BD or ECM atomic wallet location.   | Yes | Yes | Enter NA |
| BD_ATOMIC_TNS_ADMIN_PATH   | Indicates the path of the tnsnames.ora file where an entry for the BD_ATOMIC_ALIAS_NAME is present.                         | Yes | Yes | Enter NA |
| SQL Scripts  |   |     |     |          |
| FSINFODOM  | Indicates the name of the BD or ECM Infodom.  | Yes | Yes | Enter NA |
| FSSEGMENT  | Indicates the name of the BD or ECM segment.  | Yes | Yes | Enter NA |

|   |   |     |     |          |
|---|---|-----|-----|----------|
| DATAMOVEMENT_LINK_NAME  | <ul style="list-style-type: none"><li>• If the newly created schema is in a different database host, you must create a DB link and provide the same link in this parameter. Alternatively, you can provide the source schema name.</li><li>• If no DB link is present, provide the &lt;SCHEMA_NAME&gt; in this parameter.</li><li>• If the newly created schema is in the same database host, the value for this parameter is the user name of the BD or ECM atomic schema.</li></ul> | Yes | Yes | Enter NA |
| DATAMOVEMENT_LINK_TYPE  | If the DB link is used, enter DBLINK in this field. If the DB link is not used, enter SCHEMA in this field.   | Yes | Yes | Enter NA |
| Cloudera Setup Details<br>Contact System Administrator to obtain the required details for these parameters. |   |     |     |          |
| HADOOP_CREDENTIAL_PROVIDER_PATH   | Indicates the path where Hadoop credential is stored.   | Yes | Yes | Enter NA |



|                       |   |     |     |          |
|-----------------------|---|-----|-----|----------|
| HADOOP_PASSWORD_ALIAS | Indicates the Hadoop alias given when creating the Hadoop credentials.<br><b>NOTE:</b><br>Enter the alias name that was created during wallet creation.<br>For information on how to create a credential keystore, see <a href="#">Creating the Credential Keystore</a> | Yes | Yes | Enter NA |
| Hive_Host_Name        | Indicates the Hive hostname.  | Yes | Yes | Enter NA |
| Hive_Port_number      | Indicates the Hive port number.<br>Contact your System Administrator to obtain the port number.   | Yes | Yes | Enter NA |
| HIVE_PRINCIPAL        | Indicates the Hive Principal.<br>Contact your System Administrator to obtain this value.  | Yes | Yes | Enter NA |
| HIVE_SCHEMA           | Indicates the new Hive schema name.   | Yes | Yes | Enter NA |
| JAAS_CONF_FILE_PATH   | Created for future use.   | No  | No  | No       |

|                             |  |     |     |          |
|-----------------------------|--|-----|-----|----------|
| Krb_Host_FQDN_Name          | Indicates the Kerberos host FQDN name.   | Yes | Yes | Enter NA |
| Krb_Realm_Name              | Indicates the Kerberos realm name.   | Yes | Yes | Enter NA |
| Krb_Service_Name            | Indicates the Kerberos service name.<br>Example: Hive                                  | Yes | Yes | Enter NA |
| KRB5_CONF_FILE_PATH         | Created for future use.  | No  | No  | No       |
| security_krb5_kdc_server    | Created for future use.  | No  | No  | No       |
| security_krb5_realm         | Created for future use.  | No  | No  | No       |
| server_kerberos_keytab_file | Created for future use.<br><b>Note:</b> This is required when using Spark Interpreter. | Yes | Yes | Enter NA |

|                                |   |     |     |          |
|--------------------------------|---|-----|-----|----------|
| server_kerberos_krb5_conf_file | Created for future use.<br><b>Note:</b> This is required when using Spark Interpreter.  | Yes | Yes | Enter NA |
| server_kerberos_principal      | Created for future use.<br><b>Note:</b> This is required when using Spark Interpreter.  | Yes | Yes | Enter NA |
| SQOOP_HOSTMACHINE_USER_NAME    | Indicates the user name of the Big Data server where SQOOP will run.  | Yes | Yes | Enter NA |
| SQOOP_PARAMFILE_PATH           | <ol style="list-style-type: none"> <li>1. Create a file with the name <code>sqoop.properties</code> in the Big Data server and add the following entry to the same:</li> <li>2. <code>oracle.jdbc.mapDateToTimestamp=false</code></li> <li>3. Enter the location of the <code>sqoop.properties</code> file as the value for this parameter.</li> <li>4. Example: <code>/scratch/ofsa/</code></li> </ol> <p><b>NOTE:</b><br/>Ensure that the location name ends with a <code>'/'</code>.</p> | Yes | Yes | Enter NA |
| SQOOP_PARTITION_COL            | Indicates the column in which the HIVE table is partitioned.<br>The value must be <code>SNAPSHOT_DT</code>  | Yes | Yes | Enter NA |

|                    |  |     |     |          |
|--------------------|--|-----|-----|----------|
| SQOOP_TRG_HOSTNAME | Indicates the hostname of the Big Data server where SQOOP will run.  | Yes | Yes | Enter NA |
| SQOOP_TRG_PASSWORD | Indicates the password of the user of the Big Data server where SQOOP will run. This value must not be blank.  | Yes | Yes | Enter NA |
| SQOOP_WORKDIR_HDFS | Indicates the SQOOP working directory in HDFS.<br>Example: /user/ofsa  | Yes | Yes | Enter NA |
| Internal Services  |  |     |     |          |
| AUTH_SERVICE_URL   | Indicates the AUTH service URL that gets activated after the <code>fcc-studio.sh</code> file runs.<br>Example:<br><code>http://&lt;HostName&gt;:7041/authservice</code>          | Yes | Yes | No       |
| BATCH_SERVICE_URL  | Indicates the Batch service URL that gets activated after the <code>fcc-studio.sh</code> file runs.<br><b>Example:</b><br><code>http://&lt;HostName&gt;:7043/batchservice</code> | Yes | Yes | Yes      |

|   |   |     |     |     |
|---|---|-----|-----|-----|
| META_SERVICE_URL  | Indicates the META service URL that gets activated after the <code>fcc-studio.sh</code> file runs.<br><b>Example:</b><br><code>http://&lt;HostName&gt;:7045/met<br/>aservice</code>       | Yes | Yes | Yes |
| SESSION_SERVICE_URL   | Indicates the Session service URL that gets activated after the <code>fcc-studio.sh</code> file runs.<br><b>Example:</b><br><code>http://&lt;HostName&gt;:7047/ses<br/>sionservice</code> | Yes | Yes | Yes |
| PGX_SERVER_URL  | Indicates the URL of the PGX server.<br><b>Example:</b><br><code>http://&lt;HostName&gt;:&lt;PortNo&gt;</code><br>The value for PortNo must be 7007.                                      | Yes | Yes | Yes |
| ORE Interpreter Settings<br><b>NOTE:</b><br>This section is applicable only if ORE interpreter is used. |   |     |     |     |
| RSERVE_USERNAME   | Indicates the RServe username.<br>Contact your System Administrator for the username.   | No  | No  | No  |
| RSERVE_PASSWORD   | Indicates the RServe password.<br>Contact your System Administrator for the username.   | No  | No  | No  |

|                                      |  |     |     |     |
|--------------------------------------|--|-----|-----|-----|
| HTTP_PROXY                           | Indicates the proxy for the host where FCC Studio is deployed.   | No  | No  | No  |
| HTTPS_PROXY                          | Indicates the proxy for the host where FCC Studio is deployed.   | No  | No  | No  |
| <b>ElasticSearch Cluster details</b> |  |     |     |     |
| ELASTIC_SEARCH_HOSTNAME              | Indicates the hostname of the database where the elastic search service is installed.                    | Yes | Yes | Yes |
| ELASTIC_SEARCH_PORT                  | Indicates the port number where the elastic search service is installed.                                 | Yes | Yes | Yes |
| ELASTIC_SEARCH_USERNAME              | Elastic Search Username (Not Applicable, if https enabled is false and authentication is not supported). | Yes | Yes | Yes |
| ELASTIC_SEARCH_ENCRYPTED_PASSWORD    | Encrypted password (Not Applicable, if https enabled is false and authentication is not supported).      | Yes | Yes | Yes |

|   |  |     |     |     |
|---|--|-----|-----|-----|
| ELASTIC_SEARCH_HTTPS_ENABLED                    | true (If ES is https enabled, else false)  | Yes | Yes | Yes |
| ELASTIC_SEARCH_TRUSTSTORE_FILENAME              | The filename of the Elastic Search keystore that contains the certificates of ES host to trust (Not Applicable, if https enabled is false) | Yes | Yes | Yes |
| ELASTIC_SEARCH_TRUSTSTORE_PASSWORD              | The encrypted password of Elastic search keystore file. (Not Applicable, if https enabled is false).                                       | Yes | Yes | Yes |
| ELASTIC_SEARCH_HADOOP_PASSWORD_ALIAS            | Indicates the password alias for Elastic Search (Not applicable if ES ELASTIC_SEARCH_HTTPS_ENABLED is false).                              | Yes | Yes | Yes |
| ELASTIC_SEARCH_KEYSTORE_HADOOP_CREDENTIAL_ALIAS | Indicates the password alias for Elastic Search (Not applicable if ES ELASTIC_SEARCH_HTTPS_ENABLED is false).                              | Yes | Yes | Yes |
| ELASTIC_SEARCH_HADOOP_CREDENTIAL_PATH           | Indicates the elastic search hadoop credential path.   | Yes | Yes | Yes |
| <b>Matching Service</b>                         |  |     |     |     |

|                           |  |     |     |     |
|---------------------------|--|-----|-----|-----|
| MATCHING_SERVICE_URL      | Indicates matching service url<br>Fro example :<br>http://<HostName>:7049/matching-service         | Yes | Yes | Yes |
| MATCHING_SERVICE_HOSTNAME | Indicates the hostname of the database where matching service is installed.                        | Yes | Yes | Yes |
| MATCHING_SERVICE_PORT     | Indicates the port number where matching service is installed.                                     | Yes | Yes | Yes |
| ER_SERVICE_URL            | Indicates the URL of the entity resolution service.  | Yes | Yes | Yes |
| ER_SERVICE_PORT           | Indicates the port number where the entity resolution service is installed.<br>Default value: 7051 | Yes | Yes | Yes |
| Graphs                    |  |     |     |     |
| HDFS_GRAPH_FILES_PATH     | Indicates the file path in the HDFS where the graph.json file is formed.                           | Yes | Yes | Yes |
| GRAPH_FILES_PATH          | Indicates the directory in the Big Data server for graph files.                                    | Yes | Yes | Yes |



|                       |   |     |     |     |
|-----------------------|---|-----|-----|-----|
| GRAPH_NAME            | Indicates the name you want to assign to the global graph at the end of ETL.  | Yes | Yes | Yes |
| <b>ETL</b>            |   |     |     |     |
| HDFS_GRAPH_FILES_PATH | Indicates the filepath in the HDFS where the graph.json is formed.  | Yes | Yes | No  |
| GRAPH_FILES_PATH      | Indicates the directory in the Big Data server for graph files.   | Yes | Yes | No  |
| GRAPH_NAME            | Indicates the name you want to assign to the global graph at the end of ETL.  | Yes | Yes | No  |
| ETL_PROCESSING_RANGE  | Indicates the duration for which the data would be moved from Oracle to Hive.<br><br>For example: If the ETL_PROCESSING_RANGE = 2Y, 3M, 10D, that is, 2 years, 3 months, and 10 days, and the present date is 20200814, then the data movement occurs for the range 20180504 to 20200814. | Yes | Yes | No  |

|  |   |     |     |     |
|--|---|-----|-----|-----|
| OLD_GRAPH_SESSION_DURATION   | Indicates the session older than this specified duration will be removed from the PGX server. If unsure, you can set this value for a week (7D).                  | Yes | Yes | No  |
| REMOVE_TRNXS_EDGE_AFTER_DURATION   | Indicates the date range for which transaction edges will be maintained in graph. For example: 6Y, 3M, 10D, which means 6 years, 3 months and 15 days.            | Yes | Yes | No  |
| CONNECTOR_CHANGESET_SIZE   | Indicates the number of nodes or edges you want to process during an update of graph. If unsure, you can set it to 10000.   | Yes | Yes | No  |
| PGX_SERVER_URLS  | Indicates the comma ',' separated values of PGX URLs. If you have only one PGX URL, then the value is <code>http://&lt;k8s master hostname FQDN&gt;:7007</code> . | Yes | Yes | No  |
| <b>Quantifind Details</b><br>For Quantifind, the generated Quantifind token must be encoded. Use the <code>&lt;Fic_DB_path&gt;/FCCM_Studio_Base64Encoder.sh</code> file for encoding Quantifind token. |   |     |     |     |
| QUANTIFIND_URL   | Indicates the URL of the Quantifind. For example, <code>https://api-test.quantifind.com</code>  | Yes | Yes | Yes |

|                            |  |     |     |     |
|----------------------------|--|-----|-----|-----|
| ENCRYPTED_QUANTIFIND_TOKEN | Indicates the token that is generated when integrating with Quantifind.<br>For example,<br>c2FtcGxIX2VuY3J5cHRIZF9xdWFudGhmaW5kX3Rva2Vu. | Yes | Yes | Yes |
| QUANTIFIND_APPNAME         | Indicates the Quantifind App Name.<br>For example, OracleIntegrationTest   | Yes | Yes | Yes |
| QUANTIFIND_ENABLED         | Indicates that Quantifind is enabled.<br>Options are True or False.  | Yes | Yes | Yes |
| HTTPS_PROXY_HOST           | Indicates the proxy host that is used.<br>For example, www-proxy-idc.in.oracle.com   | Yes | Yes | Yes |
| HTTPS_PROXY_PORT           | Indicates the proxy port that is used.<br>For example, 80  | Yes | Yes | Yes |
| HTTPS_PROXY_USERNAME       | Indicates the proxy username used if there is any.<br>For example,<br>##HTTP_PROXY_USERNAME##  | Yes | Yes | Yes |
| HTTPS_PROXY_PASSWORD       | Indicates the proxy password used if there is any.<br>For example,<br>##HTTP_PROXY_PASSWORD##  | Yes | Yes | Yes |

