

Configuration and Deployment Guide

Oracle Banking Origination

Release 14.7.1.0.0

Part Number F81701-01

May 2023

Table of Contents

1. CONFIGURATION AND DEPLOYMENT.....	1-1
1.1 INTRODUCTION	1-1
1.2 PLACEHOLDER UPDATE FOR ORACLE BANKING MICROSERVICES ARCHITECTURE SERVICES	1-1
1.2.1 Method 1 – Via setUserOverrides.sh file	1-1
1.2.2 Method 2 – Via passing the -D params in the Server start argument.....	1-6
1.2.3 Method 3 – Using env files and setUserOverrides.sh file	1-8
1.2.4 Method 4 – Workflow Configuration	1-16
1.3 HOW TO CREATE DOMAIN AND CLUSTER CONFIGURATION.....	1-21
1.3.1 Domain Creation Configuration	1-21
1.3.2 Post Domain Creation Configurations.....	1-30
1.4 HOW TO CREATE DATASOURCE	1-32
1.5 HOW TO DEPLOY APPLICATION	1-37
1.6 HOW TO UNDEPLOY APPLICATION	1-42
1.7 HOW TO RESTART SERVERS	1-43
1.8 HOW TO CHECK PORT NUMBER	1-46
1.9 WEBLOGIC EMBEDDED LDAP SETUP	1-47
1.9.1 Configuration of Weblogic LDAP.....	1-47
1.9.2 Creation of Users	1-48
1.9.3 Oracle Banking Microservices Architecture Security Config Table Entries	1-51
1.10 ORACLE ANALYTIC SERVER SETUP	1-52
1.10.1 Prerequisite	1-52
1.10.2 Start BI Server.....	1-52
1.10.3 Upload BI Reports	1-52
1.10.4 Test BI Reports	1-52
1.11 HOW TO DEPLOY PLATO-APIGATEWAY ROUTER	1-53
1.11.1 Router deployment steps	1-53
1.11.2 Generation pem file and encryption of secrets:	1-54
1.11.3 Timeout parameters	1-55

1. Configuration and Deployment

1.1 Introduction

This guide is a supporting document for the installation of Oracle Banking Microservices Architecture applications. You can find the reference in the respective installation guides.

1.2 Placeholder Update for Oracle Banking Microservices Architecture Services

The Placeholder update can be performed in the following methods:

- Method 1 – Via **setUserOverrides.sh** file
- Method 2 – Via passing the **-D params** in the Server start argument
- Method 3 – Using **env** files and **setUserOverrides.sh** file
- Method 4 – Via Workflow creation in Plato O

1.2.1 Method 1 – Via setUserOverrides.sh file

Perform the following steps:

1. Create a file called **setUserOverrides.sh** inside the Weblogic bin location.
2. The following formats of the **setUserOverrides.sh** file and the list of parameters that need to be passed in order to run Oracle Banking Microservices Architecture services properly.

NOTE: Below are the list of **-D params** (ENV Variables) which needs to be set for all the individual services. Set a single **-Dparam** as follows:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -DParam =<ParamValue>"
export JAVA_OPTIONS
```

//Common Properties

```
-Dplato.services.config.port= <CONFIG_SERVICE_PORT>
-Dplato.services.config.uri=
http://<CONFIG_SERVICE_HOSTNAME>:<CONFIG_SERVICE_PORT>
-Ddeureka.client.serviceUrl.defaultZone=
http://<DISCOVERY_SERVICE_HOST>:<DISCOVERY_SERVICE_PORT>/plato-
discovery-service/eureka
-Dplato.services.entityservices.port= <PLATO_ORCH_SERVICE_PORT>
-Dplato.service.logging.path= <LOGGING_PATH>
-Dspring.cloud.loadbalancer.ribbon.enabled=false
-Dspring.main.allow-circular-references=true
```

//Flyway Common Placeholders

