Oracle® Banking Microservices Architecture

Installer Guide

Release 14.7.0.0.0
F74096-01
November 2022

Oracle Banking Microservices Architecture Installer Guide, Release 14.7.0.0.0

F74096-01

Contents

Index

Preface

Purpose

This guide provides the procedure for installation of Oracle Banking Microservices Architecture and related products including database creation and required schemas using the Installer.

> **Note:**
>
> For the exact version to be installed, refer to **Tech Stack** section of **Release Notes**.

Audience

This guide is intended for WebLogic admin or ops-web team who are responsible for installing the OFSS banking products.

Acronyms and Abbreviations

The list of the acronyms and abbreviations that are used in this guide are as follows:

Table 1 Acronyms and Abbreviations

Abbreviation	Description
CMC	Common Core
OS	Operating System
SMS	Security Management System
VM	Virtual Machine

List of Topics

This guide is organized as follows:

Table 2 List of Topics

Topic	Description
Installation Overview	This topic provides the overview about the Product Installation procedure.
Database Installation	This topic provides the information to install the database for product installation.
Download and Setup Installer	This topic provides the information to download and setup the installer.
Foundation Setup	This topic provides the systematic instruction to setup the Oracle Banking Microservices Platform Foundation services using Installer.
Product Setup	This topic provides the systematic instruction to setup the Product services using Installer.
Miscellaneous Tasks	This topic provides the information about the various miscellaneous tasks performed in the installer.

Related Documents

For more information, refer to the following documents:

- Product Installation Guide

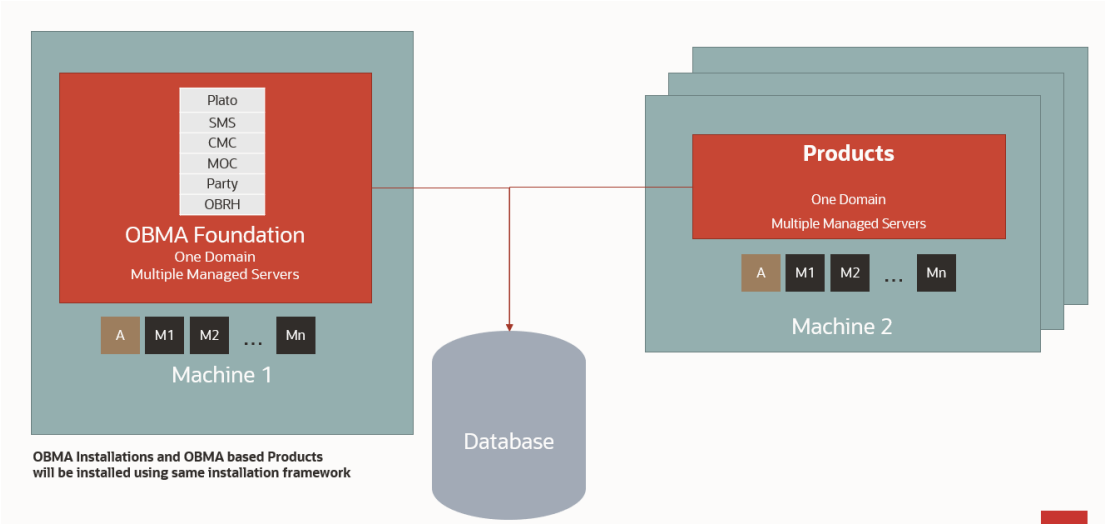
1 Installation Overview

This topic describes the overview on the installation process through the Installer.

ECO System

To run any Oracle Banking Microservices Architecture based product, the user needs to have Oracle Banking Microservices Architecture foundation installed in the ECO system and then the user can install your product in same or in different machine. The following diagram is representation of installed Oracle Banking Microservices Architecture ECO system. In a domain there will be one Admin server and there can be multiple managed servers to be used for load balancing of the services.

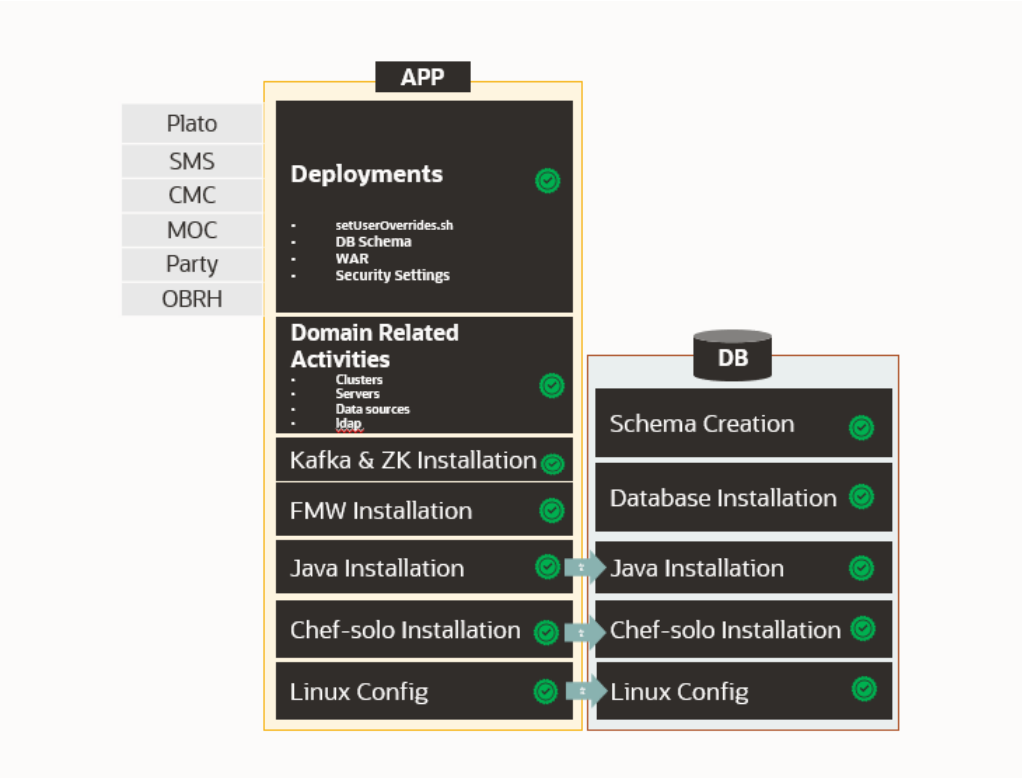
Figure 1-1 ECO System



Oracle Banking Microservices Architecture Installation

Typical Oracle Banking Microservices Architecture installation consists of the deployment of Plato, SMS, CMC, MOC, Party and Oracle Banking Routing Hub wars, there are separate managed servers under a domain where war are distributed for load balancing. Also, the Oracle Banking Microservices Architecture foundation server can have zookeeper, kafka on the same physical machine. The Eureka is also installed as part of foundation which helps service discovery. The following diagram is a representation of the Oracle Banking Microservices Architecture Foundation installation.

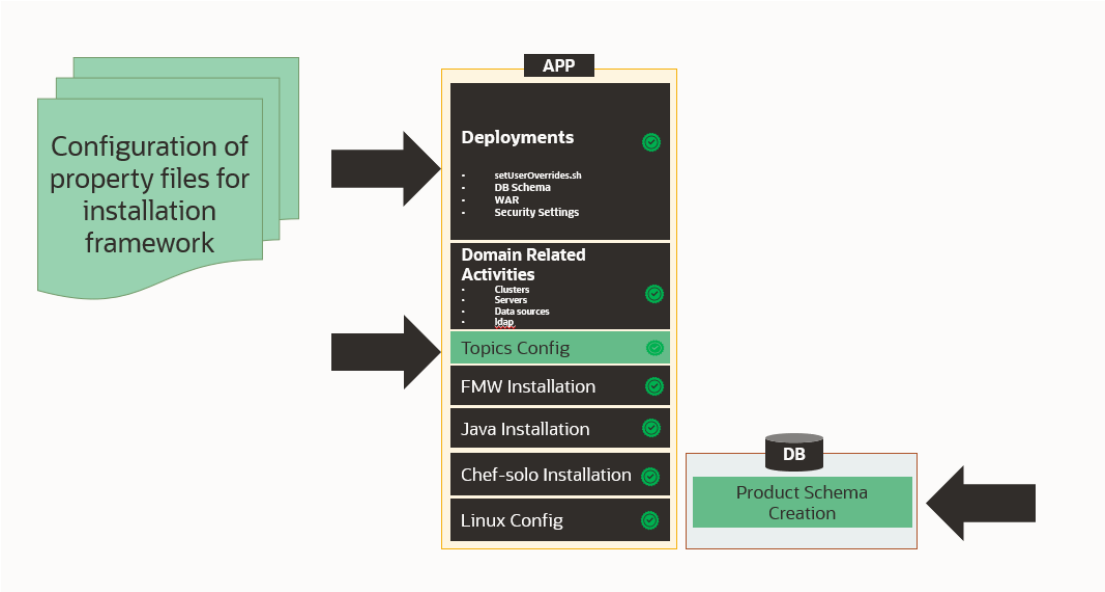
Figure 1-2 Oracle Banking Microservices Architecture Installation



Product Installation

Any Oracle Banking Microservices Architecture product installation can be visualized as like below diagram, it is assumed that you have existing Oracle Banking Microservices Architecture foundation and database installed and ready. You can then install your product on top of it. Many products can be installed in the same ECO system.