|  |   |            |            |            |
|--|---|------------|------------|------------|
| <p><b>SAML</b></p> <p>The SAML related parameters are applicable only if SAMLRealm is used in the Realm parameter.</p> | <ol style="list-style-type: none"> <li>1. For SAML Realm, the certificate from IDP (key.cert file) is required.</li> <li>2. The certificate that is obtained from the IDP must be renamed to key.cert and placed in the &lt;Studio_Installation_Path&gt;/OFS_FCCM_STUDIO/databastudio/server/conf directory.</li> <li>3. This certificate is used to identify the trust of the SAML response from the Identity Provider.</li> <li>4. Specify the Role Attribute name from IDP, in which the User Roles are present in the SAML response.</li> </ol> |            |            |            |
| <p>SAML_ISSUER</p>   | <p>Indicates the SAML entity ID (Studio URL) configured in the IDP.</p>   | <p>Yes</p> | <p>Yes</p> | <p>Yes</p> |
| <p>SAML_DESTINATION</p>  | <p>Indicates the SAML IDP URL that is provided by the Identity Provider after creating the SAML Application.</p>  | <p>Yes</p> | <p>Yes</p> | <p>Yes</p> |
| <p>SAML_ASSERTION</p>  | <p>Indicates the SAML consume URL (Studio/URL/saml/consume) that is configured in IDP.</p>  | <p>Yes</p> | <p>Yes</p> | <p>Yes</p> |
| <p>SAML_ROLE_ATTRIBUTE</p>   | <p>Indicates the SAML client identifier provided by the SAML Administrator for the Role and Attributes information, while creating the SAML application for FCC Studio.</p>   | <p>Yes</p> | <p>Yes</p> | <p>Yes</p> |

|   |   |     |     |     |
|---|---|-----|-----|-----|
| SAML_LOGOUT_URL   | Indicates the SAML client identifier provided by the SAML Administrator for the Logout URL information when creating the SAML application for FCC Studio. | Yes | Yes | Yes |
| SAML_COOKIE_DOMAIN  | Indicates the SAML client identifier provided by the SAML Administrator for the Logout URL information when creating the SAML application for FCC Studio. | Yes | Yes | Yes |
| FCC_API_USERS   | Indicates the API users.  | Yes | Yes | Yes |
| IDCS<br>NOTE:<br>The IDCS related parameters are applicable only if IdcsRealm is used in the Realm parameter. |   |     |     |     |
| IDCS_HOST   | Indicates the server of the Oracle Identity Cloud Service (IDCS) instance.  | Yes | Yes | Yes |
| IDCS_PORT   | Indicates the port number of the IDCS instance.   | Yes | Yes | Yes |
| IDCS_SSL_ENABLED  | Indicates if SSL is enabled for the IDCS application.<br>Default value: true  | Yes | Yes | Yes |

|                    |   |                 |                 |          |
|--------------------|---|-----------------|-----------------|----------|
| LOGOUT_URL         | Indicates the URL to redirect after logout from FCC Studio.   | Yes             | Yes             | Yes      |
| IDCS_TENANT        | Indicates the IDCS tenant provided by the IDCS Administrator while creating the IDCS application for FCC Studio.  | Yes             | Yes             | Yes      |
| IDCS_CLIENT_ID     | Indicates the IDCS client identifier provided by the IDCS Administrator while creating the IDCS application for FCC Studio.   | Yes             | Yes             | Yes      |
| IDCS_CLIENT_SECRET | Indicates the IDCS client secret provided by the IDCS Administrator while creating the IDCS application for FCC Studio.   | Yes             | Yes             | Yes      |
| FCDM_SOURCE        | <p>Indicates the source database for FCC Studio.</p> <p>The available options are ECM or BD.</p> <p><b>NOTE:</b></p> <ul style="list-style-type: none"><li>FCC Studio can use either the BD or ECM schema as the source of FCDM data for the graph.</li><li>Ensure to enter the value as ECM whenever ECM integration is required with Investigation Hub.</li></ul> <p>Here, ECM schema is used as the source of the FCDM data to load the case information into the graph.</p> | Enter BD or ECM | Enter BD or ECM | Enter NA |

|   |   |                     |                     |          |
|---|---|---------------------|---------------------|----------|
| Graph Settings  |   |                     |                     |          |
| CB_CONFIGURED   | Indicates the setting of the graph edges. When the corresponding edges of the graph is required, set the value to <code>true</code> . | Enter true or false | Enter true or false | Enter NA |
| <b>Keystore file and pass details for batch service</b> |   |                     |                     |          |
| KEYSTORE_FILE_NAME                                      | Indicates the keystore file name used for secure batch service.<br>For example: <code>batch_keystore.jks</code>                       | Yes                 | Yes                 | Yes      |
| KEYSTORE_PASS   | Indicates the keystore password used for the secure batch service.  | Yes                 | Yes                 | Yes      |
| KEYSTORE_ALIAS  | Indicates the keystore alias name used for the secure batch service.<br>For example: <code>batchservice</code>                        | Yes                 | Yes                 | Yes      |
| STUDIO_SERVER_SSL_PASSWORD                              | Indicates the SSL password that is defined by the user.   | Yes                 | Yes                 | Yes      |
| STUDIO_SERVER_SSL_ALIASES                               | Indicates the SSL alias that is provided by the user  | Yes                 | Yes                 | Yes      |

|                                    |   |     |     |     |
|------------------------------------|---|-----|-----|-----|
| NUM_CACHED_RESULTSET               | Indicates the cached result set.<br>For ex: 0<br>CONFIGURE THE CONFIG.SH FILE<br>OFS CRIME AND COMPLIANCE<br>STUDIO INSTALLATION GUIDE   58 | Yes | Yes | No  |
| RESULTSET_EXPIRATION_TIME_SECS     | Indicates the Result set expiration time.<br>For ex: 3600.  | Yes | Yes | No  |
| MAX_TOTAL_SHARED_DATA_MEMORY_SIZE  | Absolute memory limit of shared data (includes published graphs and pinned non-referenced graphs).<br>For example: 20G                      | Yes | Yes | Yes |
| MAX_TOTAL_PRIVATE_DATA_MEMORY_SIZE | Memory limit of private data (includes non-published graphs and PGQL results) relative to total PGX engine memory limit.<br>For example: 8G | Yes | Yes | No  |
| MAX_PER_SESSION_DATA_MEMORY_SIZE   | Absolute memory limit for any one session of the PGX engine.<br>For example: 700M   | Yes | Yes | No  |
| MAX_DATA_MEMORY_SIZE_DSADMIN       | Absolute memory limit for any user of the PGX engine for the DSADMIN role.<br>For example: 2G   | Yes | Yes | No  |
| MAX_DATA_MEMORY_SIZE_DSBATCH       | Absolute memory limit for any user of the PGX engine for the DSBATCH role.<br>For example: 10G  | Yes | Yes | No  |



|                                 |  |     |     |    |
|---------------------------------|--|-----|-----|----|
| MAX_DATA_MEMORY_SIZE_DSINTER    | Absolute memory limit for any user of the PGX engine for the DSINTER role.<br>For example: 5G    | Yes | Yes | No |
| MAX_DATA_MEMORY_SIZE_DSAPPROVER | Absolute memory limit for any user of the PGX engine for the DSAPPROVER role.<br>For example: 5G | Yes | Yes | No |
| MAX_DATA_MEMORY_SIZE_DSUSER     | Absolute memory limit for any user of the PGX engine for the DSUSER role.<br>For example: 5G     | Yes | Yes | No |

## 4.10 Configure the Extract Transfer and Load (ETL) Process

Extract Transfer and Load (ETL) is the procedure of copying data from one or more sources into a destination system which represents the data differently from the source or in a different context than the source. Data movement and graph loading is performed using ETL.

**NOTE** In case you have 8.0.7.4.0 installed and the spark cluster has both `batchservice-8.0.7.*.0.jar` and `elasticsearch-spark-20_2.11-7.* jar` files installed, you must remove them from the spark class path.

To configure the Data Movement and Graph Load, copy the `FCCM_Studio_SqoopJob.sh`, `FCCM_Studio_ETL_Graph.sh`, `FCCM_Studio_ETL_Connector.sh`, and `FCCM_Studio_ETL_BulkSimilarityEdgeGeneration.sh` files from the `<Studio_Installed_Path>/out/ficdb/bin` directory and add in the `<FIC_HOME of OFSAA_Installed_Path>/ficdb/bin` directory. For information on performing Data Movement and Graph Load, see the **Configure ETL** and **Execute ETL** sections in the *OFS Crime and Compliance Studio Administration Guide*.

**NOTE** Before you run the sqoop job, ensure that the `serverconfig.properties` file from the `<Studio_Installed_Path>/batchservice/conf` directory has the correct values.

### 4.10.1 Configure the Extract Transfer Load (ETL) Services

To configure the ETL services, follow these steps:

1. Place the Hadoop Cluster files in the `<Studio_Installation_Path>/configmaps/spark-interpreter-conf` directory. For more information on the file structure, see [Required File Structure](#).
2. Place the Kerberos files in the `<Studio_Installation_Path>/configmaps/batchservice/user/conf/` directory. For more information on the file structure, see [Required File Structure](#).
3. Place the following jars in the `<Studio_Installation_Path>/docker/user/batchservice/lib/` directory. For more information about jars, see [Appendix D – Additional Jars – Batch Service](#).

**NOTE**

- The version of the jars are client/user-specific. These jars can be obtained from existing jars of Cloudera installation.
- The `HiveJDBC4.jar` file is not available in the Cloudera setup. You must download the same from the Cloudera website.

4. Configure the `config.sh` file in `<Studio_Installation_Path>/bin` directory to replace the placeholder values in the applicable files in the `configmaps` directory as described in the following table:

**NOTE**

Do not alter the parameter values that are already set in the `config.sh` file

**Table 13:Configuring config.sh File**

| Parameter                       | Description   |
|---------------------------------|---|
| <b>Deployment Configuration</b> |   |
| NAMESPACE                       | Enter a value to create a namespace with the specified value.<br>For example: fccs  |
| <b>PGX Service</b>              |   |
| PGX_SERVER_NUM_REPLICAS         | Indicates the number of replicas of the PGX server.<br>For example: 1   |
| PGX_GLOBAL_GRAPH_NAME           | Indicates the name that the pre-loaded global graph is published with and the FCC Studio users can use to reference the global graph.<br>For example: GlobalGraphIH |

|                                    |  |
|------------------------------------|--|
| URL_GLOBAL_GRAPH_CONFIG_JSON       | Indicates the HDFS URL where the PGX graph configuration .json file is stored at the end of the ETL. The location can be either local or hdfs path.<br>For example:<br>hdfs:///user/fccstudio/graph.json |
| HDFS_GRAPH_FILES_PATH              | Indicates the filepath in the HDFS where the graph.json is formed.   |
| FCC_STUDIO_URL                     | Indicates the FCC Studio URL that starts with HTTPS.   |
| MAX_TOTAL_SHARED_DATA_MEMORY_SIZE  | Absolute memory limit of shared data (includes published graphs and pinned non-referenced graphs).<br>For example: 20G   |
| MAX_TOTAL_PRIVATE_DATA_MEMORY_SIZE | Memory limit of private data (includes non-published graphs and PGQL results) relative to total PGX engine memory limit.<br>For example: 8G  |
| MAX_PER_SESSION_DATA_MEMORY_SIZE   | Absolute memory limit for any one session of the PGX engine.<br>For example: 700M  |
| MAX_DATA_MEMORY_SIZE_DSADMIN       | Absolute memory limit for any user of the PGX engine for the DSADMIN role.<br>For example: 2G  |
| MAX_DATA_MEMORY_SIZE_DSBATCH       | Absolute memory limit for any user of the PGX engine for the DSBATCH role.<br>For example: 10G   |
| MAX_DATA_MEMORY_SIZE_DSINTER       | Absolute memory limit for any user of the PGX engine for the DSINTER role.<br>For example: 5G  |
| MAX_DATA_MEMORY_SIZE_DSAPPROVER    | Absolute memory limit for any user of the PGX engine for the DSAPPROVER role.<br>For example: 5G   |
| MAX_DATA_MEMORY_SIZE_DSUSER        | Absolute memory limit for any user of the PGX engine for the DSUSER role.<br>For example: 5G   |

|                                   |  |
|-----------------------------------|--|
| <b>Quantifind Details</b>         |  |
| QUANTIFIND_URL                    | Indicates the URL of the Quantifind.<br>For example, https://api-test.quantifind.com   |
| ENCRYPTED_QUANTIFIND_TOKEN        | Indicates the token that is generated when integrating with Quantifind.<br>For example,<br>c2FtcGxIX2VuY3J5cHRIZF9xdWFudGhmaW5kX3Rva2Vu. |
| QUANTIFIND_APPNAME                | Indicates the Quantifind App Name.<br>For example, OracleIntegrationTest.  |
| QUANTIFIND_ENABLED                | Indicates that Quantifind is enabled. Options are True or False.   |
| <b>SSL Password</b>               |  |
| STUDIO_SERVER_SSL_PASSWORD        | Indicates the Studio Server SSL password.  |
| <b>Matching Service</b>           |  |
| ELASTIC_SEARCH_HOSTNAME           | Indicates the Host name used for Elastic search.   |
| ELASTIC_SEARCH_PORT               | Indicates the search port of Elastic search.   |
| ELASTIC_SEARCH_USERNAME           | Username of Elastic search if Es is authenticated.   |
| ELASTIC_SEARCH_ENCRYPTED_PASSWORD | Elastic Search Encrypted password.   |
| ELASTIC_SEARCH_HTTPS_ENABLED      | Is Es is https enabled Value : true false.   |
| STUDIO_ALIAS_NAME                 | Indicates the alias name for Studio used for the matching service.   |

|                                  |   |
|----------------------------------|---|
| EXECUTOR_THREADS                 | Indicates the executor thread details. The limit is 4.  |
| ELASTICRESPONSE_BUFFERLIMIT_BYTE | Indicates the elastic response details. The limit is 4. |
| SCROLL_TIME                      | Indicates the scroll time details. The limit is 30.     |
| SCROLL_SIZE                      | Indicates the scroll size details. The limit is 500.    |
| BD_ATOMIC_ALIAS_NAME             | This indicated the alias name of the Atomic schema.     |

- Grant Execute permission to the `<Studio_Installation_Path>/bin` directory using the following command:

```
chmod 755 install.sh config.sh
```

- Run the following command:

```
./install.sh
```

#### NOTE

- Execution of the `install.sh` command does not generate any log file.
- The values for the `<URL_GLOBAL_GRAPH_CONFIG_JSON>` and `<PGX_GLOBAL_GRAPH_NAME>` parameters in the `<Studio_Installation_Path>/configmaps/pgx-server/pgx.conf` file are auto-populated with the values that are configured in the `<Studio_Installation_Path>/bin/config.sh` file.