```
-Dflyway.domain.placeholders.eureka.host= <DISCOVERY_SERVICE_HOST>
-Dflyway.domain.placeholders.eureka.port= <DISCOVERY_SERVICE_PORT>
-Dflyway.domain.placeholders.plato-api-gateway.server.port=
<API_GATEWAY_PORT>
-Dflyway.domain.placeholders.zipkin.host= <ZIPKIN_HOSTNAME>
```

-Dflyway.domain.placeHolders.zipkin.port= <ZIPKIN_PORT>

//SMS - Needed for other services also

-Dflyway.domain.placeHolders.sms.username= <SMS_SCHEMA_USERNAME>

-Dflyway.domain.placeHolders.sms.password= <SMS_SCHEMA_PASSWORD>

-Dflyway.domain.placeHolders.sms.jdbcUrl= <SMS_SCHEMA_URL>

-Dflyway.domain.placeHolders.sms.schemas= <SMS_SCHEMA_NAME>

//Plato Config Service - Needed for other services also

-Dflyway.domain.placeHolders.plato-config.username= <PLATO_DB_USERNAME>

-Dflyway.domain.placeHolders.plato-config.password= <PLATO_DB_PASSWORD>

-Dflyway.domain.placeHolders.plato-config.jdbcUrl= <PLATO_DB_URL>

-Dflyway.domain.placeHolders.driver.className= oracle.jdbc.driver.OracleDriver

-Dflyway.domain.placeHolders.plato-config.schemas= <PLATO_DB_SCHEMANAME>

//Plato Api Gateway - Needed for other services also

-Dflyway.domain.placeHolders.api-gateway.username=
<SECURITY_DB_USERNAME>

-Dflyway.domain.placeHolders.api-gateway.password=
<SECURITY_DB_PASSWORD>

-Dflyway.domain.placeHolders.api-gateway.jdbcUrl= <SECURITY_DB_URL>

-Dflyway.domain.placeHolders.api-gateway.schemas= <SECURITY_SCHEMANAME>

-Dflyway.domain.placeHolders.apigateway.host=
<APIGATEWAY_ROUTER_HOSTNAME>

-Dflyway.domain.placeHolders.apigateway.port= <APIGATEWAY_ROUTER_PORT>

-Dflyway.domain.placeHolders.USER.STORE= <USER.STORE>

-Dflyway.domain.placeHolders.LDAP.CORS.allowed.origin= <LDAP_CORS>

-Dflyway.domain.placeHolders.LDAP.credential.SALT=
<LDAP_CREDENTIALS_SALT>

-Dflyway.domain.placeHolders.JWT.EXPIRY.seconds= <JWT_EXPIRY_SECONDS>

-Dflyway.domain.placeHolders.LDAP.url = <LDAP_SERVER_URL >

-Dflyway.domain.placeHolders.LDAP.userId = <LDAP_SERVER_USER>

-Dflyway.domain.placeHolders.LDAP.server.base = <LDAP_SERVER_BASE>

-Dflyway.domain.placeHolders.LDAP.server.credential = <LDAP_CREDENTIALS>

-Dflyway.domain.placeHolders.LDAP.usersearch.base = <LDAP_USER_BASE>

-Dflyway.domain.placeHolders LDAP.user.prefix = <LDAP_USER_PREFIX>

-Dflyway.domain.placeHolders.LDAP.provider = <LDAP_PROVIDER>

-Dflyway.domain.placeHolders.TOKEN.autoregenerate =
<TOKEN_AUTOREGENERATION>

-Dflyway.domain.placeHolders.SSO.enabled = <SSO_ENABLED>

-Dflyway.domain.placeHolders.TOKEN.regeneration.enabled =
<TOKEN_ALWAYSNEW_GENERATION >

-Dplato-api-gateway.enableAudit=true

//Plato Discovery Service

-Dflyway.domain.placeHolders.plato-discovery-service.server.port=<DISCOVERY_SERVICE_PORT>

//Plato UI-Config Services

-Dflyway.domain.placeHolders.plato-ui-config-services.server.port=<UICONFIG_SERVICE_PORT>

-Dflyway.domain.placeHolders.plato-ui-config.username=<UICONFIG_SCHEMA_USERNAME>

-Dflyway.domain.placeHolders.plato-ui-config.password=<UICONFIG_SCHEMA_PASSWORD>

-Dflyway.domain.placeHolders.plato-ui-config.jdbcUrl=<UICONFIG_SCHEMA_URL>

-Dflyway.domain.placeHolders.plato-ui-config.schemas=<UICONFIG_SCHEMA_NAME>

//Plato Apigateway Router Service

-Dflyway.domain.placeHolders.plato-apigateway-router.server.port=<APIGATEWAY_ROUTER_PORT>

-Dflyway.domain.placeHolders.plato-apigateway-router.router.protocol=<ROUTER_PROTOCOL>

-Dflyway.domain.placeHolders.plato-apigateway-router.router.meadmin.port=<ROUTER_PORT>

//Plato Feed Services

-Dflyway.domain.placeHolders.plato-feed-services.feed.upload.directory=<FEED_SERVICE_UPLOAD_PATH>

-Dflyway.domain.placeHolders.plato-feed-services.server.port=<FEED_SERVICE_PORT>

-Dflyway.domain.placeHolders.plato-feed-services.username=<FEED_DB_USERNAME>

-Dflyway.domain.placeHolders.plato-feed-services.password=<FEED_DB_PASSWORD>

-Dflyway.domain.placeHolders.plato-feed-services.jdbcUrl=<FEED_DB_URL>

-Dflyway.domain.placeHolders.plato-feed-services.schemas=<FEED_SCHEMA_NAME>

//Plato Batch Server

-Dflyway.domain.placeHolders.plato-batch-server.server.port=<BATCH_SERVER_PORT>

-Dflyway.domain.placeHolders.plato-batch-server.plato.eventhub.kafka.brokers=<EVENTHUB_KAFKA_BROKERS>

-Dflyway.domain.placeHolders.plato-batch-server.plato.eventhub.zk.nodes=<ZK_NODES>

-Dflyway.domain.placeHolders.plato-batch-server.username=<BATCH_SCHEMA_USERNAME>

-Dflyway.domain.placeHolders.plato-batch-server.password=<BATCH_SCHEMA_PASSWORD>

-Dflyway.domain.placeholders.plato-batch-server.jdbcUrl=<BATCH_SCHEMA_URL>

-Dflyway.domain.placeholders.plato-batch-server.schemas=<BATCH_SCHEMA_NAME>

// Plato-Alerts-Management-Services

-Dflyway.domain.placeholders.plato-alerts-management-services.server.port=<ALERTS-MANAGEMENT-SERVER-PORT>

-Dflyway.domain.placeholders.plato-alerts-management-services.plato.eventhub.kafka.brokers=<EVENTHUB_KAFKA_BROKERS>

-Dflyway.domain.placeholders.plato-alerts-management-services.plato.eventhub.zk.nodes=<ZK_NODES>

-Dflyway.domain.placeholders.plato-alerts-management-services.username=<ALERTS_SCHEMA_USERNAME>

-Dflyway.domain.placeholders.plato-alerts-management-services.password=<ALERTS_SCHEMA_PASSWORD>

-Dflyway.domain.placeholders.plato-alerts-management-services.jdbcUrl=<ALERTS_SCHEMA_URL>

-Dflyway.domain.placeholders.plato-alerts-management-services.schemas=<ALERTS_SCHEMA_NAME>

//Plato Orch Service

-Dflyway.domain.placeholders.plato-orch-service.server.port=<ORCH_SERVICE_PORT>

-Dflyway.domain.placeholders.plato-orchestrator.hostname=<CONDUCTOR-EUREKA-HOSTNAME >

//Conductor

-Dconductor.properties=<CONDUCTOR_CONFIG_FILE_PATH>

//Plato Regional Configurator Service

-Dflyway.domain.placeholders.plato-regional-configurator-services.server.port=<REGIONAL_CONFIGURATOR_SERVICE_PORT>

-Dflyway.domain.placeholders.plato-regional-configurator-services.locations =

- "db/migration/domain/obrc" (By default)

- "db/migration/domain/obrc, db/migration/domain<YOUR DOMAINS>"

-Dflyway.domain.placeholders.plato-regional-configurator-services.schemas = "OBRC"

-Dflyway.domain.placeholders.plato-regional-configurator-services.db.jndi = "jdbc/OBRC"

-Dflyway.domain.placeholders.plato-regional-configurator-services.username=<OBRC_SCHEMA_USERNAME>

-Dflyway.domain.placeholders.plato-regional-configurator-services.password=<OBRC_SCHEMA_PASSWORD>

//Common core NLP services

-Dflyway.domain.placeholders.cmc-nlp-annotator-

services.server.port=<CMC_NLP_ANNOTATOR_SERVICES_PORT>

-Dflyway.domain.placeholders.cmc-nlp-dashboard-widget-
 services.server.port=<CMC_NLP_DASHBOARD_SERVICES_PORT>

-Dflyway.domain.placeholders.cmc-nlp-model-mngmnt-
 services.server.port=<CMC_NLP_MODEL_MANGEMENT_PORT>

-Dflyway.domain.placeholders.cmc-nlp-online-processing-
 services.server.port=<CMC_NLP_ONLINE_PROCESSING_PORT>

-Dflyway.domain.placeholders.cmc-nlp-tag-maint-
 services.server.port=<CMC_NLP_TAG_MAINTENANCE_PORT>

-Dflyway.domain.placeholders.cmc-nlp-text-extraction-
 services.server.port=<CMC_NLP_TEXT_EXTRACTION_PORT>

-Dflyway.domain.placeholders.cmc-nlp-txn-log-
 services.server.port=<CMC_NLP_TXN_LOG_SERVICES_PORT>

-Dflyway.domain.placeholders.cmc-nlp-util-
 services.server.port=<CMC_NLP_UTIL_SERVICES_PORT>

// Common core NLP Poller service

-Dflyway.domain.placeholders.cmc-fc-ai-ml-services.server.port=<Server_Port>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-
 services.server.postingPath=<Posting_Path>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-services.server.pollingPath=<Polling_Path>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-
 services.server.pollingEmail=<Polling_Email>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-
 services.server.emailServerHost=<Email_Server_Host>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-
 services.server.emailServerPort=<Email_Server_PORT>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-
 services.server.pollingFrequency=<Polling_Frequency>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-

services.server.pollerInitialDelay=<Poller_Initial_Delay>

-Dflyway.domain.placeholders.cmc-fc-ai-ml-

services.server.emailPassword=<Poller_Email_Password>

1.2.2 Method 2 – Via passing the -D params in the Server start argument

All the above mentioned -D parameters can be passed through the Server start argument in respective managed server. Perform the following steps:

1. Navigate to the Server **Configuration** tab and click managed server to which you want to pass the values.

Summary of Servers

Configuration Control

A server is an instance of WebLogic Server that runs in its own Java Virtual Machine (JVM) and has its own configuration.

This page summarizes each server that has been configured in the current WebLogic Server domain.

[Customize this table](#)

Servers (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

New Clone Delete Showing 1 to 2 of 2 Previous | Next

Name	Type	Cluster	Machine	State	Health	Listen Port
AdminServer(admin)	Configured		whf00dkc	RUNNING	OK	7001
managed1_server	Configured		whf00dkc	RUNNING	OK	7003

New Clone Delete Showing 1 to 2 of 2 Previous | Next

2. Select **Server Start** tab in the next screen.

Settings for managed1_server

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes

General Cluster Services Keystores SSL Federation Services Deployment Migration Tuning Overload Concurrency Health Monitoring **Server Start**

Web Services Coherence

Click the **Lock & Edit** button in the Change Center to modify the settings on this page.

Save

Node Manager is a WebLogic Server utility that you can use to start, suspend, shut down, and restart servers in normal or unexpected conditions. Use this page to configure the startup settings that Node Manager will use to start this server on a remote machine.

Java Home: The Java home directory (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

Java Vendor: The Java Vendor value to use when starting this server. [More Info...](#)

BEA Home: The BEA home directory (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

Root Directory: The directory that this server uses as its root directory. This directory must be on the computer that hosts Node Manager. If you do not specify a Root Directory value, the domain directory is used by default. [More Info...](#)

Class Path: The classpath (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

3. Edit the **Arguments** field and pass all the environment parameters required for the service to run.

Node Manager is a WebLogic Server utility that you can use to start, suspend, shut down, and restart servers in normal or unexpected conditions. Use this page to configure the startup settings that Node Manager will use to start this server on a remote machine.

Java Home:	<input type="text"/>	The Java home directory (path on the machine running Node Manager) to use when starting this server. More Info...
Java Vendor:	<input type="text"/>	The Java Vendor value to use when starting this server. More Info...
BEA Home:	<input type="text"/>	The BEA home directory (path on the machine running Node Manager) to use when starting this server. More Info...
Root Directory:	<input type="text"/>	The directory that this server uses as its root directory. This directory must be on the computer that hosts Node Manager. If you do not specify a Root Directory value, the domain directory is used by default. More Info...
Class Path:	<input type="text"/>	The classpath (path on the machine running Node Manager) to use when starting this server. More Info...
Arguments:	<pre>-Deureka.server.enable-self-preservation=false -Dspring.flyway.enabled=false -Dflyway.enabled=false -Deureka.client.serviceUrl.defaultZone=http://whf00dkx:7003 /plato-discovery-service/eureka -Dserver.port=7003</pre>	The arguments to use when starting this server. More Info...

4. Save the configuration and restart the managed server. After you restart, the service can be started or deployed properly.

1.2.3 Method 3 – Using env files and setUserOverrides.sh file

Perform the following steps:

1. Copy the **setUserOverrides.sh** file to each of the <domain>/bin folder. The example of the file is given below:

```
#!/bin/bash
# shellcheck disable=SC1090
# Common functions

set -e -x

config_file=""
PLATO_CONFIG_MANAGED_SERVER_NAME=""

# This file is used only for PLATO-CONFIG service
plato_config_file="${DOMAIN_HOME}/bin/plato-config-deploy.env"

# This file is used for rest of the services
domain_config_file="${DOMAIN_HOME}/bin/domain-config-deploy.env"

if [ -f "$plato_config_file" ] ; then
    PLATO_CONFIG_MANAGED_SERVER_NAME=`cat ${DOMAIN_HOME}/bin/plato-
config-deploy.env | grep "PLATO_CONFIG_MANAGED_SERVER_NAME" | cut -d=' ' -f2`
fi

if [ "${SERVER_NAME}" = "${PLATO_CONFIG_MANAGED_SERVER_NAME}" ] ; then
    # This will get executed only for Plato-config service entries
    config_file="${plato_config_file}"
else
    # This will get executed for all other services
    config_file="${domain_config_file}"
fi

if [ -f "$config_file" ]
then
    while read -r prop || [ -n "$prop" ]
    do
        case "$prop" in \#*) continue ;; esac
        if [ -z "${prop}" ] ; then
            continue
        else
            PLACEHOLDERS=${PLACEHOLDERS}" $(echo -D$prop)
            PLACEHOLDERS=${PLACEHOLDERS}"
        fi
    done
fi
```

```

done < "$config_file"
else
    echo "$config_file not found. please provide the property file to set -D parameter"
    exit 1
fi

PLACEHOLDERS="${PLACEHOLDERS}"

JAVA_OPTIONS="${JAVA_OPTIONS}${PLACEHOLDERS}"

export JAVA_OPTIONS

echo "${JAVA_OPTIONS}"

```

2. Place the **env** files containing all the key value pairs of the **-D params** in the respective <domain>/env folder.

NOTE: The plato-config-deploy.env file contains all the key value pairs specific only to the plato-config-service and need to be placed in the bin folder of the plato-domain. The domain-config-deploy.env file contains the key-value pairs for the rest of the services and should be placed in each <domain>/bin folder.

The sample for each of the files are given below:

plato-config-deploy.env

```

### Managed server name of plato-config service ###
PLATO_CONFIG_MANAGED_SERVER_NAME=

### plato config flyway connection entries ###
flywayTask=migrate
flyway.enabled=true
spring.flyway.enabled=false
plato-config.flyway.domain.db.username=
plato-config.flyway.domain.db.password=
plato-config.flyway.domain.db.jdbcUrl=
plato-config.flyway.domain.schemas=
plato-
config.flyway.domain.locations=db/migration/domain/plato,db/migration/domain/sms,db/migra
tion/domain/cmc,db/migration/domain/obvam

#### Kafka properties for all services ####
flyway.domain.placeholders.plato.eventhub.broker.hosts=
flyway.domain.placeholders.plato.eventhub.zookeeper.hosts=

#### Kafka Security for all services ####
flyway.domain.placeholders.plato.eventhub.broker.hosts=
flyway.domain.placeholders.plato.eventhub.zookeeper.hosts=
flyway.domain.placeholders.kafka.ssl.truststore.location=

```

```

flyway.domain.placeholders.kafka.ssl.truststore.password=
flyway.domain.placeholders.kafka.broker.username=
flyway.domain.placeholders.kafka.broker.password=

### common entries for all services ###
flyway.domain.placeholders.driver.className=oracle.jdbc.driver.OracleDriver
spring.cloud.loadbalancer.ribbon.enabled=false
spring.main.allow-circular-references=true

### eureka entries for all services ###
flyway.domain.placeholders.eureka.host=
flyway.domain.placeholders.eureka.port=

### zipkin entries for all services ###
flyway.domain.placeholders.zipkin.host=
flyway.domain.placeholders.zipkin.port=

### plato config flyway placeholder entries ###
flyway.domain.placeholders.plato-config.username=
flyway.domain.placeholders.plato-config.password=
flyway.domain.placeholders.plato-config.jdbcUrl=
flyway.domain.placeholders.plato-config.schemas=
flyway.domain.placeholders.plato-config.sessionIdleTimeout=
flyway.domain.placeholders.plato-config.sessionIdleWarningTime=
flyway.domain.placeholders.plato-config.environment=

### plato api-gateway flyway placeholder entries ###
flyway.domain.placeholders.api-gateway.host=
flyway.domain.placeholders.api-gateway.username=
flyway.domain.placeholders.api-gateway.password=
flyway.domain.placeholders.api-gateway.jdbcUrl=
flyway.domain.placeholders.api-gateway.schemas=
flyway.domain.placeholders.plato-api-gateway.server.port=

### plato api-gateway LDAP flyway placeholder entries ###
flyway.domain.placeholders.USER.STORE=
flyway.domain.placeholders.LDAP.CORS.allowed.origin=
flyway.domain.placeholders.LDAP.credential.SALT=
flyway.domain.placeholders.JWT.EXPIRY.seconds=
flyway.domain.placeholders.LDAP.url=
flyway.domain.placeholders.LDAP.userId=
flyway.domain.placeholders.LDAP.server.base=

```