Figure 1-3 Product Installation



2
Database Installation

This topic describes the information about the database installation and the schema creation.

Oracle Database needs to be installed and required schemas needs to be created before the installation. Database installation is not part of the installer.

Refer to the **Product Installation Guide** to create the database schemas.

3
Download and Setup Installer

This topic describes the information to download and setup the Installer.

Download Installer

The installer is provided in OSDC zip of each product.

Perform the following steps to download the installer.

1. Launch putty and login to the VM (where the installation is planned) with OS user.
2. Create a directory `obma_installer` in `/scratch`.

   ```
   mkdir -p /scratch/obma_installer ; chmod 755 /scratch/obma_installer
   ```

3. Navigate to the new directory `obma_installer`.

   ```
   cd /scratch/obma_installer/
   ```

4. Download the installer zip file from the product OSDC zip to `obma_installer` directory.
5. Unzip the installer zip file by executing the below command.

   ```
   unzip <product zip file>
   ```

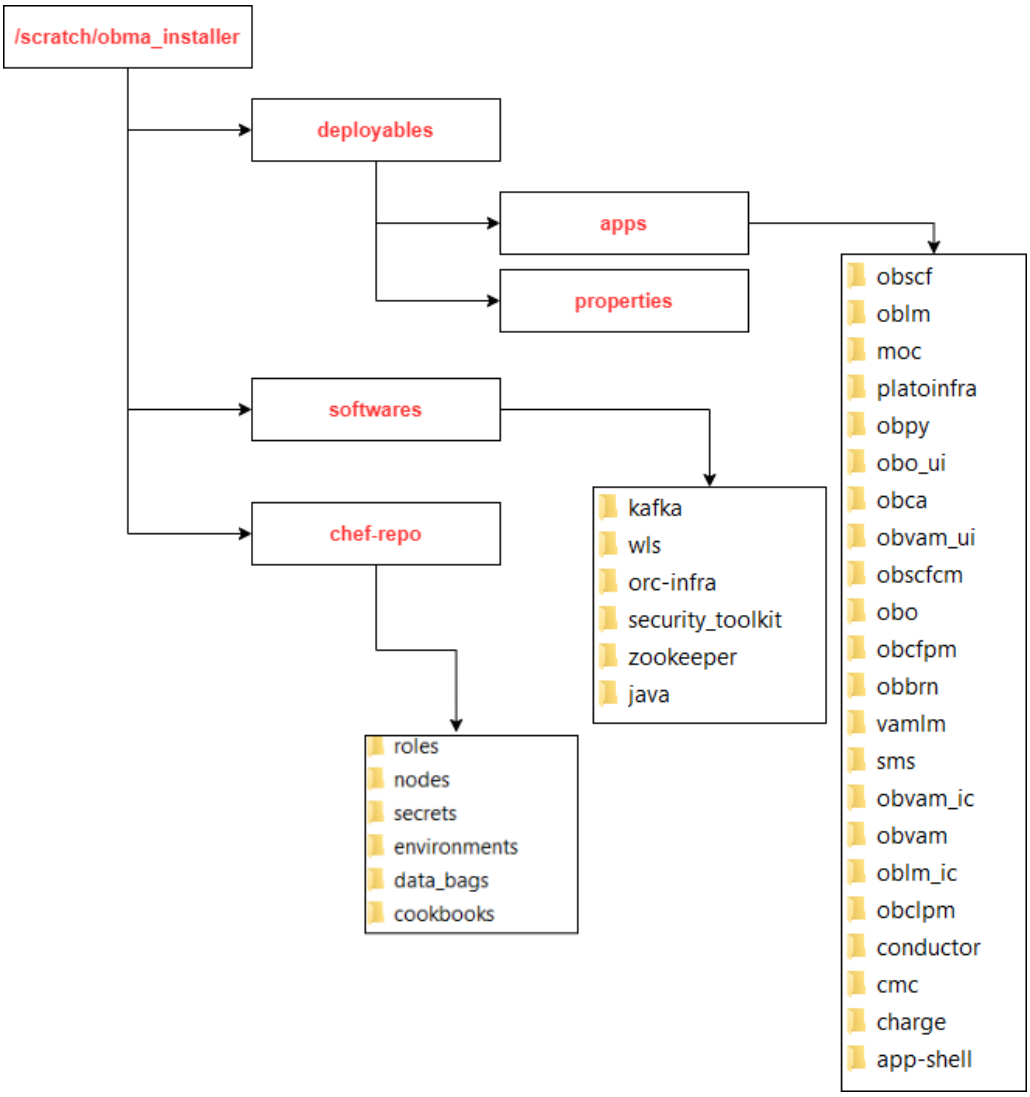
Installer Folder Structure

Post unzip of the installer file, the following directories will be displayed.

- **deployables** - contains sub-directories **apps** and **properties**.
  - The **apps** directory contains the various product wise directories where the applications or the wars files will be located for deployment. Please note, these files should be downloaded to the respective product directories from the artifactory path before starting the installation.
- **softwares** - contains the various software's required during the installation, like, java, weblogic, kafka, zookeeper, etc.
- **chef-repo** - contains various subdirectories, properties files, scripts etc., which are required for the installation.

The same is depicted in the below diagram along with their sub-directories.

Figure 3-1 Installer Folder Structure



> **Note:**
>
> For each product, the applicable folders are displayed in the respective directories.

Download Applications (Domains) Related War Files

Before performing installation, copy the WAR file from the respective artifactory path to the respective folders in the below mentioned folder structure.

`/scratch/obma_installer/deployables/apps`

> **Note:**
>
> Installer will not check the presence of files in the respective directories before installation. The user needs to ensure all the required files with correct version are available in the respective directory for the product.

Download Softwares

Before performing installation, copy the required software's to respective folders in the below mentioned folder structure.

`/scratch/obma_installer/deployables/softwares`

> **Note:**
>
> Installer will not check presence of software files in the respective directories before installation. The user needs to ensure all the required software files with correct version are available in the respective directory for the product.

Install the Oracle Replacement Configurator

1. Launch putty and login with the root user.
2. Navigate to the chef repo path: `cd /scratch/obma_installer/chef-repo`.
3. Verify the version of Oracle Replacement Configurator installed in the VM by executing the command `chef-solo --version`.
4. If the VM has older version of chef or Oracle Replacement Configurator, then remove the same by executing the command `yum remove orc-infra-<version_no.>`.

   Alternatively, in case of chef solo installation, remove the same by executing the command `yum remove chef-*`.
5. Install the new version of Oracle Replacement Configurator by executing the `install_orc.sh` script, and the command for the same is `./install_orc.sh`.
6. Verify the version as mentioned in the **Step 3**.

4
Foundation Setup

This topic describes about the Foundation setup using Oracle Banking Microservices Architecture Installer.

Post completion of **Download and Setup Installer** tasks for VM identified for Foundation Setup, perform the below mentioned configurations.

- Update Properties File
  This topic provides the systematic instructions to update the Properties file for Foundation setup.
- Update Roles File
  This topic provides the systematic instructions to update the Roles file for Foundation setup.
- Execute Installer Script
  This topic provides the systematic instructions to execute Installer Script for Foundation setup.

4.1 Update Properties File

This topic provides the systematic instructions to update the Properties file for Foundation setup.