### 4.10.1.1 Loading Graphs

Loading graphs to FCC Studio can be based on the following scenarios:

- [Loading sample graph without running ETL](#)
- [Loading the graph generated from ETL](#)

#### 4.10.1.1.1 Loading sample graph without running ETL

To load sample graph without running ETL, perform the following steps:

- Navigate to the `<Studio_Installation_Path>/configmaps/pgx-server/` directory and modify the `pgx.conf` file for the highlighted parameters:

```
"preload_graphs": [
```

```

    {
      "path": "/data/graph/sample-graph.json",
      "name": "GlobalGraphIH",
      "publish": false,
      "publish_with_snapshots": true
    }
  ],
  "pgx_realm": {
    "implementation":
"com.oracle.ofss.fccm.studio.pgx.FCCMPgxRealm"
  },
  "file_locations": [
    {
      "name": "hdfs_storage",
      "location": "/data/graph"
    }
  ],

```

#### 4.10.1.1.2 Loading the graph generated from ETL

You can load graph generated from **ETL**, based on the following:

- **PGX** fails, until you have a graph generated from **ETL** on the **HDFS**. Once the graph is generated, and then, as soon as the **PGX** server pods restart, the graph is pre-loaded to the Studio.
- **PGX** starts at the time of deployment, and user has to re-deploy Studio with the **PGX** related configuration, once you have a graph generated from **ETL** on the **HDFS**.
  - Create a backup of `pgx.conf`, the backup is can be used when the graph is generated from ETL.
  - At the time of deployment, you must deleted the following lines from `pgx.conf` file.

```

    "preload_graphs": [
      {
        "path": "##URL_GLOBAL_GRAPH_CONFIG_JSON##",
        "name": "##PGX_GLOBAL_GRAPH_NAME##",
        "publish": false,
        "publish_with_snapshots": true
      }
    ],

```

- The following lines must be deleted multiple times.

```

    {
        "preloaded_graph":
        "##PGX_GLOBAL_GRAPH_NAME##",
        "grant": "manage"
    }

```

- Proceed with the studio deployment.
- Once the graph is generated, perform the following:
  - Replace the existing `pgx.conf` file with the backup `pgx.conf` file
- Restart Studio.

## 4.11 Verify the Resource Allocation for FCC Studio Services

The required resources must be allocated to the FCC Studio services as per the architecture.

### Topics:

- [Resource Limits](#)
- [Resource Types](#)
- [Resource Parameters in FCC Studio](#)

## 4.12 Resource Limits

For FCC Studio to run reliably, the available resources of the Kubernetes cluster must be allocated accordingly. The components are memory-intensive and therefore it is recommended to set memory constraints for each component.

## 4.13 Resource Types

Each container requires a memory request and memory limit size as defined by the Kubernetes API. In short, containers specify a request, which is the amount of that resource that the system will guarantee to the container and a limit which is the maximum amount that the system will allow the container to use. For more information on troubleshooting tips, see [Managing Compute Resources for Containers](#).

Some components require additional resource limits which are set as environment variables.

## 4.14 Resource Parameters in FCC Studio

After extracting the FCC Studio application installer software, the resource limits must be adjusted for each component. The configuration files can be found in the `<Studio_Installation_Path>` directory.



**NOTE**

- The sizing recommendations are preliminary. In the case of deployment failures, a manual configuration of the sizing parameters is required.
- Depending on the use case, the recommended value changes.
- The default value in the following table is the value that is already set in the file.
- Make sure to give appropriate memory to python pod in K8s deployment to avoid out of memory issue.

**Table 14: Resource Parameters in FCC Studio**

| Configuration File/Container | Parameter type | Parameter Name                              | Description   | Recommendation |
|------------------------------|----------------|---|---|----------------|
| server.yml / server          | k8             | spec.containers[].resources.requests.memory | Memory request size for the FCC server (web application) component.     | default        |
|                              | k8             | spec.containers[].resources.requests.memory | Memory limit size for the FCC server (web application) component.       | default        |
| agent.yml / agent            | k8             | spec.containers[].resources.requests.memory | Memory request size for the Agent (manages all interpreters) component. | default        |

|                             |                     |   |   |   |
|-----------------------------|---------------------|---|---|---|
|                             | k8                  | spec.containers[].resources.limits.memory   | Memory limit size for the Agent (manages all interpreters) component.   | default   |
| pgx-server.yml / pgx-server | k8                  | spec.containers[].resources.requests.memory | Memory request size for the PGX server (manages graph processing) component.  | Slightly less than the memory of the PGX server as calculated in the sizing guide.  |
|                             | k8                  | spec.containers[].resources.requests.memory | Memory limit size for the PGX server (manages graph processing) component.  | The same as the request size above.   |
|                             | ENV VAR (JAVA_OPTS) | -Xmx<br>-Xms                                | The maximum and minimum heap memory size (mainly used for storing graphs' string properties) for the Java process of PGX. | 58% of the container's memory limit size above.<br><br>For a better understanding of this sizing parameter, see the <a href="#">PGX 20.0.2 Memory Consumption documentation</a> . |

|                               |                     |   |  |   |
|-------------------------------|---------------------|---|--|---|
|                               | ENV VAR (JAVA_OPTS) | -Dpgx.max_off_heap_size                     | The maximum off-heap memory size in megabytes (mainly used for storing graphs except for their string properties) that PGX tries to respect. | 42% of the container's memory limit size above.<br><br>For a better understanding of this sizing parameter, see the <a href="#">PGX 20.0.2 Memory Consumption documentation</a> . |
| fcc-pgx.yml / pgx-interpreter | k8                  | spec.containers[].resources.requests.memory | Memory request size for the PGX interpreter.   | 4Gi   |
|                               | k8                  | spec.containers[].resources.limits.memory   | Memory limit size for the PGX interpreter.   | 16Gi<br><br>Sizing should depend on the number and behavior (memory requirements of sessions) of concurrent users   |
| authservice.yml / authservice | k8                  | spec.containers[].resources.requests.memory | Memory request size for the authservice (used for getting roles of a user from DB) component.  | default   |

|                               |    |   |   |         |
|-------------------------------|----|---|---|---------|
|                               | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the authservice (used for getting roles of a user from DB) component.   | default |
| metaservice.yml / metaservice | k8 | spec.containers[].resources.requests.memory | Memory request size for the metaservice (used for custom interpreter api's like loaddataset, listdataset in scala interpreter etc.) component.      | default |
|                               | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the metaservice (used for custom interpreter APIs such as loaddataset, listdataset in Scala interpreter and so on) component. | default |

|                                      |    |   |  |   |
|--------------------------------------|----|---|--|---|
| sessionservice.yml / session-service | k8 | spec.containers[].resources.requests.memory | Memory request size for the session-service (used for managing session between PGX and Scala interpreter) component.                     | default   |
|                                      | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the session-service (used for managing session between PGX and Scala interpreter) component.                       | default   |
| batchservice.yml / batch-service     | k8 | spec.containers[].resources.requests.memory | Memory request size for the batch-service (used for managing batches like sqoopjob, graph load, notebook execution and so on) component. | 1 Gi as request<br><br><b>Note:</b> This depends on the amount of data that you process in the ETL. |

|   |    |   |   |  |
|---|----|---|---|--|
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the batchservice (used for managing batches like sqoopjob, graph load, notebook execution and so on) component. | 2 Gi as request<br><br><b>Note:</b> This depends on the amount of data that you process in the ETL.  |
| entity-resolution.yml/<br>entity resolution | k8 | spec.containers[].resources.requests.memory | Memory request size for the Entity Resolution component.  | By default, 4.   |
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the Entity Resolution component.  | By default, 8.   |
| matching-service.yml/<br>matching service   | k8 | spec.containers[].resources.requests.memory | Memory request size for the Matching Service component.   | <ul style="list-style-type: none"> <li>memory: 6Gi</li> <li>cpu: 3</li> </ul> <b>Note:</b> This depends on the size of the data that you are processing    |
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the Matching Service component.   | <ul style="list-style-type: none"> <li>memory: 12Gi</li> <li>cpu: "6"</li> </ul> <b>Note:</b> This depends on the size of the data that you are processing |

|   |    |   |  |         |
|---|----|---|--|---------|
| spark.yml/spark and pyspark Interpreter                                     | k8 | spec.containers[].resources.requests.memory | Memory request size for the Spark interpreter.                         | default |
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the Spark interpreter.                           | default |
| fcc-jdbc.yml / fcc-jdbc   | k8 | spec.containers[].resources.requests.memory | Memory request size for the JDBC connection.                           | default |
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the JDBC connection.                             | default |
| fcc-livy.yml / fcc-spark-scala, fcc-spark-sql, and fcc-pyspark interpreters | k8 | spec.containers[].resources.requests.memory | Memory request size for the livy connection to big data Spark cluster. | default |
|   | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the livy connection to Big data Spark cluster.   | default |
| fcc-markdown.yml / markdown-interpreter                                     | k8 | spec.containers[].resources.requests.memory | Memory request size for the Markdown interpreter.                      | default |

|                                     |    |   |   |                       |
|-------------------------------------|----|---|---|-----------------------|
|                                     | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the Markdown interpreter. | default               |
| fcc-ore.yml / ore-interpreter       | k8 | spec.containers[].resources.requests.memory | Memory request size for the ORE connection.     | default               |
|                                     | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the ORE connection.       | default               |
| fcc-python.yml / python-interpreter | k8 | spec.containers[].resources.requests.memory | Memory request size for the Python interpreter. | depending on use case |
|                                     | k8 | spec.containers[].resources.limits.memory   | Memory limit size for the Python interpreter.   | depending on use case |

## 4.15 Deploy FCC Studio on the K8s Cluster

To deploy FCC Studio on the K8s cluster, follow these steps:

1. Navigate to the `<Studio_Installation_Path>` directory.
2. Execute the following command:

```
./fcc-studio.sh --registry <registry URL>:<registry port>
```

### NOTE

Execute the `./fcc-studio.sh -h` command for usage instructions.

Congratulations! Your installation is complete.



After successful completion, the script displays a URL that can be used to access the FCC Studio Application. For more information, see [Access the FCC Studio Application](#).



## 5 Post-installation Steps when OFSAA is Installed

On successful installation of FCC Studio, you must perform the following post-installation configurations.

### Topics:

- [Verify the Installation](#)
- [Checking Logs of FCC Studio Services](#)
- [Configure the SSH Connection](#)
- [Configure Schema Creation](#)
- [Configure the ICIJ Data](#)
- [Restart the PGX Service](#)
- [Access the FCC Studio Application](#)
- [Perform the OFSAA Configuration for Batch Execution](#)
- [Configure and Run Published Notebooks](#)

### 5.1 Verify the Installation

1. To verify the FCC Studio deployment, wait for a minimum of 10 minutes after completing the execution of the `./fcc-studio.sh` command, and run the `kubectl get pods -n <Namespace>` command.

The pod details are displayed to indicate the status of the services. You can also check the logs of the FCC Studio services from the Kubernetes Dashboard.

#### NOTE

- For FCC Studio deployed with OFSAA, ensure all the pods are ready before accessing the FCC Studio application.
- For FCC Studio deployed with non-OFSAA, ensure the metaservice is up and running before accessing the FCC Studio application.

#### 5.1.1 Check Logs of FCC Studio Services

The Kubernetes UI enables you to view the logs of the services installed as part of FCC Studio. To check the logs of the FCC Studio services from the Kubernetes dashboard, follow these steps:

1. Access the Kubernetes Dashboard. The Kubernetes Dashboard login page is displayed.

Figure 5:Kubernetes Dashboard

**Kubernetes Dashboard**

Kubeconfig  
Please select the kubeconfig file that you have created to configure access to the cluster. To find out more about how to configure and use kubeconfig file, please refer to the [Configure Access to Multiple Clusters](#) section.

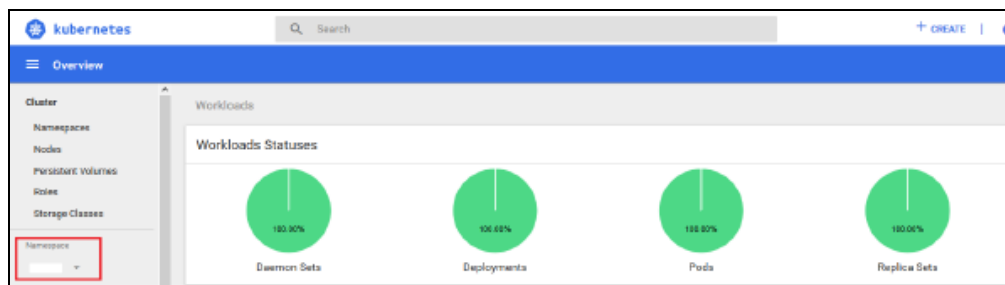
Token  
Every Service Account has a Secret with valid Bearer Token that can be used to log in to Dashboard. To find out more about how to configure and use Bearer Tokens, please refer to the [Authentication](#) section.

Enter token

**SIGN IN**

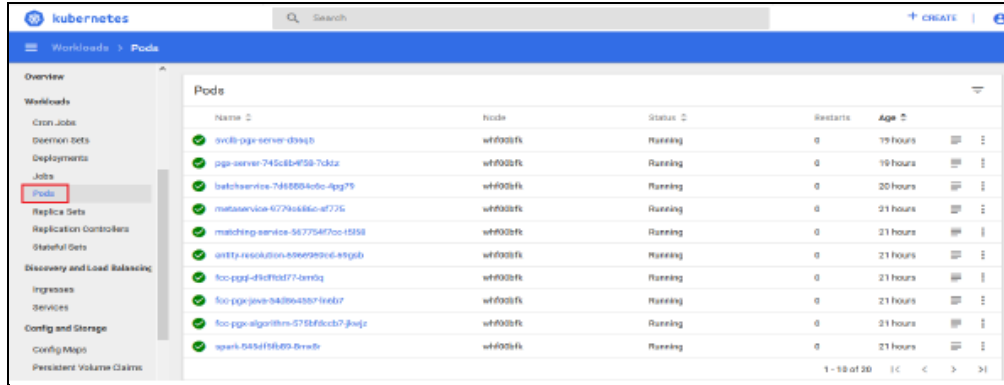
2. Select **Token** and enter the Admin user secret token in the Enter Token field.
3. Click **SIGN IN**. The Kubernetes Dashboard page is displayed.
4. Select <Namespace> from the Namespace drop-down list on the menu items displayed on the LHS.