```

flyway.domain.placeholders.LDAP.server.credential=
flyway.domain.placeholders.LDAP.usersearch.base=
flyway.domain.placeholders.LDAP.user.prefix=
# Allowed values for LDAP provider are: EMBEDDED_WEBLOGIC, PLATO
# If LDAP is running in weblogic then value should be EMBEDDED_WEBLOGIC
# If spring based LDAP(which is run through a jar provided) is used, then the value should be
PLATO
flyway.domain.placeholders.LDAP.provider=
flyway.domain.placeholders.TOKEN.autoregenerate=
flyway.domain.placeholders.SSO.enabled=
flyway.domain.placeholders.TOKEN.regeneration.enabled=

### plato-ui-config flyway placeholder entries ###
flyway.domain.placeholders.plato-ui-config.username=
flyway.domain.placeholders.plato-ui-config.password=
flyway.domain.placeholders.plato-ui-config.jdbcUrl=
flyway.domain.placeholders.plato-ui-config.schemas=
flyway.domain.placeholders.plato-ui-config-services.server.port=
flyway.domain.placeholders.apigateway.host=
flyway.domain.placeholders.apigateway.port=

### plato-discovery flyway placeholder entries ###
flyway.domain.placeholders.plato-discovery-service.server.port=

### plato-apigateway-router flyway placeholder entries ###
flyway.domain.placeholders.plato-apigateway-router.server.port=
flyway.domain.placeholders.plato-apigateway-router.router.protocol=
flyway.domain.placeholders.plato-apigateway-router.router.meadmin.port=

### plato-orch flyway placeholder entries ###
flyway.domain.placeholders.plato-orch-service.server.port=
flyway.domain.placeholders.plato-orchestrator.hostname=

### plato-feed flyway placeholder entries ###
flyway.domain.placeholders.plato-feed-services.username=
flyway.domain.placeholders.plato-feed-services.password=
flyway.domain.placeholders.plato-feed-services.jdbcUrl=
flyway.domain.placeholders.plato-feed-services.jndi=jdbc/PLATOFEEED
flyway.domain.placeholders.plato-feed-services.schemas=
flyway.domain.placeholders.plato-feed-services.feed.upload.directory=
flyway.domain.placeholders.plato-feed-services.server.port=

### plato-batch flyway placeholder entries ###

```

```

flyway.domain.placeholders.plato-batch-server.username=
flyway.domain.placeholders.plato-batch-server.password=
flyway.domain.placeholders.plato-batch-server.jdbcUrl=
flyway.domain.placeholders.plato-batch-server.schemas=
flyway.domain.placeholders.plato-batch-server.server.port=
flyway.domain.placeholders.plato-batch-server.plato.eventhub.kafka.brokers=
flyway.domain.placeholders.plato-batch-server.plato.eventhub.zk.nodes=
flyway.domain.placeholders.plato-batch-server.jndi=jdbc/PLATOBATCH

### plato-alerts-management flyway placeholder entries ###
flyway.domain.placeholders.plato-alerts-management-services.username=
flyway.domain.placeholders.plato-alerts-management-services.password=
flyway.domain.placeholders.plato-alerts-management-services.jdbcUrl=
flyway.domain.placeholders.plato-alerts-management-services.schemas=
flyway.domain.placeholders.plato-alerts-management-services.server.port=

### sms flyway placeholder entries ###
flyway.domain.placeholders.sms-core-services.server.port=
flyway.domain.placeholders.sms.username=
flyway.domain.placeholders.sms.password=
flyway.domain.placeholders.sms.jdbcUrl=
flyway.domain.placeholders.sms.schemas=

### cmncore flyway placeholder entries ###
flyway.domain.placeholders.cmncore.username=
flyway.domain.placeholders.cmncore.password=
flyway.domain.placeholders.cmncore.jdbcUrl=
flyway.domain.placeholders.cmncore.schemas=
flyway.domain.placeholders.cmc-corebanking-adapter-service.server.port=
flyway.domain.placeholders.cmc-currency-services.server.port=
flyway.domain.placeholders.cmc-account-services.server.port=
flyway.domain.placeholders.cmc-base-services.server.port=
flyway.domain.placeholders.cmc-external-virtual-account-services.server.port=
flyway.domain.placeholders.cmc-branch-services.server.port=
flyway.domain.placeholders.cmc-customer-services.server.port=
flyway.domain.placeholders.cmc-external-chart-account-services.server.port=
flyway.domain.placeholders.cmc-external-system-services.server.port=
flyway.domain.placeholders.cmc-advice-services.server.port=
flyway.domain.placeholders.cmc-facilities-services.server.port=
flyway.domain.placeholders.cmc-txn-code-services.server.port=

```

flyway.domain.placeholders.cmc-settlement-services.server.port=
 flyway.domain.placeholders.cmc-businessoverrides-services.server.port=
 flyway.domain.placeholders.cmc-resource-segment-orchestrator-service.server.port=
 flyway.domain.placeholders.cmc-screenclass-services.server.port=
 flyway.domain.placeholders.cmc-datasegment-services.server.port=
 flyway.domain.placeholders.cmc-settlements-services.server.port=
 flyway.domain.placeholders.cmc-transactioncontroller-services.server.port=
 flyway.domain.placeholders.cmc-report-services.server.port=
 flyway.domain.placeholders.cmc-nlp-annotator-services.server.port=
 flyway.domain.placeholders.cmc-nlp-dashboard-widget-services.server.port=
 flyway.domain.placeholders.cmc-nlp-model-mngmnt-services.server.port=
 flyway.domain.placeholders.cmc-nlp-online-processing-services.server.port=
 flyway.domain.placeholders.cmc-nlp-tag-maint-services.server.port=
 flyway.domain.placeholders.cmc-nlp-text-extraction-services.server.port=
 flyway.domain.placeholders.cmc-nlp-txn-log-services.server.port=
 flyway.domain.placeholders.cmc-nlp-util-services.server.port=
 flyway.domain.placeholders.cmc-batch-services.server.port=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.server.port=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.postingPath=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.pollingEmail=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.emailServerPort=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.emailServerHost=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.pollingFrequency=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.pollerInitialDelay=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.emailPassword=
 flyway.domain.placeholders.cmc-fc-ai-ml-services.pollingPath=

biPublisher related cmc-report-service entries

flyway.domain.placeholders.weblogic.userid=
 flyway.domain.placeholders.weblogic.password=
 flyway.domain.placeholders.biPublisher.host=
 flyway.domain.placeholders.biPublisher.port=
 flyway.domain.placeholders.runReportTemplate=
 flyway.domain.placeholders.emailTemplate=
 flyway.domain.placeholders.dms.host=
 flyway.domain.placeholders.dms.port=

flyway jndi connection details for shared services placeholder entries

flyway.domain.placeholders.plato.jndi=jdbc/PLATO
 flyway.domain.placeholders.plato-config.jndi=jdbc/PLATO

flyway.domain.placeholders.plato-sec.jndi=jdbc/PLATO_SECURITY
flyway.domain.placeholders.plato-ui-config.jndi=jdbc/PLATO_UI_CONFIG
flyway.domain.placeholders.sms.jndi=jdbc/sms
flyway.domain.placeholders.cmncore.jndi=jdbc/CMNCORE

flyway jndi connection details for obvam services placeholder entries

flyway.domain.placeholders.eie.jndi=jdbc/EIE
flyway.domain.placeholders.eie.server.port=
flyway.domain.placeholders.eie.schemas=

flyway.domain.placeholders.elm.jndi=jdbc/ELM
flyway.domain.placeholders.elm.server.port=
flyway.domain.placeholders.elm.schemas=

flyway.domain.placeholders.vam.jndi=jdbc/VAM
flyway.domain.placeholders.vam.server.port=
flyway.domain.placeholders.vam.schemas=

flyway.domain.placeholders.vac.jndi=jdbc/VAC
flyway.domain.placeholders.vac.server.port=
flyway.domain.placeholders.vac.schemas=

flyway.domain.placeholders.vab.jndi=jdbc/VAB
flyway.domain.placeholders.vab.server.port=
flyway.domain.placeholders.vab.schemas=

flyway.domain.placeholders.vae.jndi=jdbc/VAE
flyway.domain.placeholders.vae.server.port=
flyway.domain.placeholders.vae.schemas=

flyway.domain.placeholders.eda.jndi=jdbc/EDA
flyway.domain.placeholders.eda.server.port=
flyway.domain.placeholders.eda.schemas=

flyway.domain.placeholders.vai.jndi=jdbc/VAI
flyway.domain.placeholders.vai.server.port=
flyway.domain.placeholders.vai.schemas=

flyway.domain.placeholders.van.jndi=jdbc/VAN
flyway.domain.placeholders.van.server.port=
flyway.domain.placeholders.van.schemas=

flyway.domain.placeholders.vap.jndi=jdbc/VAP
flyway.domain.placeholders.vap.server.port=
flyway.domain.placeholders.vap.schemas=

flyway.domain.placeholders.vas.jndi=jdbc/VAS
flyway.domain.placeholders.vas.server.port=
flyway.domain.placeholders.vas.schemas=

flyway.domain.placeholders.vat.jndi=jdbc/VAT
flyway.domain.placeholders.vat.server.port=
flyway.domain.placeholders.vat.schemas=

flyway.domain.placeholders.vaj.server.port=

flyway.domain.placeholders.platoorch.domain.jndi=jdbc/PLATO-O
flyway.domain.placeholders.platoorch.domain.schemas=

flyway.domain.placeholders.plato.alerts.email.userId=
flyway.domain.placeholders.plato.alerts.email.password=
flyway.domain.placeholders.plato.alerts.cmc.userId=
flyway.domain.placeholders.plato.alerts.cmc.branchCode=
flyway.domain.placeholders.plato.alerts.cmc.applId=
flyway.domain.placeholders.plato-rule.hostname=
flyway.domain.placeholders.plato-rule-service.server.port=
flyway.domain.placeholders.platorule.domain.jndi=
flyway.domain.placeholders.platorule.domain.schemas=
flyway.domain.placeholders.obrh.import.data.disable-modify=
flyway.domain.placeholders.cmc-obrh-services.kafka.server.path=
flyway.domain.placeholders.cmc-obrh-services.zookeeper.server.path=
flyway.domain.placeholders.cmc.schemas=
flyway.domain.placeholders.cmc-nlp-opennlp-services.server.port=
flyway.domain.placeholders.cmc-nlp-maintenance-services.server.port=
flyway.domain.placeholders.cmc-nlp-pipeline-services.server.port=
flyway.domain.placeholders.cmc-nlp-docview-services.server.port=
flyway.domain.placeholders.cmc-ml-indb-services.server.port=
flyway.domain.placeholders.cmc-obrh-services.kafka.enabled=
flyway.domain.placeholders.cmc-sla-services.server.port=
flyway.domain.placeholders.cmc-obcbs-services.schemas=