1. Launch WinSCP and login to Foundation VM with OS User (eg.: ofssobp).
2. Navigate to the path `/scratch/obma_installer/chef-repo/` and update the file `obma_properties.rb` with the following details.
3. Update the local user and its group.

```
#Standard Values
INSTALL_USER = "ofssobp"
INSTALL_GROUP = "dba"
USER_ROOT = "root"
GROUP_ROOT = "root"
INSTALL_BASE_DIR = "/scratch"
EXTRACT_LOC = "/scratch/extract"
```

4. Verify the version of java, update if required, and ensure the same version is available in the software's directory.

```
#Java Installation Details
JAVA_INSTALLER_SOURCE = "filesystem"
JAVA_INSTALLER_PATH = "/java/"
#JAVA_INSTALLATION_DIR = "/scratch/app/product"
JAVA_INSTALLATION_DIR = INSTALL_BASE_DIR + "obma"
JAVA_VERSION = "1.8"
JDK_INSTALLER_VERSION = "jdk1.8.0_281"
```

```
JDK_INSTALLER_FILE = "jdk-8u281-linux-x64.tar.gz"
CERTS_DIRNAME = INSTALL_BASE_DIR + "/ssl/"
```

5. If the zookeeper installation will be done in the same foundation VM, then, update the hostname.
6. Verify the version of the zookeeper installable, update if required, and ensure the same version is available in the software's directory.

```
#Zookeeper Installation Details
ZOOKEEPER_HOST1 = "whf00jno.in.example.com"
# ZOOKEEPER_HOST2 = "whf00dwv.in.example.com"
# ZOOKEEPER_HOST3 = "whf001sz.in.example.com"
PEER_PORT = 2891
LEADER_PORT = 3881
CLIENT_PORT = 2181
#ZOOKEEPER_INSTALL_DIR = INSTALL_BASE_DIR + "/app/zookeeper"
ZOOKEEPER_INSTALL_DIR = INSTALL_BASE_DIR + "/obma/zookeeper"
ZOOKEEPER_INSTALL_USER_HOME = INSTALL_BASE_DIR
ZOOKEEPER_VERSION = "apache-zookeeper-3.6.3-bin"
ZOOKEEPER_INSTALLER_PATH = SOFTWARE_INSTALLER_HOME + "/zookeeper/"
ZOOKEEPER_INSTALLER_FILE = "apache-zookeeper-3.6.3-bin.tar.gz"
```

7. If kafka installation will be done in the same foundation VM, then update the hostname.
8. Verify the version of the kafka installable, update if required, and ensure the same version is available in the software's directory.

```
#Kafka Installation Details
#KAFKA_INSTALL_DIR = INSTALL_BASE_DIR + "/app/kafka"
KAFKA_INSTALL_DIR = INSTALL_BASE_DIR + "/obma/kafka"
KAFKA_INSTALL_USER_HOME = INSTALL_BASE_DIR
KAFKA_INSTALLER_PATH = SOFTWARE_INSTALLER_HOME + "/kafka"
KAFKA_INSTALLER_FILE = "kafka_2.13-2.6.0.tgz"
# KAFKA_VERSION = "2.13-2.6.0"
KAFKA_SCALA_VERSION = "2.13"
JMX_PORT = "9999"
#Kafka Broker Configurations
KAFKA_BROKER_ID = 1
KAFKA_LISTEN_PORT = 9092
LOG_RETENTION_HOURS = "168"
LOG_RETENTION_CHECK_INTERVAL = "300000"
LOG_SEGMENT_BYTES = "1073741824"
LOG_RETENTION_BYTES = "1073741824"

KAFKA_HOST = "whf00jno.in.example.com"
KAFKA_PORT = 9092
```

9. Verify the version of Tesseract installable, update if required, and ensure the same version is available in the software's directory.

```
#Tesseract Installation Details
TESSERACT_INSTALL_USER_HOME = INSTALL_BASE_DIR
TESSERACT_INSTALL_DIR = INSTALL_BASE_DIR + "/obma/tesseract"
TESSERACT_INSTALLER_PATH = SOFTWARE_INSTALLER_HOME + "/tesseract"
```

```
INSTALLER_ZIP = "tesseract-4.1.1.zip"

LEPTONICA_INSTALLER_FILE = "leptonica-1.80.0.tar.gz"
LEPTONICA_INSTALLER_VERSION = "leptonica-1.80.0"
TESSERACT_INSTALLER_FILE = "tesseract-4.1.1.tar.gz"
TESSERACT_INSTALLER_VERSION = "tesseract-4.1.1"
```

10. Update hostname for LDAP configuration.

```
#LDAP Details
LDAP_HOST = "ofss-mum-1315.snbomrshared1.gbucdsint02bom.oraclevcn.com"
LDAP_PORT = "7002"
```

11. Verify the version of weblogic server, update if required, and ensure the same version is available in the software's directory.

```
#Weblogic Infra Installation Details
#WLS_INSTALL_USER_HOME = "/scratch"
ORACLE_INVENTORY = "/scratch/app/oraInventory"
WLS_VERSION = "12.2.1.4"
WLS_INSTALLER_SOURCE = "filesystem"
WLS_INSTALLER_PATH = "/wls/"
WLS_PACKAGE_BASENAME = "fmw_12.2.1.4.0_infrastructure.jar"
WLS_INSTALLER_FILE = "fmw_12.2.1.4.0_infrastructure_Disk1_1of1.zip"
#WLS_INSTALL_DIR = "/app/product/fmw"
WLS_INSTALL_DIR = INSTALL_BASE_DIR + "/obma"
WLS_INSTALLER_TYPE = 'Fusion Middleware Infrastructure'
```

12. Update the hostname for plato configuration.

```
#Product Specific Weblogic Server runtime parameters
#PLATO_CONFIG_SERVICES_URI = "http://whf00jno.in.example.com"
PLATO_CONFIG_SERVICES_PORT = "8001"
APPLICATION_ENVIRONMENT = "DEV"
APPLICATION_LOGGING_PATH = "/scratch/work_area/logs"
PLATO_APIGATEWAY_URI = "http://whf00jno.in.example.com"
```

13. Update the flyway domain locations i.e., update the details of domain locations for all the products that are considered for installation.

```
#FLYWAY_DOMAIN_LOCATIONS-
"db/migration/domain/plato, db/migration/domain/sms, db/migration/domain/
moc,
db/migration/domain/cmc, db/migration/domain/obpy, db/migration/domain/
obremo,
db/migration/domain/obtfpm, db/migration/domain/obedx, db/migration/
domain/oblm,
db/migration/domain/obic, db/migration/domain/vamlm, db/migration/domain/
oflo,
db/migration/domain/obvam, db/migration/domain/obclpm, db/migration/
domain/obcfpm,
db/migration/domain/obpm, db/migration/domain/obcm, db/migration/domain/
obscf,
db/migration/domain/obscfcm
```

14. The default servers and their respective ports are already defined. Any new addition of server details needs to be appended here under “#Product specific Weblogic Server runtime parameters”.

```
#PLATO CommonCore, SMS and Midoffice common Server ports details
PLATO_CONFIG_SVCS_MAN_SERVER_LISTEN_PORT = "8001"
PLATO_CONFIG_SVCS_MAN_SERVER_SSL_PORT = "8002"

PLATO_DISCOVERY_SVCS_MAN_SERVER_LISTEN_PORT = "8003"
PLATO_DISCOVERY_SVCS_MAN_SERVER_SSL_PORT = "8004"

PLATO_API_GATEWAY_MAN_SERVER_LISTEN_PORT = "8005"
PLATO_API_GATEWAY_MAN_SERVER_SSL_PORT = "8006"
```

15. The default datasources are already defined. Any new addition of datasource needs to be appended here under “#PLATO ComonCore, SMS and Midoffice Datasource and Datasource target details”

```
#PLATO CommonCore, SMS and Midoffice Datasource and Datasource
target details
PLATO_SCHEMA = "PLATO"
PLATO_JNDI = "jdbc/PLATO"
PLATO_DS_TARGET =
"cmc_cluster1,cmc_cluster2,cmc_cluster3,cmc_cluster4,moc_cluster,pla
to_o_cluster,
plato_api_gateway_cluster,plato_others_cluster,plato_config_cluster,
plato_orch_cluster,
plato_ui_config_cluster,sms_cluster"

PLATOSEC_SCHEMA = "PLATOSEC"
PLATOSEC_JNDI = "jdbc/PLATO_SECURITY"
PLATO_SECURITY_DS_TARGET =
"plato_config_cluster,plato_api_gateway_cluster,plato_others_cluster
"

PLATO_UI_SCHEMA = "PLATO"
PLATO_UI_JNDI = "jdbc/PLATO_UI_CONFIG"
PLATO_UI_CONFIG_DS_TARGET =
"plato_ui_config_cluster,cmc_cluster1,cmc_cluster2,cmc_cluster3,cmc_
cluster4,
moc_cluster,plato_others_cluster,plato_orch_cluster"

SMS_SCHEMA = "SMS"
SMS_JNDI = "jdbc/sms"
SMS_DS_TARGET =
"plato_orch_cluster,sms_cluster,cmc_cluster1,cmc_cluster2,cmc_cluste
r3,cmc_cluster4,
moc_cluster,plato_others_cluster"
```

> **Note:**
>
> The password for all the default schema's is "wlcome1". In case there is change in the password for the schemas, user needs to update the same in databag. Refer *Password Update in Databag* section for more details.