Figure 6:Kubernetes Dashboard Overview



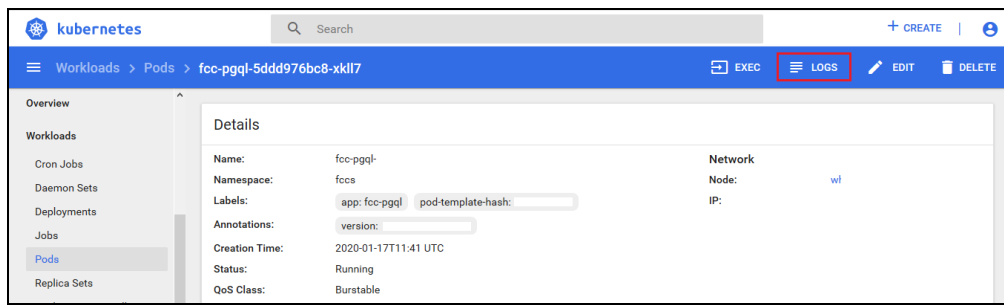
5. Navigate to Workloads > Pods from the menu items displayed on the LHS. The Pods page is displayed with the details of all the services installed as part of the FCC Studio installation.

Figure 7:Kubernetes Dashboard Pods



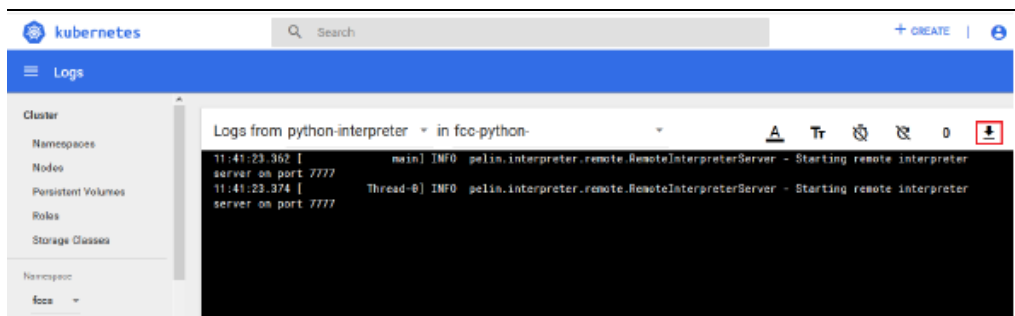
6. Click the service name from the Name column. The service details are displayed.
7. Click the Logs tab.

Figure 8:Kubernetes Dashboard Logs



The service logs are displayed and you can download the logs by clicking Download Logs icon.

Figure 9:Sample Log



**WARNING**

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

If the installation of FCC Studio is unsuccessful, you must reinstall the application after performing the cleanup tasks. For more information, see [Reinstall FCC Studio](#).

## 5.2 Configure the Schema Creation

Schema creation is a one- time activity that replicates the table structure from the Financial Crime Data Model (FCDM) Atomic schema to the Hive Atomic schema.

### Topics:

- [Configure the Schema Creation from the FCC Studio Server](#)

### 5.2.1 Configure the Schema Creation from the FCC Studio Server

To configure Schema Creation from FCC Studio server, follow these steps:

1. Set `FIC_DB_HOME` path to `<Studio_Installation_Path>/out/ficdb`.

**NOTE** The `$FIC_DB_HOME` path can be set from the `.profile` file.

2. Create a Hive Schema with the name mentioned in the `HIVE_SCHEMA` parameter in the `studio-env.yml` file.

For information on the `studio-env.yml` file, see [Configure the studio-env.yml File](#).

3. Execute the following shell script in the `<Studio_Installation_Path>/out/ficdb/bin/` directory to create tables in Hive Schema:

```
./FCCM_Studio_SchemaCreation.sh HIVE
```

4. Check the batch service logs in the Kubernetes Dashboard.
5. Check the hive schema for table creation, if the table is not created, then follow these steps:

- a. Login to the atomic schema of BD/ECM and run the following query:

```
select * from fcc_orahive_datatypemapping;
```

The `fcc_orahive_datatypemapping` table must have only 5 rows. If there are more than 5 rows, run the following query to delete the additional rows:

```
select * from fcc_orahive_datatypemapping for update
```

- b. 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`.

- c. Repeat the step no 3.

## 5.3 Configure the ICIJ Data

To configure the International Consortium of Investigative Journalists (ICIJ) data, you must configure the file path. To configure the file path, follow these steps:

### NOTE

- The FCC Studio graph model is configured to include ICIJ watchlist files.
- Only applicable if customer wants ICIJ data in the graph generated by ETL.

1. Place the watchlist files in HDFS, that is accessible by the user.
2. Update the `FILEPATH` of the watch list files in the `fcc_studio_etl_files` table. The `FILEPATH` refers to the complete hdfs filepath of the csv file. For mapping between `DF_NAME` and `FILEPATH`, see the following image:

**Figure 10: fcc\_studio\_etl\_files Table**

|   | DF_NAME                      | FILEPATH | DF_SEQ_NO | FILE_ORDER |
|---|------------------------------|----------|-----------|------------|
| 1 | Offshore_edges_is_related_to | ...      | 12        | 1          |
| 2 | Bahama_External_Address      | ...      | 13        | 1          |

The following table provides the file path for the csv files:

**Table 15: Path of CSV Files**

| DF_NAME                    | File Path                            |
|----------------------------|--------------------------------------|
| Panama_External_Ad dress   | panama_papers.nodes.address.csv      |
| Panama_External_Entity     | panama_papers.nodes.entity.csv       |
| Panama_External_Entity     | panama_papers.nodes.intermediary.csv |
| Panama_External_Entity     | panama_papers.nodes.officer.csv      |
| Panama_edges_address_of    | panama_papers.edges.csv              |
| Panama_edges_is_related_to | panama_papers.edges.csv              |
| Offshore_External_Address  | offshore_leaks.nodes.address.csv     |
| Offshore_External_Entity   | offshore_leaks.nodes.entity.csv      |

|                              |  |
|------------------------------|--|
| Offshore_External_Entity     | offshore_leaks.nodes.intermediary.csv  |
| Offshore_External_Entity     | offshore_leaks.nodes.officer.csv       |
| Offshore_edges_address_of    | offshore_leaks.edges.csv               |
| Offshore_edges_is_related_to | offshore_leaks.edges.csv               |
| Bahama_External_Address      | bahamas_leaks.nodes.address.csv        |
| Bahama_External_Entity       | bahamas_leaks.nodes.entity.csv         |
| Bahama_External_Entity       | bahamas_leaks.nodes.intermediary.csv   |
| Bahama_External_Entity       | bahamas_leaks.nodes.officer.csv        |
| Bahama_edges_address_of      | bahamas_leaks.edges.csv                |
| Bahama_edges_is_related_to   | bahamas_leaks.edges.csv                |
| Paradise_External_Address    | paradise_papers.nodes.address.csv      |
| Paradise_External_Entity     | paradise_papers.nodes.entity.csv       |
| Paradise_External_Entity     | paradise_papers.nodes.intermediary.csv |
| Paradise_External_Entity     | paradise_papers.nodes.officer.csv      |
| Paradise_External_Entity     | paradise_papers.nodes.other.csv        |
| Paradise_edges_is_related_to | paradise_papers.edges.csv              |
| Paradise_edges_address_of    | paradise_papers.edges.csv              |
| Paradise_edges_is_linked_to  | paradise_papers.edges.csv              |

## 5.4 Restart the PGX Service

You must restart the PGX service to ensure that graphs are created. To restart the PGX service, follow these steps:



**NOTE** Restart the PGX service only after the  
`./FCCM_Studio_ETL_Graph.sh sqoop` job is run.

1. Navigate to the `<Studio_Installed_Path>/out` directory.
2. Start a new putty session and run the following commands:
 

```
kubectl -n fccs delete -f deployments/pgx-server.yml
kubectl -n fccs apply -f deployments/pgx-server.yml
```

## 5.5 Loading Analysis Data to Elastic Search

**NOTE** Before running Bulk execution for the first time, after graph job, you must remove `fcc_idx_lookup*` entries if available from ES screen and need to run `FCCM_Studio_LoadAnalysisData.sh`. It will update the entries in ES screen

To load the synonyms and stop words to elastic search, perform the following steps:

1. Once, the contents are finalised, before running the similarity edge script. In the `ficdb` path, execute the following shell script with no parameters.

```
./FCCM_Studio_LoadAnalysisData.sh
```

## 5.6 Integration with MMG

Financial institutions require models that work on traditional statistical techniques, modern machinelearning 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.

As a Model Management and Governance toolkit, it enables institutions to implement their IT policies while providing flexibility and freedom that Data Scientists and Statistical Modelers desire.

To know more about MMG roles, sandbox creation and publishing, see the [MMG User Documentation](#).

- NOTE**
- To integrate FCC Studio with MMG, you must provide the data source, atomic and config schema parameters of MMG.
  - The MMG config schema's **"NEXTGENEMF\_CONFIG"** table, must be updated with the following parameters.

The **"NEXTGENEMF\_CONFIG"** table details are as follows:

- **DATASTUDIO\_URL:** The host and port details where Studio is running (`https://<servername>:<port no>`).
- **EMFSTUDIO\_SERVICE\_URL:** The host and port details where MMG service is running and accepting request (`https://<ofsaa web host>:<port>/<context>/rest-api/emf-service/v1`)
- **IS\_APPLICATION\_AIF:** Configuration for AIF Application. Specify **N** to use Studio only.
- **IS\_JAVA\_ENGINE:** Configuration for java engine. Specify **Y** to use Studio.
- **IS\_MULTI\_LEVEL\_AUTHORIZATION:** These are the level of authorization for Model User groups. Specify **N** to use Studio. For more information, see [MMG User Documentation](#).

| V_NAME                         | V_VALUE   | V_DESC   |
|--------------------------------|---|--|
| 1 IS_APPLICATION_AIF           | N   | Configuration for AIF Application                                |
| 2 IS_MULTI_LEVEL_AUTHORIZATION | N   | Level of Authorization for Model User groups                     |
| 3 IS_JAVA_ENGINE               | Y   | Configuration for java engine.                                   |
| 4 DATASTUDIO_URL               | <code>https://&lt;hostname&gt;:&lt;port no&gt;</code>   | Host and port where MMG Studio is running                        |
| 5 EMFSTUDIO_SERVICE_URL        | <code>http(s)://&lt;ofsaa web host&gt;:&lt;port&gt;/&lt;context&gt;/rest-api/emf-service</code> | Host and port where MMG service is running and accepting request |

### 5.6.1 Placing the Files for MMG

To place the files for MMG integration, perform the following steps:

1. Copy the following jars from **MMG tomcat, Weblogic, or Webserver** `<WEB-INF>/lib` directory.
  - `aai-client.jar`
  - `aai-common-client.jar`
  - `aai-comonlanding-service.jar`
  - `aai-common-service.jar`
  - `aai-core.jar`
  - `AESCryptor.jar`
  - `FICServer.jar`
2. Place the jars in `<Studio_Installation_Path>/docker/user/authservice` directory.
3. Copy following files from the `../FIC_HOME/conf` directory.
  - `Dynamic.dtd`
  - `Lookup.dtd`
  - `AESCryptKey.ext`
  - `keypair.store`
  - `keystore.properties`
  - `ofsaa8xskey.ext`
  - `LookupServices.xml`

- `DynamicServices.xml`
4. Place the files in this path `<Studio_Installation_Path>/secrets/mmg-conf` directory. Perform your Installation.

## 5.6.2 Integration with MMG (SAML)

To integration with MMG using SAML, you must perform the following:

1. Apply the patch MMG Patch (*32281016\_GENERIC.zip*) from [My Oracle Support \(MOS\)](#).
2. Specify the following details in the **nextgenemf config** table:
  - **STUDIO\_AUTH\_TYPE**: It is the Data Studio authentication type, in case of SAML, it is **FCC\_SSO**.
  - **FCC\_API\_USER**: Indicates the API User details for FCC Studio, applicable only when **STUDIO\_AUTH\_TYPE** is **FCC\_SSO**.
  - **SSO\_TOKEN**: The token that is generated for accessing MMG. For more information , see [SSO Token Generation](#).

| V_NAME                         | V_VALUE  | V_DESC  |
|--------------------------------|--|---|
| 1 IS_APPLICATION_AIF           | Y  | Configuration for AIF Application   |
| 2 IS_MULTI_LEVEL_AUTHORIZATION | N  | Level of Authorization for Model User groups                              |
| 3 STUDIO_AUTH_TYPE             | DEFAULT  | Data Studio authentication type -DEFAULT/ <b>FCC_SSO</b>                  |
| 4 FCC_API_USER                 | NULL   | API User for FCC Studio. Applicable only when STUDIO_AUTH_TYPE is FCC...  |
| 5 SSO_TOKEN                    | eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTUyYyoiQkFUQ0hfVWVNFU9.FpHhH_5KvRci0V | SSO Token value for Data Studio authentication. Applicable only when S... |
| 6 IS_JAVA_ENGINE               | Y  | Configuration for java engine.  |
| 7 DATASTUDIO_URL               | https://whf00pab.in.oracle.com:30078                                   | Host and port where MMG Studio is running                                 |
| 8 EMFSTUDIO_SERVICE_URL        | http://whf00bii.in.oracle.com:8310/BD808/rest-api/emf-service/v1       | Host and port where MMG service is running and accepting request          |

### 5.6.2.1 SSO Token Generation

To generate the SSO Token, perform the following steps:

1. Navigate to the `<Studio_K8_Installation_Path>/out/ficdb/bin` directory.
2. Export the ficdb contents from the `<Studio_K8_Installation_Path>/out/ficdb` directory.
3. Run the script: `fic_db_home/FCCM_Studio_Generate_APIToken.sh`  
`<FCC_API_USER>`.
 

```
For example, use the ./FCCM_Studio_Generate_APIToken.sh
BATCH_USER.
```
4. The token is generated in the command prompt. The generated SSO Token can be used in the **NEXTGENEMF\_CONFIG** table.