```

flyway.domain.placeholders.obcbs.server.port=
flyway.domain.placeholders.orch.cmc.brn=
flyway.domain.placeholders.orch.cmc.user=
flyway.domain.placeholders.orch.enableDynamicAllocation=
flyway.domain.placeholders.orch.enableSLA=
flyway.domain.placeholders.report-service.server.port=
flyway.domain.placeholders.report-service.hostname=
flyway.domain.placeholders.report-service.domain.jndi=jdbc/PLATOREPORT
flyway.domain.placeholders.report-service.template-metadata-directory=
flyway.domain.placeholders.report-service.output-directory=
flyway.domain.placeholders.report-service.fop-config-file=

```

```

### generic entries for all services ###

```

```

spring.cloud.config.uri=
apigateway.url=
service.logging.environment=
service.logging.path=

```

domain-config-deploy.env

```

### domain config flyway connection entries ###

```

```

flywayTask=migrate
flyway.enabled=true
spring.flyway.enabled=false

```

```

### generic entries for all services ###

```

```

spring.cloud.config.uri=
apigateway.url=
service.logging.environment=
service.logging.path=

```

1.2.4 Method 4 – Workflow Configuration

Follow the below steps to create a workflow:

1. Metadata of the workflow creation. The sample DSL for workflow creation is given below:

```

{
  "name": "initialTest",
  "description": "Test workflow",

```

```

"version": 4,
"tasks": [
  {
    "name": "TEST",
    "taskReferenceName": "TESTING3",
    "description": "TESTING2",
    "inputParameters": {
      "FUNCTIONAL_CODE": "TEST_FA_ILS_REGTN2",
      "processRefNo":
"${workflow.input.transactionModel.txnIdentification.processRefNo}",
      "processName": "Testing Process2",
      "processCode":
"${workflow.input.transactionModel.txnIdentification.processName}",
      "transactionModel": "${workflow.input.transactionModel}",
      "stage": "TESTING2",
      "priority":
"${workflow.input.transactionModel.transactionData.moduleData.taskPriority}",
      "applicationDate":
"${workflow.input.transactionModel.txnIdentification.applicationDate}",
      "applicationNumber":
"${workflow.input.transactionModel.txnIdentification.processRefNo}",
      "processRefNumber":
"${workflow.input.transactionModel.txnIdentification.processRefNo}",
      "branch": "${workflow.input.transactionModel.txnIdentification.branchCode}",
      "user": "${workflow.input.transactionModel.txnIdentification.currentUser}",
      "customerNumber":
"${workflow.input.transactionModel.transactionData.moduleData.customerId}",
      "amount":
"${workflow.input.transactionModel.transactionData.moduleData.amount}",
      "currencyCode":
"${workflow.input.transactionModel.transactionData.moduleData.currency}",
      "TASK_OUTCOMES": [
        "PROCEED"
      ],
      "moduleCode": "OBTFFPM",
      "customFilter": [
        {
          "key": "contractRefNo",
          "label": "Back Office Reference"
        }
      ]
    }
  }
]