16. Update the database details under “#Database details for weblogic datasource configuration”

```
#Database details for Weblogic datasource configuration
ORACLE_PDB_SID = "PBP0163A"
ORACLE_PDB_HOSTNAME = "whf00ivq.in.example.com"
ORACLE_PDB_PORT = "1521"
ORACLE_DRIVER = "oracle.jdbc.driver.OracleDriver"
```

17. Also, there are attributes and values related to individual products. Verify the details for your product/s, and in case, any changes to the default values are required, update accordingly.

> **Note:**
>
> The below snapshot from Oracle Banking Cash Management product for reference.

Figure 4-1 FLYWAY PLACEHOLDER

```
#*********************************************************************************************************************************************#
#--------------------------------
# OBCA FLYWAY PLACEHOLDER DETAILS
#--------------------------------

# OBCA Server Port Details
OBCA1_MAN_SERVER_LISTEN_PORT= "8400"
OBCA1_MAN_SERVER_SSL_PORT = "8401"

OBCA2_MAN_SERVER_LISTEN_PORT= "8402"
OBCA2_MAN_SERVER_SSL_PORT = "8403"

OBCA3_MAN_SERVER_LISTEN_PORT= "8404"
OBCA3_MAN_SERVER_SSL_PORT = "8405"

OBCA4_MAN_SERVER_LISTEN_PORT= "8406"
OBCA4_MAN_SERVER_SSL_PORT = "8407"

# OBCA Datasource Details
OBCACFG_SCHEMA = "OBCACFG"
OBCACFG_JNDI = "jdbc/OBCACFG"
OBCACFG_DS_TARGET = "obca_cluster1"

OBCAPM_SCHEMA = "OBCAPM"
OBCAPM_JNDI = "jdbc/OBCAPM"
OBCAPM_DS_TARGET = "obca_cluster2"

OBCAPP_SCHEMA = "OBCAPP"
OBCAPP_JNDI = "jdbc/OBCAPP"
OBCAPP_DS_TARGET = "obca_cluster3"

OBCASTMNT_SCHEMA = "OBCASTMNT"
OBCASTMNT_JNDI = "jdbc/OBCASTMNT"
OBCASTMNT_DS_TARGET = "obca_cluster4"

OBCA_PLATO_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
OBCA_PLATO_UI_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
OBCA_SMS_UI_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
OBCA_CMNCORE_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
OBCA_PLATOFEED_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
OBCA_PLATOBATCH_DS_TARGET = "obca_cluster1,obca_cluster2,obca_cluster3,obca_cluster4"
```

4.2 Update Roles File

This topic provides the systematic instructions to update the Roles file for Foundation setup.

Navigate to the path `/scratch/obma_installer/chef-repo/roles/` and update the file "`obma_mw.rb`" with the below details.

1. In case of addition or changes to the existing kafka topics, modify the same under “topics”.

Figure 4-2 Kafka Topics

```
  },
obma_kafka: {
   install_dir: KAFKA_INSTALL_DIR,
   install_user: INSTALL_USER,
   user_home: KAFKA_INSTALL_USER_HOME,
   install_group: INSTALL_GROUP,
       kafka_installer_path: KAFKA_INSTALLER_PATH,
   kafka_package_name: KAFKA_INSTALLER_FILE,
   # kafka_version: KAFKA_VERSION,
   kafka_scala_version: KAFKA_SCALA_VERSION,
   jmx_port: JMX_PORT,
   log: {
     retention_hours: LOG_RETENTION_HOURS,
     retention_check_interval: LOG_RETENTION_CHECK_INTERVAL,
     segment_bytes: LOG_SEGMENT_BYTES,
     retention_bytes: LOG_RETENTION_BYTES
   },
 topics: {
    topic1: {
     topic_name: "rpmDashboard",
     replication_factor: "1",
     partitions: "1",
     config: {
     "segment.bytes": "1073741824",
     "retention.ms": "604800000"
     }
     },
     topic2: {
     topic_name: "InitialFundingAck",
     replication_factor: "1",
     partitions: "1",
     config: {
     "segment.bytes": "1073741824",
     "retention.ms": "604800000"
     }
     },
            topic3: {
     topic_name: "PartyKYCStatusUpdate",
     replication_factor: "1",
     partitions: "1",
     config: {
     "segment.bytes": "1073741824",
     "retention.ms": "604800000"
     }
     },
            topic4: {
     topic_name: "PartyHandoffNotification",
     replication_factor: "1",
     partitions: "1",
```

2. In case of addition or changes to the existing cluster configuration, modify the same under “cluster_config”.

Figure 4-3 Cluster Configuration

```
plato_config_services_port: PLATO_CONFIG_SERVICES_PORT,
plato_service_logging_path: APPLICATION_LOGGING_PATH,
plato_service_env: APPLICATION_ENVIRONMENT,
        oracle_driver: ORACLE_DRIVER,
cluster_configure: CONFIGURE_WLS_CLUSTER,
is_node_primary: "true",
cluster_config: {
    plato_config_cluster: {
        managed_servers: {
            Config_Server1: {
                listen_port: PLATO_CONFIG_SVCS_MAN_SERVER_LISTEN_PORT,
                ssl_port: PLATO_CONFIG_SVCS_MAN_SERVER_SSL_PORT,
                java_memory_min: "512",
                java_memory_max: "1024",
            }
        }
    },
    plato_discovery_cluster: {
        managed_servers: {
            Discovery_Server1: {
                listen_port: PLATO_DISCOVERY_SVCS_MAN_SERVER_LISTEN_PORT,
                ssl_port: PLATO_DISCOVERY_SVCS_MAN_SERVER_SSL_PORT,
                java_memory_min: "512",
                java_memory_max: "1024",
            }
        }
    },
    plato_api_gateway_cluster: {
        managed_servers: {
           API_Gateway_Server1: {
                listen_port: PLATO_API_GATEWAY_MAN_SERVER_LISTEN_PORT,
                ssl_port: PLATO_API_GATEWAY_MAN_SERVER_SSL_PORT,
                java_memory_min: "512",
                java_memory_max: "1024",
            }
        }
    },
    plato_ui_config_cluster: {
        managed_servers: {
           Plato_UI_Config_Server1: {
                listen_port: PLATO_UI_MAN_SERVER_LISTEN_PORT,
                ssl_port: PLATO_UI_MAN_SERVER_SSL_PORT,
                java_memory_min: "512",
                java_memory_max: "1024",
            }
        }
    },
    plato_o_cluster: {
        managed_servers: {
           Plato_O_Server1: {
```