## 5.7 Access the FCC Studio Application

To access FCC Studio, follow these steps:

1. Enter the URL in the following format in the web browser:  
`https://<Host_Name>:<Port_Number>`

Here <Port\_Number> is 30078 for the FCC Studio application installed on kubernetes.

The FCC Studio application login page is displayed.

**Figure 11:FCC Studio Application Login Page**



2. Enter the **Username** and **Password**.
3. Click **Login**.

After you access the application, you can view the ready-to-use notebooks. To check if you have been assigned any roles, create a notebook. If you cannot create a notebook, contact [My Oracle Support \(MOS\)](#).

## 5.8 Perform the OFSAA Configuration for Batch Execution

**NOTE** This configuration is not applicable for FCC Studio installed without OFSAA.

To perform OFSAA configuration for batch execution, follow these steps:

1. Copy the files in the `<Studio_Installation_Path>/ficdb/bin` directory to the server where the BD or ECM pack is installed and to the `$FIC_DB_HOME/bin` directory of the OFSAA setup.
2. Execute the following command to grant Execute permission to the files:  

```
chmod +x <filenames>
```
3. Copy all the files from the `<Studio_Installation_Path>/ficdb/lib` directory into the `$FIC_DB_HOME/lib` directory.

For information on running FCC Studio Batches, see *Managing Studio Batches* chapter in the *Oracle Financial Services Crime and Compliance Studio Administration Guide*.

## 5.9 Configure and Run Published Notebooks

---

**NOTE**

This configuration is not applicable for FCC Studio installed without OFSAA.

To perform the configuration required to run published notebooks, copy the required `FCCM_Studio_NotebookExecution.sh` file from the `<Studio_Installation_Path>/out/ficdb/bin` directory into the `<FIC_HOME of OFSAA_Installed_Path>/ficdb/bin` directory.

For information on running published notebooks, see the *Executing Published Notebook* section in the *OFS Crime and Compliance Studio Administration Guide*.

## 6 Post-installation Steps when OFSAA is Not Installed

On successful installation of FCC Studio, you must perform the following post-installation configurations.

### Topics:

- [Verify the Installation](#)
- [Checking Logs of FCC Studio Services](#)
- [Restart the PGX Service](#)
- [Access the FCC Studio Application](#)

#### NOTE

Before running the post-installation steps, an SSH connection to the Big Data server must be configured

### 6.1 Verify the Installation

1. To verify the FCC Studio deployment, wait for a minimum of 10 minutes after completing the execution of the `./fcc-studio.sh` command, and run the `kubectl get pods -n <Namespace>` command.

The pod details are displayed to indicate the status of the services. You can also check the logs of the FCC Studio services from the Kubernetes Dashboard.

#### NOTE

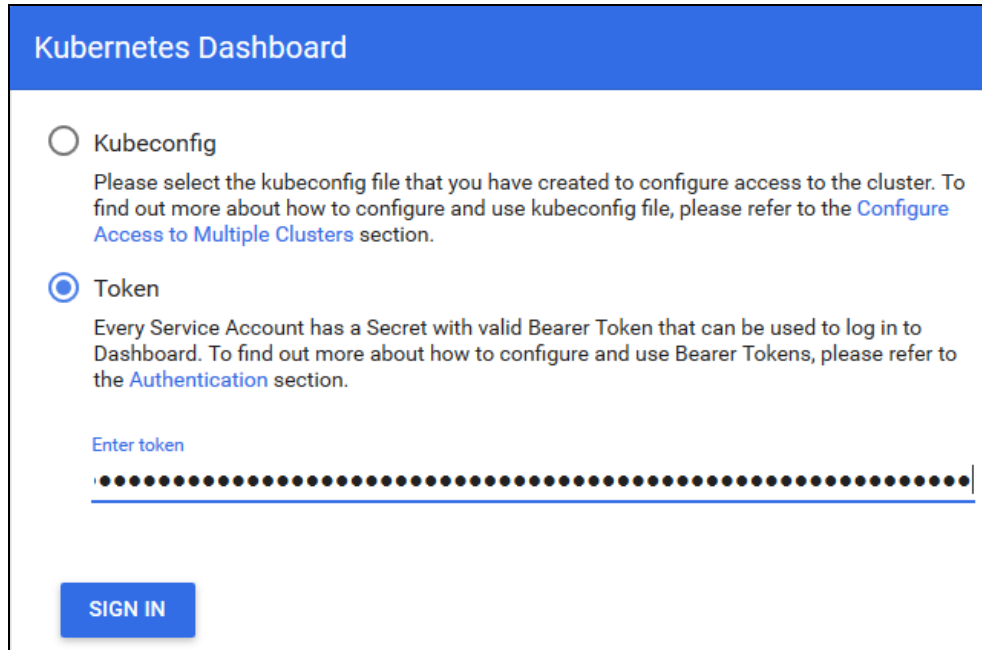
- For FCC Studio deployed with OFSAA, ensure all the pods are ready before accessing the FCC Studio application.
- For FCC Studio deployed with non-OFSAA, ensure the metaservice is up and running before accessing the FCC Studio application.
- PGX Server pod does not start, until ETL process is completed.

#### 6.1.1 Checking Logs of FCC Studio Services

The Kubernetes UI enables you to view the logs of the services installed as part of FCC Studio. To check the logs of the FCC Studio services from the Kubernetes dashboard, follow these steps:

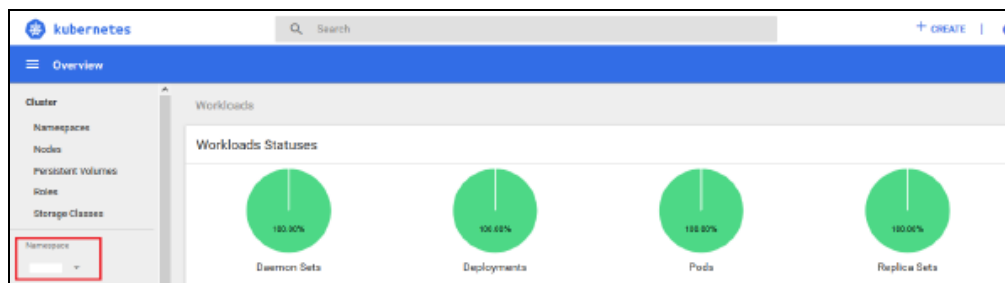
1. Access the Kubernetes Dashboard. The Kubernetes Dashboard login page is displayed.

**Figure 12:Kubernetes Dashboard**



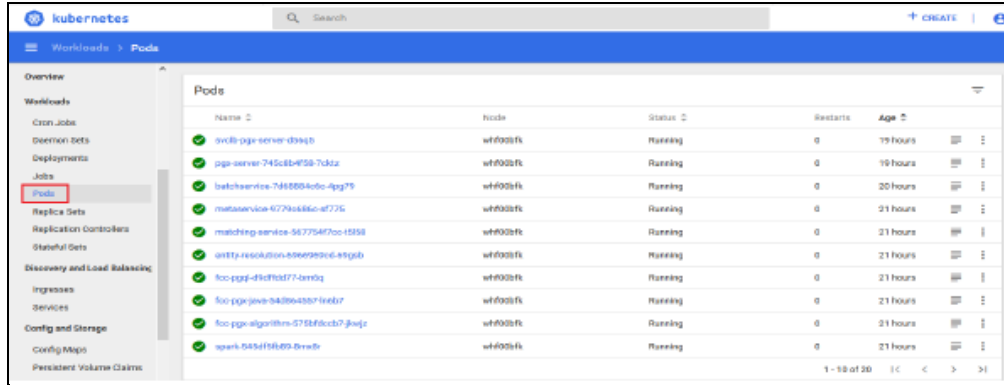
2. Select **Token** and enter the Admin user secret token in the Enter Token field.
3. Click **SIGN IN**. The *Kubernetes Dashboard* page is displayed.
4. Select <Namespace> from the Namespace drop-down list on the menu items displayed on the LHS.

**Figure 13:Kubernetes Dashboard Overview**



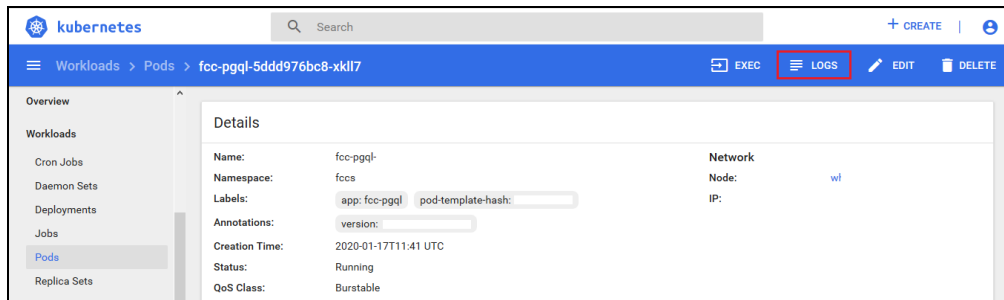
5. Navigate to Workloads > Pods from the menu items displayed on the LHS. The *Pods* page is displayed with the details of all the services installed as part of the FCC Studio installation.

**Figure 14:Kubernetes Dashboard Pods**



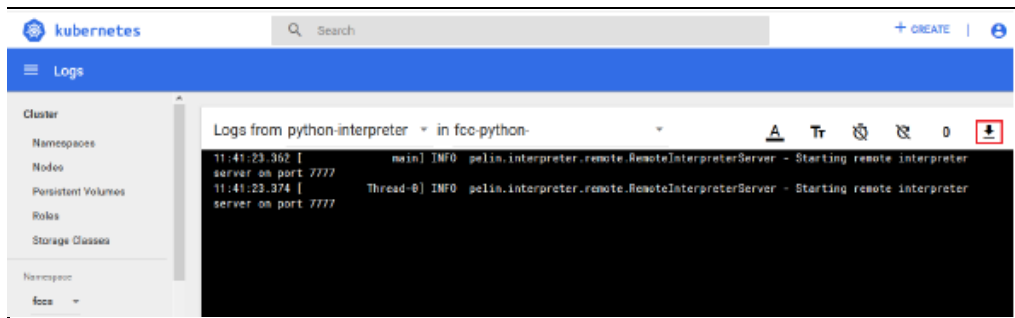
6. Click the service name from the Name column. The service details are displayed.
7. Click the **Logs** tab.

Figure 15:Kubernetes Dashboard Logs



The service logs are displayed and you can download the logs by clicking Download Logs icon.

Figure 16:Sample Log



**WARNING** If you notice any errors in the log files, do not proceed further. Contact [My Oracle Support \(MOS\)](#) and provide the applicable error code and log files.

If the installation of FCC Studio is unsuccessful, you must reinstall the application after performing the cleanup tasks. For more information, see [Reinstall FCC Studio](#).



## 6.2 Restart the PGX Service

You must restart the PGX service to ensure that graphs are created. To restart the PGX service, follow these steps:

**NOTE** Restart the PGX service only after the `./FCCM_Studio_ETL_Graph.sh` graph job is run.

1. Navigate to the `<Studio_Installed_Path>/out` directory.
2. Start a new putty session and run the following commands:  

```
kubectl -n fccs delete -f deployments/pgx-server.yml  
kubectl -n fccs apply -f deployments/pgx-server.yml
```

## 6.3 Access the FCC Studio Application

To access FCC Studio, follow these steps:

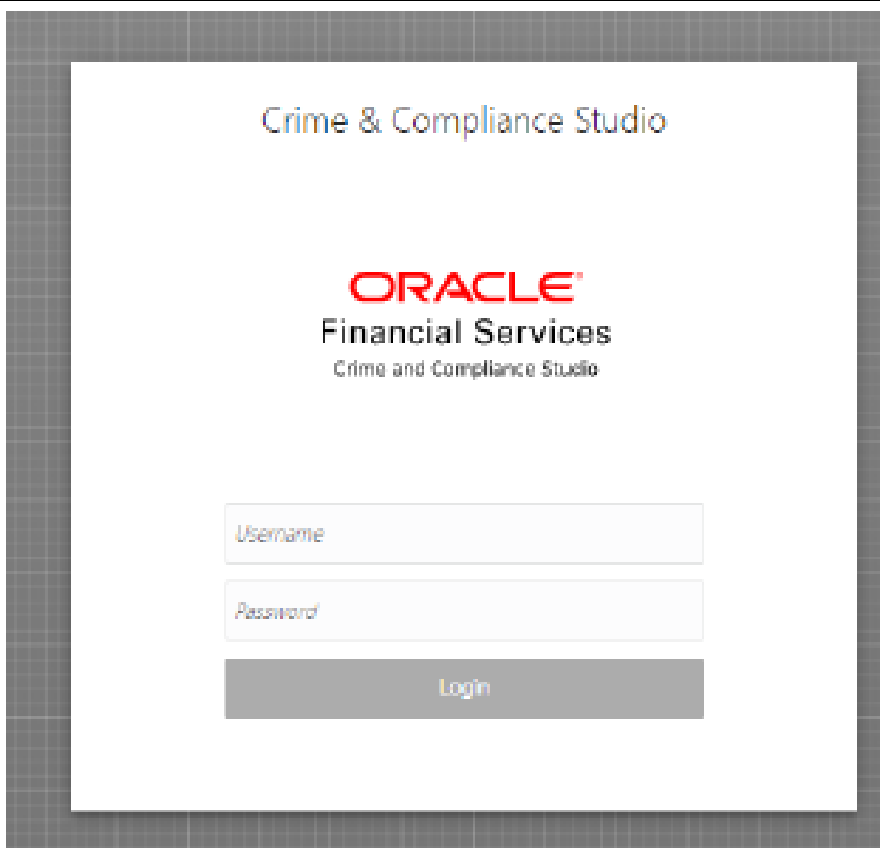
1. Enter the URL in the following format in the web browser:

```
https://<Host_Name>:<Port_Number>
```

Here `<Port_Number>` is 30078 for the FCC Studio application installed on-kubernetes.