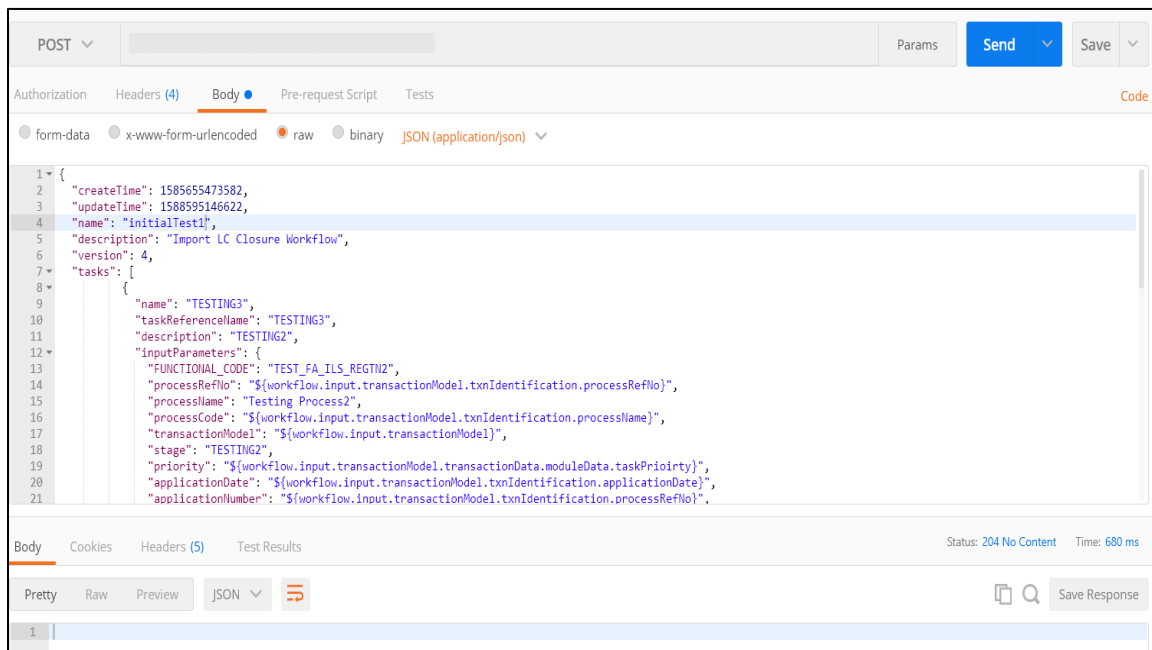
```

```

        "key": "otherRefNo",
        "label": "External Reference"
    }
]
},
"type": "WAIT",
"startDelay": 0,
"optional": false,
"asyncComplete": false
}
],
"outputParameters": {
"stage": "CLMO_FA_SNPOAR_APPEN",
"taskOutcome": "PROCEED_WITH_PARTICIPANT"
},
"schemaVersion": 2,
"restartable": true,
"workflowStatusListenerEnabled": false
}

```

Call the API (/api/metadata/workflow) and pass the DSL in body. The following screen depicts the sample workflow:



2. Workflow Creation

Call the API (/api/workflow) to create the workflow. This API provides the information to the workflow metadata which we have created using previous call.

Body:

```

{
  "name": "initialTest",
  "description": "initialTest",
  "version": 4,
  "input": {
    "transactionModel": {
      "transactionData": {
        "moduleData": {
          "amount": 122,
          "currency": "GBP",
          "customerId": "001506",
          "customerName": "MARKS AND SPENCER",
          "taskPriority": "H"
        }
      }
    },
    "txnIdentification": {
      "branchCode": "000",
      "currentStage": "TEST_FA_ILS_REGTN2",
      "currentUser": "SWAGATIKA",
      "key1": "Desk",
      "moduleCode": "TRMO",
      "processName": "Testing Process2",
      "processRefNo": "300ILCI012260",
      "applicationDate": 1588582461960,
      "taskOutcome": "PROCEED",
      "taskPriority": "H"
    }
  }
}

```

The following screen depicts the sample workflow:

POST <https://uhf00fnc-2000/nlata-cerb/api/workflow> Params Send Save

Authorization Headers (4) Body Pre-request Script Tests Code

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary JSON (application/json)

```
1 {
2   "name": "InitialTest1",
3   "description": "Import LC Closure Workflow",
4   "version": 4,
5   "input": {
6     "transactionModel": {
7       "transactionData": {
8         "moduleData": {
9           "amount": 122,
10          "currency": "GBP",
11          "customerId": "001506",
12          "customerName": "MARKS AND SPENCER",
13          "taskPriority": "H"
14        }
15      },
16      "txnIdentification": {
17        "branchCode": "000",
18        "currentStage": "TEST_FA_ILS_REGTN2",
19        "currentUser": "SNAGATIKA",
20        "key1": "Desk",
21        "moduleCode": "TRMD",
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 187272 ms

Pretty Raw Preview Text Save Response

```
1 151d78d7-6711-46ae-be15-bf7b550c4b36
```

1.3 How to Create Domain and Cluster Configuration

This section contains the following sub-sections:

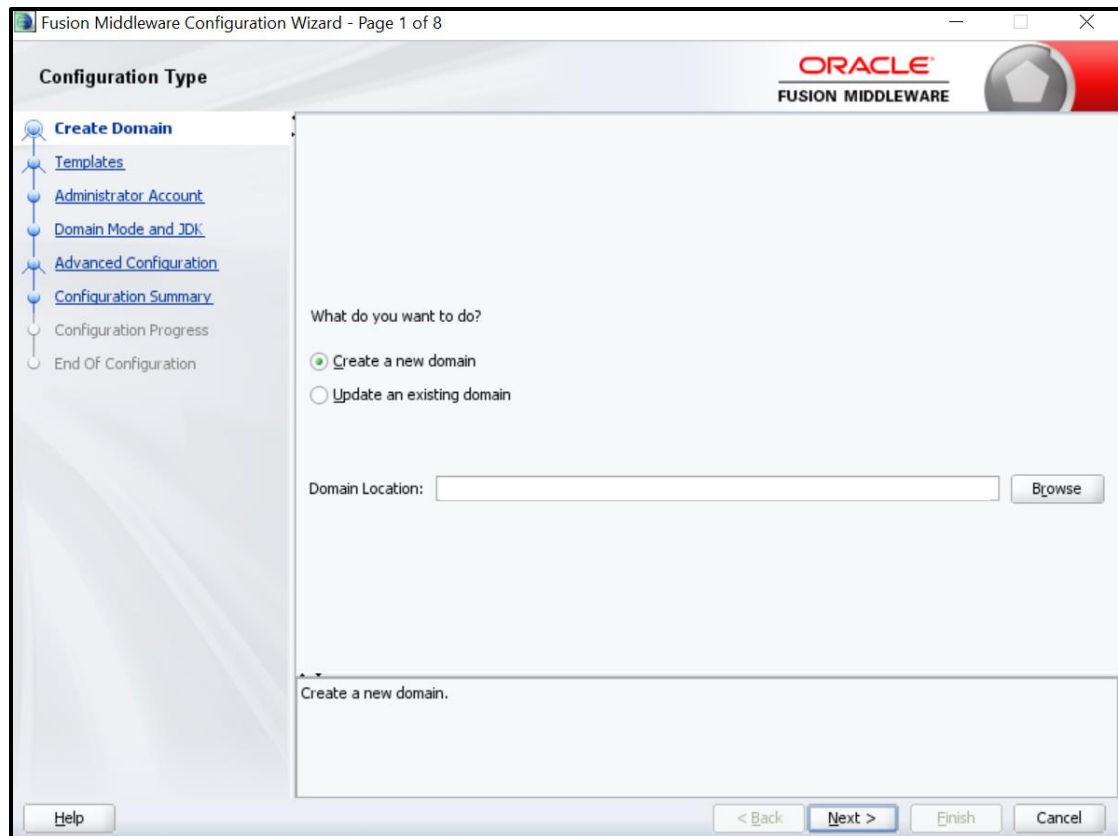
- Domain Creation Configuration
- Post Domain Creation Configurations