3. In case of addition or changes to the existing data source configuration, modify the same under “datasource_config”.

Figure 4-4 Datasource Configuration

```
            }
        }
    }
},
datasource_configure: "true",
datasource_config: {
    PLATO: {
        database_name: ORACLE_PDB_SID,
        driver_class: "oracle.jdbc.OracleDriver",
        jndi_name: PLATO_JNDI,
        host_name: ORACLE_PDB_HOSTNAME,
        port: ORACLE_PDB_PORT,
        global_transaction_protocol: "OnePhaseCommit",
        database_user_name: PLATO_SCHEMA,
                        target: PLATO_DS_TARGET
       },
               PLATOSEC: {
        database_name: ORACLE_PDB_SID,
        driver_class: "oracle.jdbc.OracleDriver",
        jndi_name: PLATOSEC_JNDI,
        host_name: ORACLE_PDB_HOSTNAME,
        port: ORACLE_PDB_PORT,
        global_transaction_protocol: "OnePhaseCommit",
        database_user_name: PLATOSEC_SCHEMA,
                        target: PLATO_SECURITY_DS_TARGET
     },
    PLATO_UI: {
        database_name: ORACLE_PDB_SID,
        driver_class: "oracle.jdbc.OracleDriver",
        jndi_name: PLATO_UI_JNDI,
        host_name: ORACLE_PDB_HOSTNAME,
        port: ORACLE_PDB_PORT,
        global_transaction_protocol: "OnePhaseCommit",
        database_user_name: PLATO_UI_SCHEMA,
                        target: PLATO_UI_CONFIG_DS_TARGET
    },
    SMS: {
        database_name: ORACLE_PDB_SID,
        driver_class: "oracle.jdbc.OracleDriver",
        jndi_name: SMS_JNDI,
        host_name: ORACLE_PDB_HOSTNAME,
        port: ORACLE_PDB_PORT,
        global_transaction_protocol: "OnePhaseCommit",
        database_user_name: SMS_SCHEMA,
                        target: SMS_DS_TARGET
    },
    CONDUCTOR: {
        database_name: ORACLE_PDB_SID,
        driver_class: "oracle.jdbc.OracleDriver",
```

4. In case of addition or changes to the existing services or war files, modify the same under “app_deployment”.

Figure 4-5 Application Deployment

```
            driver_class: "oracle.jdbc.OracleDriver",
            jndi_name: COMMON_CORE_JNDI,
            host_name: ORACLE_PDB_HOSTNAME,
            port: ORACLE_PDB_PORT,
            global_transaction_protocol: "OnePhaseCommit",
            database_user_name: COMMON_CORE_SCHEMA,
                            target: COMMON_CORE_DS_TARGET
        }
    },
    app_installer_path: "filesystem",
        app_dirname_url: PRODUCT_BUNDLE_HOME,
        app_deployment: {
        app1: {
            app_file_path: "/deployables/apps/platoinfra",
            app_file_name: "plato-config-service-7.3.0.1.war",
            app_target_name: "plato_config_cluster"
        },
        app2: {
            app_file_path: "/deployables/apps/platoinfra",
            app_file_name: "plato-discovery-service-7.2.0.war",
            app_target_name: "plato_discovery_cluster"
        },
                        app3: {
            app_file_path: "/deployables/apps/platoinfra",
            app_file_name: "plato-api-gateway-7.3.0.war",
            app_target_name: "plato_api_gateway_cluster"
        },
        app4: {
            app_file_path: "/deployables/apps/platoinfra",
            app_file_name: "plato-ui-config-services-7.3.0.war",
            app_target_name: "plato_ui_config_cluster"
        },
        app5: {
            app_file_path: "/deployables/apps/conductor",
            app_file_name: "conductor-server-v2.30.1_3.war",
            app_target_name: "plato_o_cluster"
        },
        app6: {
            app_file_path: "/deployables/apps/sms",
            app_file_name: "sms-core-services-7.3.0.war",
            app_target_name: "sms_cluster"
        },
        app7: {
            app_file_path: "/deployables/apps/platoinfra",
            app_file_name: "plato-orch-service-7.3.0.war",
            app_target_name: "plato_orch_cluster"
        },
                    app8: {
           app_file_path: "/deployables/apps/platoinfra",
           app_file_name: "plato-alerts-management-services-7.3.0.war",
```

5. Set the respective product installation as true, which will be configured as part of this Oracle Banking Microservices Architecture Environment Setup activity i.e. if “OBCFPM” will be installed, set the attribute “is_obcfpm_installation” to true.

Figure 4-6 Environment Setup - True

```
                    app69: {
      app_file_path: "/deployables/apps/obo_ui",
      app_file_name: "oboflo-component-server-7.3.0.war",
      app_target_name: "plato_api_gateway_cluster"
    }
},
      is_obcfpm_installation: "true",
   obcfpm_flyway_placefolder: {
obcfpm_Server1_port: OBCFPM1_MAN_SERVER_SSL_PORT,
obcfpm_Server2_port: OBCFPM2_MAN_SERVER_SSL_PORT,
obcfpm_Server3_port: OBCFPM3_MAN_SERVER_SSL_PORT,
obcfpm_Server4_port: OBCFPM4_MAN_SERVER_SSL_PORT,
obcfpm_Server5_port: OBCFPM5_MAN_SERVER_SSL_PORT,
obcfpm_Server6_port: OBCFPM6_MAN_SERVER_SSL_PORT,
obpy_Server_port: OBPY_MAN_SERVER_SSL_PORT,
        collateral_schema: COLLATERAL_SCHEMA,
        collateral_jndi: COLLATERAL_JNDI,
        externalcheck_schema: EXTERNALCHECK_SCHEMA,
        externalcheck_jndi: EXTERNALCHECK_JNDI,
        risk_schema: RISK_SCHEMA,
        risk_jndi: RISK_JNDI,
        fieldinvestigation_schema: FIELDINVESTIGATION_SCHEMA,
        fieldinvestigation_jndi: FIELDINVESTIGATION_JNDI,
        facility_schema: FACILITY_SCHEMA,
        facility_jndi: FACILITY_JNDI,
        maintennce_schema: MAINTENANCE_SCHEMA,
        maintenance_jndi: MAINTENANCE_JNDI,
        valuation_schema: VALUATION_SCHEMA,
        valuation_jndi: VALUATION_JNDI,
        legal_schema: LEGAL_SCHEMA,
        legal_jndi: LEGAL_JNDI,
        safekeeping_schema: SAFEKEEPING_SCHEMA,
        safekeeping_jndi: SAFEKEEPING_JNDI,
        registration_schema: REGISTRATION_SCHEMA,
        registration_jndi: REGISTRATION_JNDI,
        stage_schema: STAGE_SCHEMA,
        stage_jndi: STAGE_JNDI,
        scoring_schema: SCORING_SCHEMA,
        scoring_jndi: SCORING_JNDI,
        covenant_schema: COVENANT_SCHEMA,
        covenant_jndi: COVENANT_JNDI,
        exception_schema: EXCEPTION_SCHEMA,
        exception_jndi: EXCEPTION_JNDI
```

6. Similarly, set all the other product installation to true, if the same will be configured as part of the environment setup, else set the same as false.

Figure 4-7 Environment Setup - False

```
                CDDAPPTXNBTCH_JNDI: CDDAPPTXNBTCH_JNDI,
                DDASTMNTAPP_SCHEMA: DDASTMNTAPP_SCHEMA,
                DDASTMNTAPP_JNDI: DDASTMNTAPP_JNDI
},
            is_obvam_installation: "false",
        obvam_flyway_placefolder: {
                    obvam_hostname: OBVAM_HOSTNAME,
                    obvam_ic_Server_port: OBVAM_IC_MAN_SERVER_SSL_PORT,
                    obvam_Server_port: OBVAM_MAN_SERVER_SSL_PORT,
                    charge_Server_port: CHARGE_MAN_SERVER_SSL_PORT,
                    vam_schema: VAM_SCHEMA,
                vam_jndi: VAM_JNDI,
                vat_schema: VAT_SCHEMA,
                vat_jndi: VAT_JNDI,
                vas_schema: VAS_SCHEMA,
                vas_jndi: VAS_JNDI,
                van_schema: VAN_SCHEMA,
                van_jndi: VAN_JNDI,
                eda_schema: EDA_SCHEMA,
                eda_jndi: EDA_JNDI,
                vab_schema: VAB_SCHEMA,
                vab_jndi: VAB_JNDI,
                vac_schema: VAC_SCHEMA,
                vac_jndi: VAC_JNDI,
                vai_schema: VAI_SCHEMA,
                vai_jndi: VAI_JNDI,
                vae_schema: VAE_SCHEMA,
                vae_jndi: VAE_JNDI,
                eie_schema: EIE_SCHEMA,
                eie_jndi: EIE_JNDI,
                elm_schema: ELM_SCHEMA,
                elm_jndi: ELM_JNDI,
                vap_schema: VAP_SCHEMA,
                vap_jndi: VAP_JNDI,
                vas_ds_schema: VAS_DS_SCHEMA,
                vas_ds_jndi: VAS_DS_JNDI,
                vamlmchg_schema: VAMLMCHG_SCHEMA,
                vamlmchg_jndi: VAMLMCHG_JNDI
            },
 is_obo_installation: "false",
    obo_flyway_placefolder: {
                    obo_hostname: OBO_HOSTNAME,
                    obo1_server_port: OBO1_MAN_SERVER_SSL_PORT,
                    obo2_server_port: OBO2_MAN_SERVER_SSL_PORT,
                    obo3_server_port: OBO3_MAN_SERVER_SSL_PORT,
                    obremobussprc_schema: OBREMOBUSSPRC_SCHEMA,
                    obremobussprc_jndi: OBREMOBUSSPRC_JNDI,
                    obremobpdetails_schema: OBREMOBPDETAILS_SCHEMA,
                    obremobpdetails_jndi: OBREMOBPDETAILS_JNDI,
```

7. Navigate to the bottom of the file and verify the recipes to be executed. All the listed recipes will be executed in sequential order as shown below.