The FCC Studio application login page is displayed.

**Figure 17:FCC Studio Application Login Page**



2. Enter the Username and Password.
3. Click Login.

After you access the application, you can view the ready-to-use notebooks. To check if you have been assigned any roles, create a notebook. If you cannot create a notebook, contact [My Oracle Support \(MOS\)](#).

## 6.4 Configuring Infer and Incremental Training API

If you want to use the INFER and Incremental Training API, follow the procedures in the following sections:

- [Prerequisite](#)
- [Configurations](#)

### 6.4.1 Prerequisite

You need to perform the following based on the Python packages installed as mentioned in the table:

| If Python 3.6 (Specific to Python 3.6) is installed  | If Python 3 or any other version installed   |
|--|--|
| 5. Install Python 3.6.<br>6. Navigate to <code>./OFS_FCCM_STUDIO/python-packages/packages</code> directory.<br>7. Verify the list from the packages folder <b>python3.6 -m</b> .<br>8. Navigate to the <code>./OFS_FCCM_STUDIO/python-packages/bin</code> directory.<br>9. Run the <b>Install.sh</b> . The packages are installed. | 10. Install Python3 or any other version.<br>11. Install <b>oracle instantclient (basic)</b> latest version. (If required, login with root unzip in <code>/opt/oracle</code> directory give permissions.)<br>12. Install <b>python3-devel</b> , in case it does not exist.<br>13. Install <b>numpy</b> .<br>Install namepackage whl package ( <code>pip3 install &lt;whl package&gt;</code> ). This is the python packages folder of fcc-studio. |

### 6.4.2 Configurations

**NOTE** The following instructions are applicable if you want to add an incremental model.

After installing the packages, you must configure the **OFSAA URL** in `config.sh`.

To configure the **OFSAA URL**, perform the following steps:

1. Configure the **OFSAA Service URL** to **MMG URL** and **FCDM Source** as **BD** in the `config.sh` in the `./OFS_FCCM_STUDIO/bin` directory.

**NOTE**

- It is a mandatory to provide the **OFSAA Service URL** to **MMG URL** and **FCDM Source** as **BD** in the `config.sh` for using the Training API as an incremental model.

- For incremental model, you need to apply the patch (**80805 One-Off Patch**) to MMG if you have already installed MMG.
- For ground up MMG installation, you need to apply the patch (**80805 One-Off Patch**).

2. After installing Studio with **BD/ECM** with **MMG**, Run the utility for updating link of Notebooks in MMG CONFIG Schema from **MMG**.
3. Navigate to `$FIC_HOME/utility/mmgstudio/utility/bin` directory.
4. Run the `linkUtility.sh` by using the following command in putty.

```
./linkUtility.sh USERNAME
```

Where, **USERNAME** is a ofsaa and data studio user. For example:

```
./linkUtility.sh MMGUSER
```

This updates the notebook link to id in `NEXTGENEMF_MODEL` and `NEXTGENEMF_NOTEBOOK_PARAGRAPHS` tables.

#### NOT E

Make sure that the Link Utility – is added with “iccent.jar” jar file in classpath in `linkUtility.sh` file. That is the line number 28:  
`_CLASSPATH=$JAR_LIST:$FIC_APP_HOME/icc/lib/iccent.jar`

5. Update the sandbox name in the following tables inorder to see the notebooks in MMG UI, whenever the respective sandbox is accessed:
  - Update the **WORKSPACE** column for **Model ID's** with value **9999999999998 (NameMatchingIncrementalTraining)** and **9999999999997 (NameMatchingManualTraining)** in the `nextgenemf_model` table
  - Update the **V\_DEST\_INFODOM** column with infodom of **MMG/BD** for the respective sandbox in the `mf_sandbox_master` table.

Sample query:

```
update mf_sandbox_master set v_dest_infodom='<Infodom of MMG/BD>' where v_sandbox_id='<SANDBOX NAME>'
```

6. Navigating to **MMG Application**, and open the sandbox that you have used as **INFODOM**, then the sandbox displays **NameMatchingTraining** folder. This folder should display the following notebooks:
  - **NameMatchingIncrementalTraining**
  - **NameMatchingManualTraining**

**NOTE**

The **NameMatchingIncrementalTraining** and **NameMatchingManualTraining** notebooks are available and displayed only if the sandbox is created with the same `INFODOM` as the `INFODOM` provided in the Studio's **config.sh** file.

# 7 Upgrade

Follow these steps to upgrade an existing instance of FCC Studio:

1. [Upgrade Scenarios](#)
2. [Upgrade Steps with OFSAA](#)
3. [Upgrade Steps without OFSAA](#)
4. [Configure the studio-env.yml File for Upgrade](#)

You can upgrade an existing instance of FCC Studio as follows:

## 7.1 Upgrade Scenarios

This section provides information about Studio Upgrade. Following are the upgrade scenarios:

- **Upgrade FCC Studio from v8.0.8.0.0 onwards to v8.0.8.x**
  - **With OFSAA.**

**NOTE** Here, ensure to provide the same BD database, Studio schema, Hive schema, wallet related information that you used during the installation of the existing instance FCC Studio.

- **Without OFSAA.**

**NOTE** Here, ensure to provide the same Studio schema and wallet related information that you used during the installation of the existing instance of FCC Studio.

For more information, about cleanup steps that is required for your upgrade, see [Cleanup for Upgrade](#).

## 7.2 Upgrade Steps with OFSAA

The following table provides the steps to upgrade FCC Studio with OFSAA.

**Table 16: Upgrade Steps with OFSAA**

| Sl. No.                       | Activity |
|-------------------------------|----------|
| <b>Pre-installation Steps</b> |          |

|                                |  |
|--------------------------------|--|
| 1                              | <a href="#">Download</a> the installer kit.  |
| <b>Installation Steps</b>      |  |
| 1                              | <a href="#">Extract</a> the installer kit.   |
| 2                              | <a href="#">Place</a> files in the Installation Directories and <a href="#">Place</a> files in wallet. |
| 3                              | <a href="#">Generate</a> an Encrypted Password   |
| 4                              | <a href="#">Generate</a> the Public Key and Private Key  |
| 5                              | <a href="#">Generate</a> the Key Store File for Secure Batch Service                                   |
| 6                              | <a href="#">Add</a> the Batch Service (SSL) to PGX Trust Store   |
| 7                              | <a href="#">Configure</a> the Preferred Services   |
| 8                              | <a href="#">Configure</a> the studio-env.yml File  |
| 9                              | <a href="#">Verify</a> the Resource Allocation for FCC Studio Services                                 |
| 10                             | <a href="#">Deploy</a> FCC Studio on the K8s Cluster   |
| 11                             | <a href="#">Clean up</a> activity if required.   |
| <b>Post-Installation Steps</b> |  |
| 1                              | <a href="#">Verify</a> the Installation  |
| 4                              | <a href="#">Access</a> the FCC Studio Application  |
| 3                              | <a href="#">Upgrade</a> (Replacing Batch service jar for ETL)  |
| 5                              | <a href="#">Perform</a> the OFSAA Configuration for Batch Execution                                    |
| 6                              | <a href="#">Configure</a> and Run Published Notebooks  |

## 7.3 Cleanup for Upgrade

For upgrade from **8.0.8.0.0** to **8.0.8.x**, if you are using **Spark**, **Py-Spark**, and **FCC ORE** interpreters, then you must delete all the entries from the following tables from **Studio Schema** and then **8.0.8.x** must be deployed.

**Table Names** as follows:

- **DS\_INTERPRETER\_ABILITIES**
- **DS\_INTERPRETER\_CREDENTIALS**
- **DS\_INTERPRETER\_PROPS**
- **DS\_INTERPRETER\_VARIANT**

Before you begin the upgrade process, perform the following cleanup steps:

**NOTE** You must delete the previous namespace, before upgrading.

1. Navigate to the `DATABASECHANGELOG` table in the Studio Schema.
2. Run the following SQL statements in the Studio Schema:

```
select * from DatabaseChangeLog a where REGEXP_LIKE
(a.id, 'FCC_DATASTUDIO_CONFIG');
```

```
select * from DatabaseChangeLog a where REGEXP_LIKE
(a.id, 'FCC_STUDIO_DATASTUDIO_CONFIG_808');
```

The entries in the `DATABASECHANGELOG` table that match the criteria are displayed.

3. Delete all the entries generated as the result of above SQL statements command. The entries are as follows:

```
FCC_DATASTUDIO_CONFIG_8.0.7.3.0
FCC_DATASTUDIO_CONFIG_8.0.7
FCC_DATASTUDIO_CONFIG_8.0.7.4.0
FCC_STUDIO_DATASTUDIO_CONFIG_808
```

### 7.3.1 Start the FCC Studio Application

To start the FCC Studio application, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/bin/` directory.
2. Run the following command:

```
./fcc-studio.sh --registry <registry URL>:<registry port>
```

The FCC Studio application is restarted.

Check the logs of the FCC Studio services from the Kubernetes Dashboard. For more information, see [Checking Logs of FCC Studio Services](#).

Once all the services are up and running, the FCC Studio application can be accessed using the following URL:

```
http://<HostName>:30078
```

## 7.4 Upgrade Steps without OFSAA

The following table provides the steps to upgrade FCC Studio without OFSAA.

**Table 17: Upgrade Steps without OFSAA**

| Sl. No.                       | Activity |
|-------------------------------|----------|
| <b>Pre-installation Steps</b> |          |



|                                |  |
|--------------------------------|--|
| 1                              | <a href="#">Download</a> the installer kit.  |
| <b>Installation Steps</b>      |  |
| 1                              | <a href="#">Extract</a> the installer kit.   |
| 2                              | <a href="#">Place</a> files in the Installation Directories and <a href="#">Place</a> files in wallet. |
| 3                              | <a href="#">Generate</a> an Encrypted Password   |
| 4                              | <a href="#">Generate</a> the Public Key and Private Key  |
| 5                              | <a href="#">Generate</a> the Key Store File for Secure Batch Service                                   |
| 6                              | <a href="#">Add</a> the Batch Service (SSL) to PGX Trust Store   |
| 7                              | <a href="#">Configure</a> the Preferred Services   |
| 8                              | <a href="#">Configure</a> the studio-env.yml File  |
| 9                              | <a href="#">Verify</a> the Resource Allocation for FCC Studio Services                                 |
| 10                             | <a href="#">Deploy</a> FCC Studio on the K8s Cluster   |
| <b>Post-Installation Steps</b> |  |
| 1                              | <a href="#">Verify</a> the Installation  |
| 2                              | <a href="#">Access</a> the FCC Studio Application  |
| 3                              | <a href="#">Upgrade</a> (Replacing Batch service jar for ETL)  |

## 7.5 Configure the studio-env.yml File for Upgrade

Configure the `studio-env.yml` file to upgrade FCC Studio with OFSAA. For more information, see [Configure the studio-env.yml File](#).

## 7.6 Post-Installation Steps for Upgrade

For upgrade from 80800 to 808x, perform the following steps:

1. Navigate to the `config.sh` and copy the `GRAPH_FILES_PATH` parameter's value.
2. Navigate to the path from the previous step in big data node.
3. Navigate to `jars` folder.
4. Remove older batchservice jar.
5. Navigate to the `<STUDIO>/ficdb/etlJars` directory.
6. Copy the batch service jar present in this directory into the big data node inside the folder `jars`.

## 8 Reinstall FCC Studio

If the installation of FCC Studio is unsuccessful, you must reinstall the application after performing the required cleanup tasks.

To reinstall FCC Studio, follow these steps:

1. Execute the following command to delete the namespace of FCC Studio:  
`kubectl delete namespace <Namespace>`
2. Navigate to the `<Studio_Installation_Path>` path and correct the parameters or files as suggested by the error.
3. Navigate to `<Studio_Installation_Path>/secrets` directory and re-enter the values of sensitive information in the `studio-env.yml` file. For example, `STUDIO_DB_PASSWORD`, `HADOOP_PASSWORD-ALIAS`, and so on.
4. Perform database clean up by performing the following:

| Schema  | Applicable for OFSAA | Applicable for non-OFSAA |
|---|----------------------|--------------------------|
| <a href="#">Cleanup for Studio Schema</a>           | Yes                  | Yes                      |
| <a href="#">Cleanup for BD or ECM Atomic Schema</a> | Yes                  | No                       |
| <a href="#">Cleanup for BD or ECM Config Schema</a> | Yes                  | No                       |

5. Redeploy FCC Studio. For more information, see [Deploying FCC Studio on the K8s Cluster](#).

### 8.1 Cleanup for Studio Schema

To clean up the Studio schema, follow these steps:

1. Drop the existing Studio schema and create a new Studio schema.

**NOTE**

The username and password credentials of the Studio Schema in the wallet files must be updated accordingly. (If applicable)

2. Grant the following permissions to the newly created Oracle Database Schema:
  - `GRANT create session TO <Schema User>;`

- GRANT create table TO <Schema User>;
- GRANT create view TO <Schema User>;
- GRANT create any trigger TO <Schema User>;
- GRANT create any procedure TO <Schema User>;
- GRANT create sequence TO <Schema User>;
- GRANT execute on dbms\_ols TO <Schema User>;
- GRANT execute on sys.dbms\_session TO <Schema User>;
- ALTER USER <Schema User> QUOTA 100M ON users;
- GRANT create sequence TO <Schema User>;
- GRANT create SYNONYM TO <Schema User>;
- GRANT ALL privileges TO <Studio Schema User>;

---

**NOTE** If dropping the schema is not an option, drop the tables and sequences as mentioned in [Appendix - Tables and Sequences](#).

## 8.2 Cleanup for BD or ECM Atomic Schema

To clean up the BD or ECM Atomic Schema, follow these steps:

1. Log in to the BD or ECM Atomic Schema.
2. Truncate the DATABASECHANGELOG and DATABASECHANGELOGLOCK tables using the following command:

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;  
TRUNCATE TABLE DATABASECHANGELOG;
```

## 8.3 Cleanup for BD or ECM Config Schema

To clean up the BD or ECM Config schema, follow these steps:

1. Log in to the BD or ECM Config Schema.
2. Truncate the DATABASECHANGELOG and DATABASECHANGELOGLOCK tables using the following command:

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;  
TRUNCATE TABLE DATABASECHANGELOG;
```

## 9 Appendix - Starting and Stopping Services

Following are the steps to start and stop the service and authservice files. Similar steps is applicable for other services.