1.3.1 Domain Creation Configuration

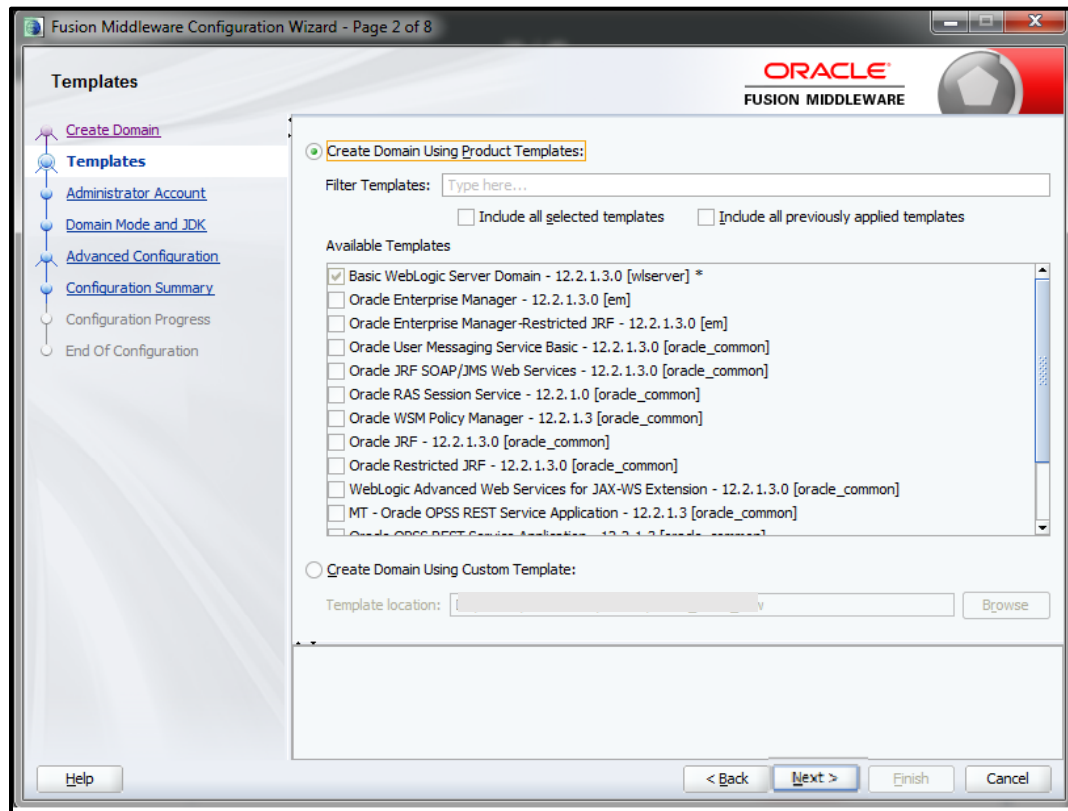
Perform the following steps for domain and cluster configuration:

NOTE: Name need not to be same as provided in Screenshot.

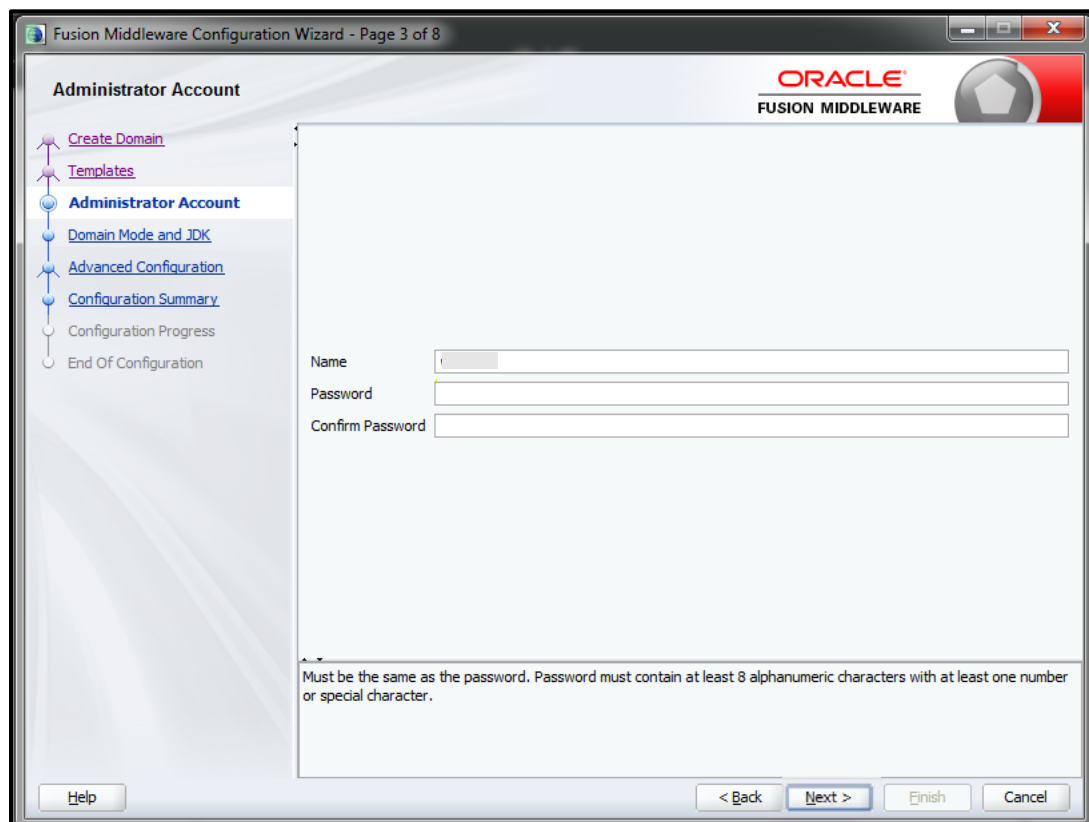
1. Open **/oracle_common/common/bin** and run **config.cmd** (or **.sh** if operating system is linux). Create domain with required cluster and server configurations. Refer to the screenshots below.
2. Select **Create a new domain** and provide domain name. For example, **platoinfra_domain**.



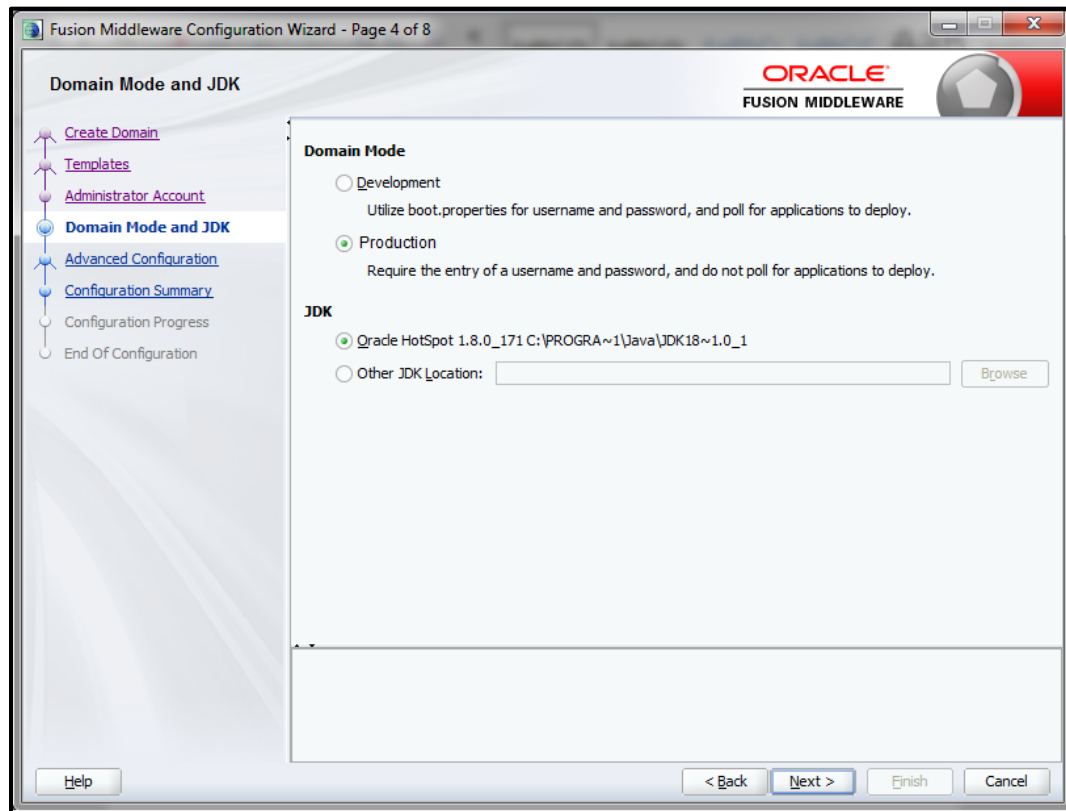
3. Click **Next** to create simple domain with default templates.