Figure 4-8 Recipes List

```
run_list
['recipe[obma_sysprep::user_creation]','recipe[obma_sysprep::ulimit]','recipe[obma_java::_install_java]','
recipe[obma_java::create_certs]','recipe[obma_zookeeper]','recipe[obma_kafka]','recipe[obma_tesseract::tes
seract_prerequisite]','recipe[obma_tesseract::install_leptonica]','recipe[obma_tesseract::install_tesserac
t]','recipe[obma_weblogic::install_wls]','recipe[obma_weblogic::domain]','recipe[obma_weblogic::startadmin
]','recipe[obma_weblogic::startnm]','recipe[obma_weblogic::configureembaddedwlsldap]','recipe[obma_weblogi
c::ssl_admin]','recipe[obma_weblogic::stopadmin]','recipe[obma_weblogic::ssl_nodemanager]','recipe[obma_we
blogic::restartadmin]','recipe[obma_weblogic::cluster]','recipe[obma_weblogic::addjdbcconnections_plato]',
'recipe[obma_weblogic::setuseroverridesupdate_plato]','recipe[obma_weblogic::startman]','recipe[obma_weblo
gic::deployapp]']
```

> **Note:**
>
> Remove the `user_creation` and `ulimit` recipes from the above list since it is used for internal purpose only.

4.3 Execute Installer Script

This topic provides the systematic instructions to execute Installer Script for Foundation setup.

1. Launch putty and login to foundation VM with NIS user (eg.: dkarkera) and then switch to root user.
2. Navigate to the chef-repo path by executing the command,

   ```
   cd /scratch/obma_installer/chef-repo
   ```

3. Execute the installer script by executing the command `./obma_installer.sh`.

This will perform the silent installation of Oracle Banking Microservices Architecture foundation.

5

Product Setup

This topic describes about the Product setup using Oracle Banking Microservices Architecture Installer.

Post completion of **Download and Setup Installer** tasks for VM identified for Product Setup, perform the below mentioned configurations.

> **Note:**
>
> Oracle Banking Origination product is used as reference for understanding purpose.

- Update Properties File
  This topic provides the systematic instructions to update the Properties file for Product setup.
- Update Roles File
  This topic provides the systematic instructions to update the Roles file for Product setup.
- Execute Installer Script
  This topic provides the systematic instructions to execute Installer Script for Product setup.

5.1 Update Properties File

This topic provides the systematic instructions to update the Properties file for Product setup.

1. Navigate to the path `/scratch/obma_installer/chef-repo/`.
2. Open the respective product properties file and update the `obma_properties.rb` with the following details.
3. Update the local user and its group.

```
#Standard Values
INSTALL_USER = "ofssobp"
INSTALL_GROUP = "dba"
USER_ROOT = "root"
GROUP_ROOT = "root"
INSTALL_BASE_DIR = "/scratch"
EXTRACT_LOC = "/scratch/extract"
```

4. Verify the version of java, update if required, and ensure the same version is available in the software's directory.

```
#Java Installation Details
JAVA_INSTALLER_SOURCE = "filesystem"
JAVA_INSTALLER_PATH = "/java/"
#JAVA_INSTALLATION_DIR = "/scratch/app/product"
```

```
JAVA_INSTALLATION_DIR = INSTALL_BASE_DIR + "obma"
JAVA_VERSION = "1.8"
JDK_INSTALLER_VERSION = "jdk1.8.0_281"
JDK_INSTALLER_FILE = "jdk-8u281-linux-x64.tar.gz"
CERTS_DIRNAME = INSTALL_BASE_DIR + "/ssl/"
```

5. Verify the version of weblogic server, update if required, and ensure the same version is available in the software's directory.

```
#Weblogic Infra Installation Details
#WLS_INSTALL_USER_HOME = "/scratch"
ORACLE_INVENTORY = "/scratch/app/oraInventory"
WLS_VERSION = "12.2.1.4"
WLS_INSTALLER_SOURCE = "filesystem"
WLS_INSTALLER_PATH = "/wls/"
WLS_PACKAGE_BASENAME = "fmw_12.2.1.4.0_infrastructure.jar"
WLS_INSTALLER_FILE = "fmw_12.2.1.4.0_infrastructure_Disk1_1of1.zip"
#WLS_INSTALL_DIR = "/app/product/fmw"
WLS_INSTALL_DIR = INSTALL_BASE_DIR + "/obma"
WLS_INSTALLER_TYPE = 'Fusion Middleware Infrastructure'
```

6. Update the Product setup hostname and verify various ports, and update if required.

```
#Product Specific parameters
PLATO_HOST = "ofss-
mum-1315.snbomprshared1.gbucdsint02bom.oraclevcn.com"
PLATO_CONFIG_PORT = "8002"
DISCOVERY_PORT = "8004"
API_GATEWAY_PORT = "8006"
SMS_PORT = "8026"
PROTOCOL = "https"
```

7. For the respective product, the default servers and their ports are already defined. Any new addition of server or datasource details needs to be appended here under respective product “Flyway configuration details”.

> **Note:**
>
> The below snapshot from Oracle Banking Origination FLYWAY Configuration Details is given for reference.

Figure 5-1 FLYWAY Configurations Details

```
#**************************************************************************************************
#----------------------------------
# OBO FLYWAY Configurations Details
#----------------------------------

OBO_HOSTNAME = "whf00dxw.in.oracle.com"

# OBO Server ports details
OBO1_MAN_SERVER_LISTEN_PORT = "7101"
OBO1_MAN_SERVER_SSL_PORT = "7102"
OBO2_MAN_SERVER_LISTEN_PORT = "7103"
OBO2_MAN_SERVER_SSL_PORT = "7104"
OBO3_MAN_SERVER_LISTEN_PORT = "7105"
OBO3_MAN_SERVER_SSL_PORT = "7106"

# OBO Datasource Details
OBREMOBUSSPRC_SCHEMA = "OBREMOBUSSPRC"
OBREMOBUSSPRC_JNDI = "jdbc/OBREMOBUSSPRC"
OBREMOBUSSPRC_DS_TARGET = "obo1_cluster1"

OBREMOBPDETAILS_SCHEMA = "OBREMOBPDETAILS"
OBREMOBPDETAILS_JNDI = "jdbc/OBREMOBPDETAILS"
OBREMOBPDETAILS_DS_TARGET = "obo1_cluster1"

CMNAPPLICANT_SCHEMA = "CMNAPPLICANT"
CMNAPPLICANT_JNDI = "jdbc/CMNAPPLICANT"
CMNAPPLICANT_DS_TARGET = "obo2_cluster1"

OBREMOCOLLATERAL_SCHEMA = "OBREMOCOLLATERAL"
OBREMOCOLLATERAL_JNDI = "jdbc/OBREMOCOLLATERAL"
OBREMOCOLLATERAL_DS_TARGET = "obo2_cluster1"

RPMHOST_SCHEMA = "RPMHOST"
RPMHOST_JNDI = "jdbc/RPMHOST"
RPMHOST_DS_TARGET = "obo2_cluster1"

IPA_SCHEMA = "IPA"
IPA_JNDI = "jdbc/IPA"
IPA_DS_TARGET = "obo2_cluster1"
```

> **Note:**
>
> The password for all the default schema's is "wlcome1". In case there is change in the password for the schemas, user needs to update the same in databag. Refer *Password Update in Databag* section for more details.

8. Update database details under “#Database details for weblogic datasource configuration”

```
#Database details for Weblogic datasource configuration
ORACLE_PDB_SID = "PBP0163A"
ORACLE_PDB_HOSTNAME = "whf00ivq.in.example.com"
ORACLE_PDB_PORT = "1521"
ORACLE_DRIVER = "oracle.jdbc.driver.OracleDriver"
```

5.2 Update Roles File

This topic provides the systematic instructions to update the Roles file for Product setup.