- [Start a Service](#)
- [Stop a Service](#)

### 9.1 Start a Service

To start a service, follow these steps:

1. Open the `<Studio_Installation_Path>/out/deploy.sh` file  
The `deploy.sh` file contains details of the commands to start the FCC Studio services.
2. Navigate to the `<Studio_Installation_Path>/out` directory.
3. Run the start command in the console to start a service:

**For example:**

```
kubectl -n <Namespace> apply -f deployments/authservice.yml
```

You can check the logs of FCC Studio from the Kubernetes Dashboard.

### 9.2 Stop a Service

To stop a service, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/out` directory.
2. Run the stop command in the console to stop service:

**For example:**

```
kubectl -n <Namespace> delete -f deployments/authservice.yml
```

## 10 Appendix A - Tables and Sequences

The list of tables and sequences to be dropped during reinstallation of FCC Studio are as follows:

- [Studio Schema Tables](#)
- [Studio Schema Sequences](#)

### 10.1 Studio Schema Tables

The following table lists the Studio Schema tables that must be dropped during reinstallation of FCC Studio.

|                            |                           |                       |
|----------------------------|---------------------------|-----------------------|
| DS_PARAGRAPH               | DS_NOTEBOOK_TAGS          | DS_TASK_RESULTS       |
| DS_ENTITY_PERMISSIONS      | DS_ROLE                   | DS_PERMISSION_ACTIONS |
| DS_GROUP                   | DS_IS_PERMITTED           | DS_PERMISSION_MAPPING |
| DS_USER_PERMS_MAP          | DS_USER_ROLES             | DS_NOTEBOOK           |
| DS_INTERPRETER_RESULT_MSGS | DS_USER                   | DS_PERMS_MAP_ACTIONS  |
| DS_ENTITY_PERMS_MAP        | DS_TASK                   | DS_GRAPH              |
| DS_INTERPRETER_RESULT      | DS_GROUP_PERMS_MAP        | DS_NOTEBOOK_RELATIONS |
| DS_INTERPRETER_PROPS       | DS_JOB                    | DS_PERMISSION         |
| DS_ROLE_PERMS_MAP          | DS_VISUALIZATION_TEMPLATE | DS_RESULT_MESSAGE     |

|                          |                        |                       |
|--------------------------|------------------------|-----------------------|
| DS_INTERPRETER_ABILITIES | DATABASECHANGELOG      | DATABASECHANGELOGLOCK |
| DS_USER_GROUPS           | DS_INTERPRETER_VARIANT | DS_COMMENT            |
| DS_PARAGRAPH_RELATIONS   |                        |                       |

## 10.2 Studio Schema Sequences

The following table lists the Studio Schema sequences that must be dropped during reinstallation of FCC Studio.

|                            |                        |                         |
|----------------------------|------------------------|-------------------------|
| SEQ_COMMENT                | SEQ_ENTITY_PERMISSIONS | SEQ_GRAPH               |
| SEQ_GROUP                  | SEQ_INTERPRETER_RESULT | SEQ_INTERPRETER_VARIANT |
| SEQ_JOB                    | SEQ_NOTEBOOK           | SEQ_PARAGRAPH           |
| SEQ_PERMISSION             | SEQ_PERMISSION_MAPPING | SEQ_RESULT_MESSAGE      |
| SEQ_ROLE                   | SEQ_TASK               | SEQ_USER                |
| SEQ_VISUALIZATION_TEMPLATE |                        |                         |

# 11 Appendix B - Using Python Virtual Environments with PySpark

To use Python Virtual Environments with PySpark, follow these steps:

- [Create a Virtual Environment with Conda](#)
- [Include a Virtual Environment in the Init Container](#)
- [Update the Interpreter Properties](#)

## 11.1 Create a Virtual Environment with Conda

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

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

1. Ensure that you have `conda` and `conda-pack` installed.
2. Create your virtual environment using the following command:

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

**NOTE** The `<environment-name>` can be chosen freely and subsequently must be used in further commands.

3. Activate your virtual environment using the following command:  
`conda activate <environment-name>`
4. Execute the following command to obtain the path to your virtual environment:

```
which python
```

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

5. Compress your virtual environment using the following command:

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

## 11.2 Include a Virtual Environment in the Init Container

To include the virtual environment in the Init container, you must place the Virtual Environment in the same path as the Spark libraries. For information on how to create a new Init container, see [Yarn Mode](#).

## 11.3 Update the Interpreter Properties

All the properties can either be configured in the interpreter JSON files or from the Interpreters page of the FCC Studio application UI after starting the FCC Studio application.

- In the Spark Interpreter Settings page of the FCC Studio application UI (or `spark.json` file), change the following values:
- Change the value of the `spark.yarn.dist.archives` property to `/var/olds-spark-interpreter/interpreter/spark/libs/<environment-name>/<environment-name>.tar.gz#<environment-name>`
- Change the value of the `spark.pyspark.python` property to `./<environment-name>/bin/python`
- In the PySpark Interpreter Settings page of the FCC Studio application UI (or `pyspark.json` file), change the value of the `zeppelin.pyspark.python` property to `/var/olds-spark-interpreter/interpreter/spark/libs/<environment-name>/bin/python`.



## 12 Appendix C – Additional Jars – PGX

PGX-Server doesn't include Hadoop-client for reading graph from HDFS. This section can be skipped if the deployer intend to use only ready to use sample-graph or PGX server without ETL.

When deploying Studio, you must obtain the following libraries. These libraries can be obtained from your existing big data cluster or from the internet. The following list of jars are for **Hadoop-client 3.0.0-cdh6.3.0**. These libraries are referred as '**hdfs-libs**'.

**NOTE** The following Jar files for your reference. you can use the similar **hdfs-libs** jars based on your Big Data cluster.

List of libraries required are as follows:

|   |                                     |
|---|-------------------------------------|
| accessors-smart-1.2.jar                           | jaxb-api-2.2.11.jar                 |
| aopalliance-1.0.jar                               | jaxb-impl-2.2.3-1.jar               |
| asm-5.0.4.jar                                     | jcip-annotations-1.0-1.jar          |
| avro-1.8.2-cdh6.3.1.jar                           | jersey-client-1.19.jar              |
| commons-beanutils-1.9.4.jar                       | jersey-core-1.19.jar                |
| commons-cli-1.2.jar                               | jersey-guice-1.19.jar               |
| commons-codec-1.11.jar                            | jersey-json-1.19.jar                |
| commons-collections-3.2.2.jar                     | jersey-server-1.19.jar              |
| commons-compress-1.18.jar                         | jersey-servlet-1.19.jar             |
| commons-configuration2-2.1.1.jar                  | jettison-1.1.jar                    |
| commons-io-2.6.jar                                | jetty-security-9.3.25.v20180904.jar |
| commons-lang-2.6.jar                              | jetty-servlet-9.3.25.v20180904.jar  |
| commons-lang3-3.7.jar                             | jetty-util-9.3.25.v20180904.jar     |
| commons-logging-1.2.jar                           | jetty-webapp-9.3.25.v20180904.jar   |
| commons-math3-3.1.1.jar                           | jetty-xml-9.3.25.v20180904.jar      |
| commons-net-3.1.jar                               | jline-0.9.94.jar                    |
| curator-client-2.12.0.jar                         | json-smart-2.3.jar                  |
| curator-framework-2.12.0.jar                      | jsp-api-2.1.jar                     |
| curator-recipes-2.12.0.jar                        | jsr305-3.0.0.jar                    |
| gson-2.2.4.jar                                    | jsr311-api-1.1.1.jar                |
| guava-16.0.1.jar                                  | kerb-admin-1.0.0.jar                |
| guice-4.0.jar                                     | kerb-client-1.0.0.jar               |
| hadoop-annotations-3.0.0-cdh6.3.1.jar             | kerb-common-1.0.0.jar               |
| hadoop-auth-3.0.0-cdh6.3.1.jar                    | kerb-core-1.0.0.jar                 |
| hadoop-client-3.0.0-cdh6.3.1.jar                  | kerb-crypto-1.0.0.jar               |
| hadoop-common-3.0.0-cdh6.3.1.jar                  | kerb-identity-1.0.0.jar             |
| hadoop-hdfs-client-3.0.0-cdh6.3.1.jar             | kerb-server-1.0.0.jar               |
| hadoop-mapreduce-client-common-3.0.0-cdh6.3.1.jar | kerb-simplekdc-1.0.0.jar            |

|  |                            |
|--|----------------------------|
| hadoop-mapreduce-client-core-3.0.0-cdh6.3.1.jar      | kerb-util-1.0.0.jar        |
| hadoop-mapreduce-client-jobclient-3.0.0-cdh6.3.1.jar | kerby-asn1-1.0.0.jar       |
| hadoop-yarn-api-3.0.0-cdh6.3.1.jar                   | kerby-config-1.0.0.jar     |
| hadoop-yarn-client-3.0.0-cdh6.3.1.jar                | kerby-pkix-1.0.0.jar       |
| hadoop-yarn-common-3.0.0-cdh6.3.1.jar                | kerby-util-1.0.0.jar       |
| htrace-core4-4.1.0-incubating.jar                    | kerby-xdr-1.0.0.jar        |
| httpclient-4.5.3.jar                                 | log4j-1.2.17.jar           |
| httpcore-4.4.6.jar                                   | netty-3.7.0.Final.jar      |
| jackson-annotations-2.9.9.jar                        | nimbus-jose-jwt-4.41.1.jar |
| jackson-core-2.9.9.jar                               | okhttp-2.7.5.jar           |
| jackson-core-asl-1.9.13.jar                          | okio-1.6.0.jar             |
| jackson-databind-2.9.9.3.jar                         | paranamer-2.8.jar          |
| jackson-jaxrs-1.9.2.jar                              | protobuf-java-2.5.0.jar    |
| jackson-jaxrs-base-2.9.9.jar                         | re2j-1.1.jar               |
| jackson-jaxrs-json-provider-2.9.9.jar                | slf4j-api-1.7.25.jar       |
| jackson-mapper-asl-1.9.13-cloudera.1.jar             | slf4j-log4j12-1.7.25.jar   |
| jackson-module-jaxb-annotations-2.9.9.jar            | snappy-java-1.1.4.jar      |
| jackson-xc-1.9.2.jar                                 | stax2-api-3.1.4.jar        |
| javax.activation-api-1.2.0.jar                       | woodstox-core-5.0.3.jar    |
| javax.inject-1.jar                                   | xz-1.6.jar                 |
| javax.servlet-api-3.1.0.jar                          | zookeeper-3.4.8.jar        |

## 13 Appendix D – Additional Jars – Batch Service

When deploying Studio, you must obtain the following files for Batch Service.

**NOTE** The following Jar files for your reference. you can use the similar **hdfs-libs** jars based on your Big Data cluster.

List of files required are as follows:

|                                       |                                     |
|---------------------------------------|-------------------------------------|
| accessors-smart-1.2.jar               | jersey-server-1.19.jar              |
| activation-1.1.jar                    | jersey-servlet-1.19.jar             |
| asm-5.0.4.jar                         | jettison-1.1.jar                    |
| avro-1.8.2-cdh6.3.1.jar               | jetty-http-9.3.25.v20180904.jar     |
| commons-beanutils-1.9.4.jar           | jetty-io-9.3.25.v20180904.jar       |
| commons-cli-1.2.jar                   | jetty-security-9.3.25.v20180904.jar |
| commons-codec-1.11.jar                | jetty-server-9.3.25.v20180904.jar   |
| commons-collections-3.2.2.jar         | jetty-servlet-9.3.25.v20180904.jar  |
| commons-compress-1.18.jar             | jetty-util-9.3.25.v20180904.jar     |
| commons-configuration2-2.1.1.jar      | jetty-webapp-9.3.25.v20180904.jar   |
| commons-io-2.6.jar                    | jetty-xml-9.3.25.v20180904.jar      |
| commons-lang-2.6.jar                  | jline-0.9.94.jar                    |
| commons-lang3-3.7.jar                 | jsch-0.1.54.jar                     |
| commons-logging-1.2.jar               | json-smart-2.3.jar                  |
| commons-math3-3.1.1.jar               | jsp-api-2.1.jar                     |
| commons-net-3.1.jar                   | jsr305-3.0.0.jar                    |
| curator-client-2.12.0.jar             | jsr311-api-1.1.1.jar                |
| curator-framework-2.12.0.jar          | kerb-admin-1.0.0.jar                |
| curator-recipes-2.12.0.jar            | kerb-client-1.0.0.jar               |
| gson-2.2.4.jar                        | kerb-common-1.0.0.jar               |
| guava-16.0.1.jar                      | kerb-core-1.0.0.jar                 |
| hadoop-annotations-3.0.0-cdh6.3.1.jar | kerb-crypto-1.0.0.jar               |
| hadoop-auth-3.0.0-cdh6.3.1.jar        | kerb-identity-1.0.0.jar             |
| hadoop-common-3.0.0-cdh6.3.1.jar      | kerb-server-1.0.0.jar               |
| hive-exec-1.1.0-cdh5.13.0.jar         | kerb-simplekdc-1.0.0.jar            |
| HiveJDBC4.jar                         | kerb-util-1.0.0.jar                 |
| hive-metastore-1.1.0-cdh5.13.0.jar    | kerby-asn1-1.0.0.jar                |
| hive-service-1.1.0-cdh5.13.0.jar      | kerby-config-1.0.0.jar              |
| htrace-core4-4.1.0-incubating.jar     | kerby-pkix-1.0.0.jar                |
| httpClient-4.5.3.jar                  | kerby-util-1.0.0.jar                |
| httpcore-4.4.6.jar                    | kerby-xdr-1.0.0.jar                 |
| jackson-annotations-2.9.0.jar         | log4j-1.2.17.jar                    |

---

|  |                            |
|--|----------------------------|
| jackson-core-2.9.9.jar                   | netty-3.7.0.Final.jar      |
| jackson-core-asl-1.9.13.jar              | nimbus-jose-jwt-4.41.1.jar |
| jackson-databind-2.9.9.3.jar             | paranamer-2.8.jar          |
| jackson-jaxrs-1.9.2.jar                  | protobuf-java-2.5.0.jar    |
| jackson-mapper-asl-1.9.13-cloudera.1.jar | re2j-1.1.jar               |
| jackson-xc-1.9.2.jar                     | slf4j-api-1.7.25.jar       |
| javax.activation-api-1.2.0.jar           | slf4j-log4j12-1.7.25.jar   |
| javax.servlet-api-3.1.0.jar              | snappy-java-1.1.4.jar      |
| jaxb-api-2.2.2.jar                       | stax2-api-3.1.4.jar        |
| jaxb-impl-2.2.3-1.jar                    | stax-api-1.0-2.jar         |
| jcip-annotations-1.0-1.jar               | woodstox-core-5.0.3.jar    |
| jersey-core-1.19.jar                     | xz-1.6.jar                 |
| jersey-json-1.19.jar                     | zookeeper-3.4.8.jar        |

## 14 Frequently Asked Questions (FAQs)

The FCC Studio installer performs all the pre-requisite validation checks during installation. The questions below help you resolve some issues that you may face during installation.