Navigate to the path `/scratch/obma_installer/chef-repo/roles/` and open the respective product role file. Here, we will consider "`obo_mw.rb`" for reference.

1. In case of addition or changes to the existing cluster configuration, modify the same under “cluster_config”.

Figure 5-2 Cluster Configuration

```
        nodemgr_mode: NODEMGR_MODE,
        domain_path: DOMAIN_PATH,
        domain_start_mode: DOMAIN_START_MODE,
        nodemgr_port: NODEMGR_PORT,
        admin_server_name: ADMIN_SERVER_NAME,
        plato_config_services_uri: PLATO_CONFIG_SERVICES_URI,
plato_config_services_port: PLATO_CONFIG_SERVICES_PORT,
        plato_apigateway_uri: PLATO_CONFIG_SERVICES_URI,
        plato_apigateway_port: PLATO_API_GATEWAY_MAN_SERVER_LISTEN_PORT,
plato_service_logging_path: APPLICATION_LOGGING_PATH,
plato_service_env: APPLICATION_ENVIRONMENT,
        cluster_configure: CONFIGURE_WLS_CLUSTER,
entityservices_port: SMS_MAN_SERVER_LISTEN_PORT,
is_node_primary: "true",
cluster_config: {
    obo1_cluster1: {
        managed_servers: {
            obo1_Server1: {
                                    listen_port: OBO1_MAN_SERVER_LISTEN_PORT,
                ssl_port: OBO1_MAN_SERVER_SSL_PORT,
                java_memory_min: "2048",
                java_memory_max: "3072",
            }
        }
    },
                obo2_cluster1: {
        managed_servers: {
            obo2_Server1: {
                                    listen_port: OBO2_MAN_SERVER_LISTEN_PORT,
                ssl_port: OBO2_MAN_SERVER_SSL_PORT,
                java_memory_min: "2048",
                java_memory_max: "3072",
            }
        }
    },
                obo3_cluster1: {
        managed_servers: {
            obo3_Server1: {
                                    listen_port: OBO3_MAN_SERVER_LISTEN_PORT,
                ssl_port: OBO3_MAN_SERVER_SSL_PORT,
                java_memory_min: "2048",
                java_memory_max: "3072",
            }
        }
    }
```

2. In case of addition or changes to the existing data source configuration, modify the same under “datasource_config”.

Figure 5-3 Datasource Configuration

```
            }
        }
    },
    datasource_configure: "true",
    datasource_config: {
        OBREMOBUSSPRC: {
            database_name: ORACLE_PDB_SID,
            driver_class: "oracle.jdbc.OracleDriver",
            jndi_name: OBREMOBUSSPRC_JNDI,
            host_name: ORACLE_PDB_HOSTNAME,
            port: ORACLE_PDB_PORT,
            global_transaction_protocol: "OnePhaseCommit",
            database_user_name: OBREMOBUSSPRC_SCHEMA,
                        target: OBREMOBUSSPRC_DS_TARGET
        },
                OBREMOBPDETAILS: {
            database_name: ORACLE_PDB_SID,
            driver_class: "oracle.jdbc.OracleDriver",
            jndi_name: OBREMOBPDETAILS_JNDI,
            host_name: ORACLE_PDB_HOSTNAME,
            port: ORACLE_PDB_PORT,
            global_transaction_protocol: "OnePhaseCommit",
            database_user_name: OBREMOBPDETAILS_SCHEMA,
                        target: OBREMOBPDETAILS_DS_TARGET
         },
        CMNAPPLICANT: {
            database_name: ORACLE_PDB_SID,
            driver_class: "oracle.jdbc.OracleDriver",
            jndi_name: CMNAPPLICANT_JNDI,
            host_name: ORACLE_PDB_HOSTNAME,
            port: ORACLE_PDB_PORT,
            global_transaction_protocol: "OnePhaseCommit",
            database_user_name: CMNAPPLICANT_SCHEMA,
                        target: CMNAPPLICANT_DS_TARGET
        },
        OBREMOCOLLATERAL: {
            database_name: ORACLE_PDB_SID,
            driver_class: "oracle.jdbc.OracleDriver",
            jndi_name: OBREMOCOLLATERAL_JNDI,
            host_name: ORACLE_PDB_HOSTNAME,
            port: ORACLE_PDB_PORT,
            global_transaction_protocol: "OnePhaseCommit",
            database_user_name: OBREMOCOLLATERAL_SCHEMA,
                        target: OBREMOCOLLATERAL_DS_TARGET
        },
        RPMHOST: {
            database_name: ORACLE_PDB_SID,
            driver_class: "oracle.jdbc.OracleDriver",
```

3. In case of addition or changes to the existing services or war files, modify the same under “app_deployment”.

Figure 5-4 Application Deployment

```
    },
app_installer_path: "filesystem",
    app_dirname_url: PRODUCT_BUNDLE_HOME,
    app_deployment: {
    app1: {
        app_file_path: "/deployables/apps/obo",
        app_file_name: "obremo-rpm-batch-services-7.3.0.war",
        app_target_name: "obo1_cluster1"
    },
    app2: {
        app_file_path: "/deployables/apps/obo",
        app_file_name: "obremo-rpm-businessprocess-services-7.3.0.war",
        app_target_name: "obo1_cluster1"
    },
            app3: {
        app_file_path: "/deployables/apps/obo",
        app_file_name: "obremo-rpm-businessproductdetails-services-7.3.0.war",
        app_target_name: "obo1_cluster1"
    },
    app4: {
        app_file_path: "/deployables/apps/obo",
        app_file_name: "obremo-rpm-cmn-applicantservices-7.3.0.war",
        app_target_name: "obo2_cluster1"
    },
    app5: {
        app_file_path: "/deployables/apps/obo",
        app_file_name: "obremo-rpm-cmn-collateralservices-7.3.0.war",
        app_target_name: "obo2_cluster1"
    },
    app6: {
        app_file_path: "/deployables/apps/obo",
```

4. Navigate to the bottom of the file and verify the recipes to be executed. All the listed recipes will be executed in sequential order as shown below.

Figure 5-5 Recipes List

```
run_list
['recipe[obma_sysprep::user_creation]','recipe[obma_sysprep::ulimit]','recipe[obma_java::_install_java]','
recipe[obma_java::create_certs]','recipe[obma_zookeeper]','recipe[obma_kafka]','recipe[obma_tesseract::tes
seract_prerequisite]','recipe[obma_tesseract::install_leptonica]','recipe[obma_tesseract::install_tesserac
t]','recipe[obma_weblogic::install_wls]','recipe[obma_weblogic::domain]','recipe[obma_weblogic::startadmin
]','recipe[obma_weblogic::startnm]','recipe[obma_weblogic::configureembaddedwlsldap]','recipe[obma_weblogi
c::ssl_admin]','recipe[obma_weblogic::stopadmin]','recipe[obma_weblogic::ssl_nodemanager]','recipe[obma_we
blogic::restartadmin]','recipe[obma_weblogic::cluster]','recipe[obma_weblogic::addjdbcconnections_plato]',
'recipe[obma_weblogic::setuseroverridesupdate_plato]','recipe[obma_weblogic::startman]','recipe[obma_weblo
gic::deployapp]']
```

> **Note:**
>
> Remove the `user_creation` and `ulimit` recipes from the above list since it is used for internal purpose only.

5.3 Execute Installer Script

This topic provides the systematic instructions to execute Installer Script for Product setup.

1. Launch putty and login to foundation VM with NIS user (eg.: dkarkera) and then switch to the root user.

2. Navigate to the chef-repo path by executing the command,

   ```
   cd /scratch/obma_installer/chef-repo
   ```

3. Execute the installer script by executing the command `./obo_installer.sh`.

This will perform the silent installation of Oracle Banking Origination product.

> **Note:**
>
> - The above steps remain the same for all the other products.
> - If the Foundation and Product setups are in two different VM's, then the Certificate syncup between these VM's needs to be performed before the deployment activity.
>   Refer to **Certificate Sync Up between Foundation and Product VMs** section for more details.

6
Miscellaneous Tasks

This topic describes about the Miscellaneous Tasks while installing the application using Oracle Banking Microservices Architecture Installer.

- Password Update in Databag
  This topic provides the systematic instructions to update the password in databag.
- Certificate Sync Up between Foundation and Product VMs
  This topic provides the systematic instructions to sync up the certification between the Foundation and Product VMs.

6.1 Password Update in Databag

This topic provides the systematic instructions to update the password in databag.

1. Launch putty and login to foundation VM with NIS user (eg.: dkarkera) and then switch to the root user.
2. Navigate to the chef-repo path by executing the command,

   ```
   cd /scratch/obma_installer/chef-repo
   ```
3. Set the required editor by executing the command, `export EDITOR=vim`.
4. Execute the below command to open the databag file in edit mode, `knife data bag edit --local-mode <databag_sub_directory> <datasource_credential_json_file> --secret-file <secret_key_path>`.

Table 6-1 Databag File - Attribute Description

Attribute Name	Attribute Description
databag_sub_directory	Name of sub directory where the datasource credential json file is located inside databag directory. **Example:** obma_weblogic, obma_java, obma_kafka etc
datasource_credential_json_file	Name of the datasource credential json file where all the credential related to respective product is listed. **Example:** datasourceCred_obo, datasourceCred_obca, etc **Note:** Mention the filename without the json extension.
secret_key_path	Location to the secret key. **Example:** `/scratch/obma_installer_ssl/chef-repo/secrets/secret_key`

Example:

```
knife data bag edit --local-mode obma_weblogic datasourceCred_obvam --
secret-file /scratch/obma_installer/chef-repo/secrets/secret_key
```

Figure 6-1 Sample Databag File

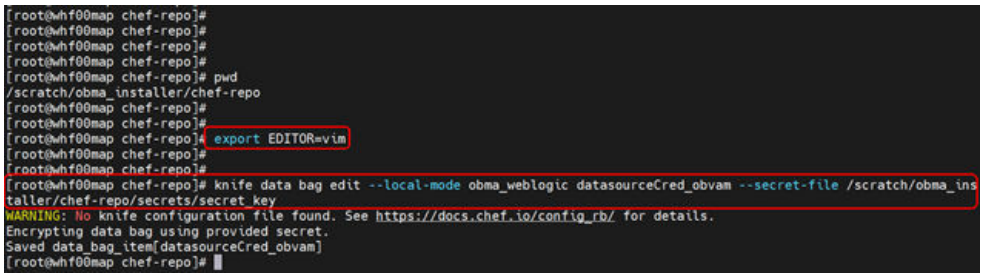


Figure 6-2 Sample Password

5. Post updating the credential file, Click **Save and Close**.

6.2 Certificate Sync Up between Foundation and Product VMs

This topic provides the systematic instructions to sync up the certification between the Foundation and Product VMs.

1. Launch putty and login to foundation VM with OS user (i.e. ofssobp).
2. Navigate to certificate directory by executing the command.

   ```
   cd /scratch/ssl/cacerts
   ```

3. Copy the certificate file of foundation VM to Product VM by executing the command.

```
scp -r <cert_foundation>
<credential_of_product>@<ip_product>:<cert_path_product>
```

Table 6-2 Certificate File - Attribute Description

Attribute Name	Attribute Description
cert_foundation	Certificate of Foundation VM. **Example:** whf00map.crt
credential_of_product	OS user of Product VM. **Example:** ofssobp
ip_product	IP or Hostname of Product VM. **Example:** 10.40.73.66
cert_path_product	Product Certificate Path. **Example:** `/scratch/ssl/cacerts`

Example: `scp -r whf00map.crt ofssobp@10.40.89.28:/scratch/ssl/cacerts`

4. Launch putty and login to foundation VM with OS user (i.e. ofssobp).
5. Navigate to certificate directory by executing the command.

   ```
   cd /scratch/ssl/cacerts
   ```

6. Copy the certificate file of foundation VM by executing the command.

   ```
   scp -r <cert_foundation>
   <credential_of_foundation>@<ip_foundation>:<cert_path_foundation>
   ```

Table 6-3 Foundation Certificate File - Attribute Description

Attribute Name	Attribute Description
cert_path_foundation	Foundation Certificate Path. **Example:** `/scratch/ssl/cacerts`
cert_product	Certificate of product VM. **Example:** whf00gbl.crt
credential_of_foundation	OS user of foundation VM. **Example:** ofssobp
ip_foundation	IP or Hostname of foundation VM. **Example:** 10.40.73.66

Example: `scp -r whf00map.crt ofssobp@10.40.89.28:/scratch/ssl/cacerts`

7. In product VM, navigate to the certificate path by executing the command, `cd / scratch/ssl/cacerts`.
8. Sync the certificate of foundation VM by executing the command.

   ```
   /scratch/obma/jdk-11.0.14/bin/keytool -import -v -trustcacerts -alias
   selfsigned6 -file <cert_foundation> -keystore <trust_certificate_product> --
   storepass wlcome1 -noprompt
   ```

Table 6-4 Sync Foundation Certificate File - Attribute Description

Attribute Name	Attribute Description
cert_foundation	Certificate of foundation VM. **Example:** whf00gbl.crt
trust_certificate_product	Trust certificate of product VM. **Example:** whf00map.in.example.com_trust.jks

Example: `/scratch/obma/jdk-11.0.14/bin/keytool -import -v -trustcacerts -alias selfsigned6 -file whf00gbl.crt -keystore whf00map.in.example.com_trust.jks --storepass wlcome1 -noprompt`

9. Now, switch to foundation VM putty session and navigate to the certificate path by executing the command, `cd /scratch/ssl/cacerts`.
10. Sync the certificate of product VM by executing the command.

    ```
    /scratch/obma/jdk-11.0.14/bin/keytool -import -v -trustcacerts -alias selfsigned6 -file <cert_product> -keystore <trust_certificate_foundation> --storepass wlcome1 -noprompt
    ```

Table 6-5 Sync Product Certificate File - Attribute Description

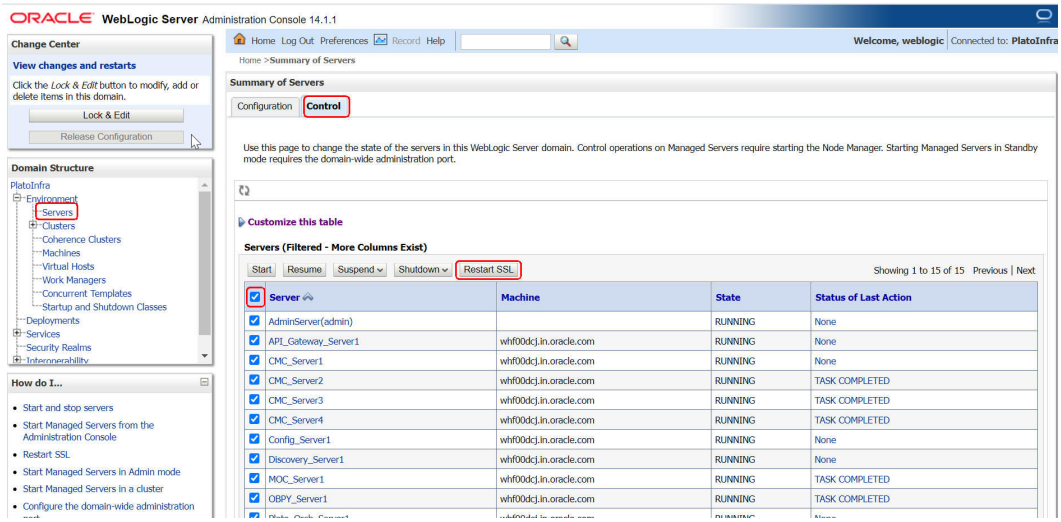
Attribute Name	Attribute Description
cert_product	Certificate of product VM. **Example:** whf00map.crt
trust_certificate_foundation	Trust certificate of foundation VM. **Example:** whf00gbl.in.example.com_trust.jks

Example: `/scratch/obma/jdk-11.0.14/bin/keytool -import -v -trustcacerts -alias selfsigned6 -file whf00map.crt -keystore whf00gbl.in.example.com_trust.jks --storepass wlcome1 -noprompt`

11. Post Syncup, launch the browser and login to Admin Console of Foundation setup.
12. Navigate to **Servers** and then click on **Control** tab.
13. Select all the servers and Click **Restart SSL** button.

    All the selected servers are restarted.

Figure 6-3 WebLogic Server - Control



14. Repeat the steps 11 to 13 in Admin Console of Product VM.

This concludes the certificate sync up activity.

Index