**NOTE** This is not an official support document and just attempts to share the knowledge of problem resolution to a few of the known issues. If your issue is not resolved after performing the proposed solution, contact [My Oracle Support \(MOS\)](#).

### 1. Why does my console show an unsuccessful message during wallet creation?

Please check if you have run the following commands correctly. For more information on wallet creation, see [Setup Password Stores with Oracle Wallet](#).

- a. `mkstore -wrl <wallet_location> -create //creates a wallet in the specified location`
- b. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the studio schema`
- c. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the atomic schema`
- d. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name> //creates an alias in the config schema`

### 2. Where can I find my created wallet?

Your wallet will be in the directory you have set as your wallet location.

### 3. Can I run `./install.sh` again if I make updates to the `studio-env.yml` file?

Make updates to the `studio-env.yml` file and then trigger the installation by running the `./install.sh` command. If you have already triggered the installation, then you must do the following. For more information on the attributes you can configure, see [Configure the studio-env.yml File](#).

- Perform a backup of the `studio-env.yml` file.
- Update the applicable attributes in the `studio-env.yml` file.
- run `./install.sh`
- run `./fcc-studio.sh`.

### 4. When should I create a Database link, and if yes, how do I do it?

Create a Database link to connect the Atomic and Config 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 config 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.

## 5. Why does my installed studio setup not have any notebooks?

**There are some default notebooks that are ready-to-use when you install FCC Studio. If you do not see any notebooks when you log in to the application, you may not be assigned any roles. Check the <Studio\_Installation\_Path>/logs directory to see if you have been assigned any roles, and if not, contact your administrator.**

## 6. 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 any errors in the applicable log file in the <Studio\_Installation\_Path>/logs directory.

## 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 applicable log file in the <Studio\_Installation\_Path>/logs directory.

## 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.BatchEx
ecute
```

Set the `FIC_DB_HOME` path in the `<Studio_Installation_Path>/ficdb` directory.

You can also check for any errors in the applicable log file in the `<Studio_Installation_Path>/logs` directory.

### 9. Why is the PGX Server not starting?

The PGX server starts only after the FCDM tables are created, that is, 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 applicable log file in the `<Installed path>/logs` directory.

### 10. 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 applicable log file in the `<Installed path>/logs` directory.

### 11. What should I do if there is an 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
(CombinedPgxDriver.java:147) at
oracle.pgx.graphviz.driver.PgxDriver.getGraph(PgxDriver.java:334) at
oracle.pgx.graphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer
.java:223) at
oracle.pgx.graphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer
.java:209) at
oracle.pgx.graphviz.library.QueryEnhancer.query(QueryEnhancer.java:150
) at
oracle.pgx.graphviz.library.QueryEnhancer.execute(QueryEnhancer.java:1
36) at
oracle.pgx.graphviz.interpreter.PgsqlInterpreter.interpret(PgsqlInterpre
ter.java:131) at
oracle.datastudio.interpreter.pgx.PgxInterpreter.interpret(PgxInterpre
ter.java:120) at
org.apache.zepplin.interpreter.LazyOpenInterpreter.interpret(LazyOpen
Interpreter.java:103) at
org.apache.zepplin.interpreter.remote.RemoteInterpreterServer$Interpr
etJob.jobRun(RemoteInterpreterServer.java:632) at
org.apache.zepplin.scheduler.Job.run(Job.java:188) at
org.apache.zepplin.scheduler.FIFOScheduler$1.run(FIFOScheduler.java:1
40) at
java.base/java.util.concurrent.Executors$RunnableAdapter.call(Executors
.java:515) at
java.base/java.util.concurrent.FutureTask.run(FutureTask.java:264) at
java.base/java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFu
tureTask.run(ScheduledThreadPoolExecutor.java:304) at
java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPool
Executor.java:1128) at
java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoo
lExecutor.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(Completable
Future.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) at
oracle.datastudio.interpreter.pgx.CombinedPgxDriver.getOrCreateSession
(CombinedPgxDriver.java:145) ... 17 more Caused 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-CQAZPV67UM4H at
oracle.pgx.common.marshalers.ExceptionMarshaler.toUnserializedExceptio
n(ExceptionMarshaler.java:107) at
oracle.pgx.common.marshalers.ExceptionMarshaler.unmarshal(ExceptionMar
shaler.java:123) at
oracle.pgx.client.RemoteUtils.parseExceptionalResponse(RemoteUtils.jav
a:130) at
oracle.pgx.client.HttpRequestExecutor.executeRequest(HttpRequestExecut
or.java:198) at
oracle.pgx.client.HttpRequestExecutor.get(HttpRequestExecutor.java:165
) at
oracle.pgx.client.RemoteControlImpl$10.request(RemoteControlImpl.java:
313) at
oracle.pgx.client.RemoteControlImpl$ControlRequest.request(RemoteContr
olImpl.java:119) at
oracle.pgx.client.RemoteControlImpl$ControlRequest.request(RemoteContr
olImpl.java:110) at
oracle.pgx.client.AbstractAsyncRequest.execute(AbstractAsyncRequest.ja
va:47) at
oracle.pgx.client.RemoteControlImpl.request(RemoteControlImpl.java:107
) at
oracle.pgx.client.RemoteControlImpl.getSessionInfo(RemoteControlImpl.j
ava:296) at
oracle.pgx.api.ServerInstance.lambda$getSessionInfoAsync$14(ServerInst
ance.java:490) at
java.base/java.util.concurrent.CompletableFuture.uniComposeStage(Compl
etableFuture.java:1106) at
java.base/java.util.concurrent.CompletableFuture.thenCompose(Completab
leFuture.java:2235) at
oracle.pgx.api.PgxFuture.thenCompose(PgxFuture.java:158)
```

Then, please perform below steps as a workaround -

- a. Export the "Manual Decision" Notebook
- b. Add the link parameter just below Description
- c. for Ex - "link": "manualDecision",



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

- d. truncate the table "fcc\_er\_paragraph\_manual" in Studio Schema.
- e. Import the modified notebook again.

**12. What should I do when the result set is truncated if the size goes above '102400' bytes?**

Perform the following steps:

- a. Login to FCC Studio.
- b. Navigate to interpreter **zeppelin.interpreter.output.limit**.



- c. Set the value to the required size.
- d. Restart the Studio Application.

**13. What should I do when the spark interpreter is not working?**

- a. Login to Server.
- b. Navigate to \$SPARK\_HOME directory. If the path is not set, then navigate to \$Studio\_HOME/interpreters/interpreter/spark/extralibs/spark\*.
- c. Export the following environment variables:
 

```
export HADOOP_CONF_DIR=<HADOOP Configuration Directory path>
export SPARK_HOME=<SPARK CLIENT DIRECTORY path>
export SPARK_CONF_DIR=<spark-defaults.conf directory path >
export SPARK_SUBMIT_OPTS="-Djava.security.krb5.conf=<kerberos
directory path>/krb5.conf"
```
- d. Run the following command:
 

```
./bin/run-example --master yarn SparkPi 10
```
- e. The result of the above command should be Pie value. (It ensures that the client is configured successfully.

**14. What should I do when the spark interpreter is not working?**

- a. Login to the Server.
- b. Navigate to \$SPARK\_HOME directory. If the path is not set, then navigate to \$Studio\_HOME/interpreters/interpreter/spark/extralibs/spark\*.

**c. Export the following environment variables:**

```
export HADOOP_CONF_DIR=<HADOOP Configuration Directory path>
export SPARK_HOME=<SPARK CLIENT DIRECTORY path>
export SPARK_CONF_DIR=<spark-defaults.conf directory path >
export SPARK_SUBMIT_OPTS="-Djava.security.krb5.conf=<kerberos
directory path>/krb5.conf"
```

**d. Run the following command to ...**

```
./bin/spark-submit --class org.apache.spark.examples.SparkPi --
master yarn <SPARK_HOME/examples/jars/>/spark-examples_<Version>.jar
10
```

For example, in case of spark 2.11-2.4.0, the command is as follows:

```
./bin/spark-submit --class org.apache.spark.examples.SparkPi --
master yarn <SPARK_HOME/examples/jars/>/spark-examples_2.11-
2.4.0.jar 10
```

**e. The result of the above command is displayed as a Pie value. (It ensures that the client can successfully connect to the remote cluster.)****15. What should I do when the spark interpreter is not working?****a. Login to Server.****b. Navigate to \$SPARK\_HOME directory. If the path is not set, then navigate to \$Studio\_HOME/interpreters/interpreter/spark/extralibs/spark\*.****c. Export the following environment variables:**

```
export HADOOP_CONF_DIR=<HADOOP Configuration Directory path>
export SPARK_HOME=<SPARK CLIENT DIRECTORY path>
export SPARK_CONF_DIR=<spark-defaults.conf directory path >
export SPARK_SUBMIT_OPTS="-Djava.security.krb5.conf=<kerberos
directory path>/krb5.conf"
```

**d. Run the following command:**

```
./bin/spark-submit --class
org.apache.spark.sql.hive.thriftserver.SparkSQLCLIDriver --master
yarn -e "Show databases"
```

**e. The result of the above command displays the list of databases that exist in HIVE.****16. What should I do when the spark interpreter is not working?****a. Login to Server.****b. Navigate to \$SPARK\_HOME directory. If the path is not set, then navigate to \$Studio\_HOME/interpreters/interpreter/spark/extralibs/spark\*.****c. Export the following environment variables:**

```
export HADOOP_CONF_DIR=<HADOOP Configuration Directory path>
export SPARK_HOME=<SPARK CLIENT DIRECTORY path>
```

```
export SPARK_CONF_DIR=<spark-defaults.conf directory path >
export SPARK_SUBMIT_OPTS="-Djava.security.krb5.conf=<kerberos
directory path>/krb5.conf"
```

**d.** Run the following command:

```
./bin/spark-submit --class
org.apache.spark.sql.hive.thriftserver.SparkSQLCLIDriver --master
yarn -e "select * from <hiveSchema>.<tableName> limit 10"
```

**e.** The result of the above command ensures that the client can query from the HIVE schema.

**17.** What should I do when you see the following error in the `spark.log` file?

```
Could not find or load main class
org.apache.spark.deploy.yarn.ExecutorLauncher
```

- a.** Login to the FCC Studio.
- b.** Navigate to Interpreter configurations.
- c.** Click on Spark Interpreter.
- d.** The `spark.yarn.archive` field value must be empty.

**18.** What should I do when you see the following error in the `spark.log` file?

```
INFO client.TransportClientFactory: Successfully created connection to
after 105 ms (0 ms spent in bootstraps)

Exception in thread "main"
java.lang.reflect.UndeclaredThrowableException
at
org.apache.hadoop.security.UserGroupInformation.doAs (UserGroupInformati
on.java:1713)
at
org.apache.spark.deploy.SparkHadoopUtil.runAsSparkUser (SparkHadoopUtil.
scala:64)
at
org.apache.spark.executor.CoarseGrainedExecutorBackend$.run (CoarseGrain
edExecutorBackend.scala:188)
at
org.apache.spark.executor.CoarseGrainedExecutorBackend$.main (CoarseGrai
nedExecutorBackend.scala:281)
at
org.apache.spark.executor.CoarseGrainedExecutorBackend.main (CoarseGrain
edExecutorBackend.scala)
Caused by: org.apache.spark.rpc.RpcTimeoutException: Futures timed out
after [120 seconds]. This timeout is controlled by spark.rpc.askTimeout
at
org.apache.spark.rpc.RpcTimeout.org$apache$spark$rpc$RpcTimeout$$create
RpcTimeoutException (RpcTimeout.scala:47)
at
org.apache.spark.rpc.RpcTimeout$$anonfun$addMessageIfTimeout$1.applyOrElse
lse (RpcTimeout.scala:62)
at
org.apache.spark.rpc.RpcTimeout$$anonfun$addMessageIfTimeout$1.applyOrElse
lse (RpcTimeout.scala:58)
at
scala.runtime.AbstractPartialFunction.apply (AbstractPartialFunction.sca
```

```
la:36)
at org.apache.spark.rpc.RpcTimeout.awaitResult (RpcTimeout.scala:76)
at org.apache.spark.rpc.RpcEndpointRef.askSync (RpcEndpointRef.scala:92)
at org.apache.spark.rpc.RpcEndpointRef.askSync (RpcEndpointRef.scala:76)
at
org.apache.spark.executor.CoarseGrainedExecutorBackend$$anonfun$run$1.a
pply$mcV$sp (CoarseGrainedExecutorBackend.scala:202)
at
org.apache.spark.deploy.SparkHadoopUtil$$anon$2.run (SparkHadoopUtil.sca
la:65)
at
org.apache.spark.deploy.SparkHadoopUtil$$anon$2.run (SparkHadoopUtil.sca
la:64)
at java.security.AccessController.doPrivileged (Native Method)
at javax.security.auth.Subject.doAs (Subject.java:422)
at
org.apache.hadoop.security.UserGroupInformation.doAs (UserGroupInformati
on.java:1698)
```

- a. Login to the FCC Studio.
- b. Navigate to Interpreter configurations.
- c. Click on Spark Interpreter.
- d. The `spark.master` field vlue must be configured as `yarn`.
- e. The `spark.master` should not be set in `spark-default.conf` file.

## OFSAA Support

Raise a Service Request (SR) in [My Oracle Support \(MOS\)](#) for queries related to the OFSAA applications.

## Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site that has all the revised/recently released documents.