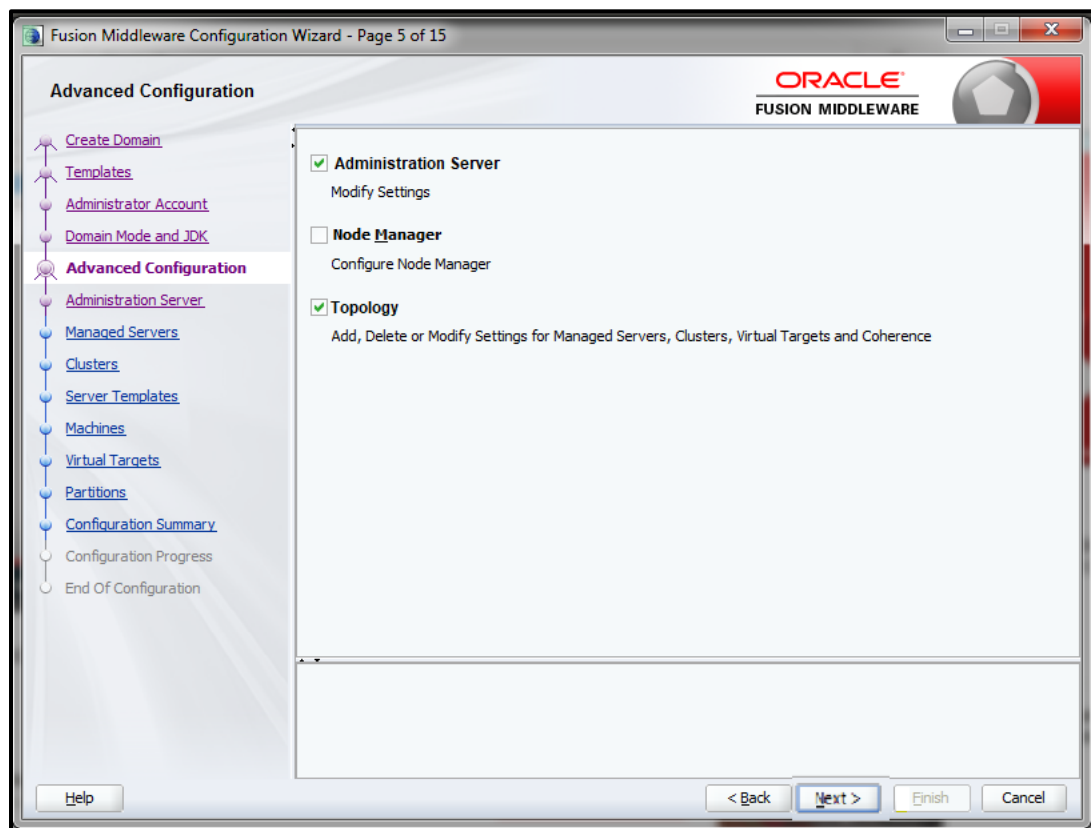
4. Set password and confirm, click **Next** to proceed.



5. Select **Domain Mode** as **Production** and select **JDK**.



6. Select **Administration Server** and **Topology** in advanced configurations.



7. Edit the port and host configurations as required and click **Next**.

The screenshot shows the 'Administration Server' configuration page. The left sidebar lists the configuration steps, with 'Administration Server' highlighted. The main area contains the following fields:

- Server Name: AdminServer
- Listen Address: All Local Addresses
- Listen Port: 7001
- Enable SSL: ☐
- SSL Listen Port: (empty)

A validation message at the bottom states: 'The name must not be null or empty and may not contain any : , = * ? % / _ cloned.'

Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

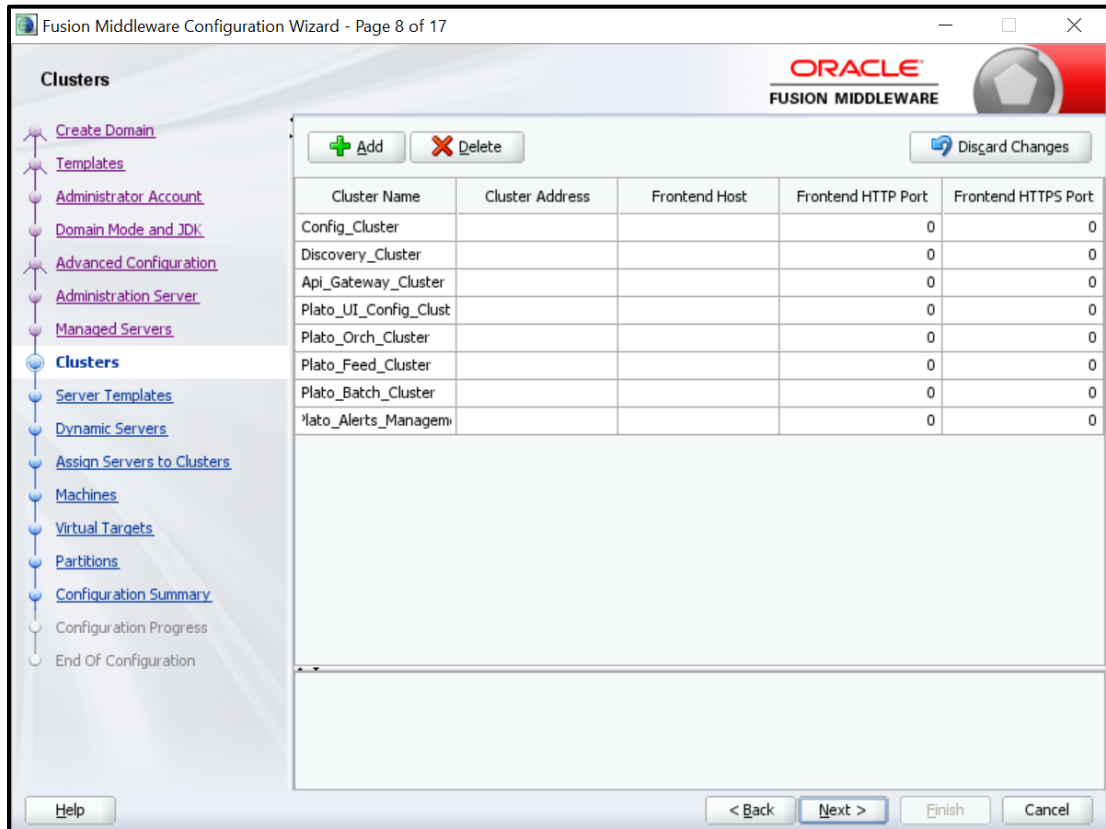
8. Add managed servers and provide meaningful **Server Name**, edit listen address and port as required.

The screenshot shows the 'Managed Servers' configuration page. The left sidebar lists the configuration steps, with 'Managed Servers' highlighted. The main area contains a table of managed servers and buttons to 'Add', 'Clone', 'Delete', and 'Discard Changes'.

Server Name	Listen Address	Listen Port	Enable SSL	SSL Listen Port
Config_Server	All Local Addresses	7003	<input type="checkbox"/>	Disabled
Discovery_Server	All Local Addresses	7004	<input type="checkbox"/>	Disabled
Api_Gateway_Server	All Local Addresses	7005	<input type="checkbox"/>	Disabled
Plato_UI_Config_Server	All Local Addresses	7006	<input type="checkbox"/>	Disabled
Plato_Orch_Server	All Local Addresses	7007	<input type="checkbox"/>	Disabled
Plato_Feed_Server	All Local Addresses	7008	<input type="checkbox"/>	Disabled
Plato_Batch_Server	All Local Addresses	7009	<input type="checkbox"/>	Disabled
Plato_Alerts_Management_Se	All Local Addresses	7010	<input type="checkbox"/>	Disabled

Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

9. Add clusters one for each **managed servers**.

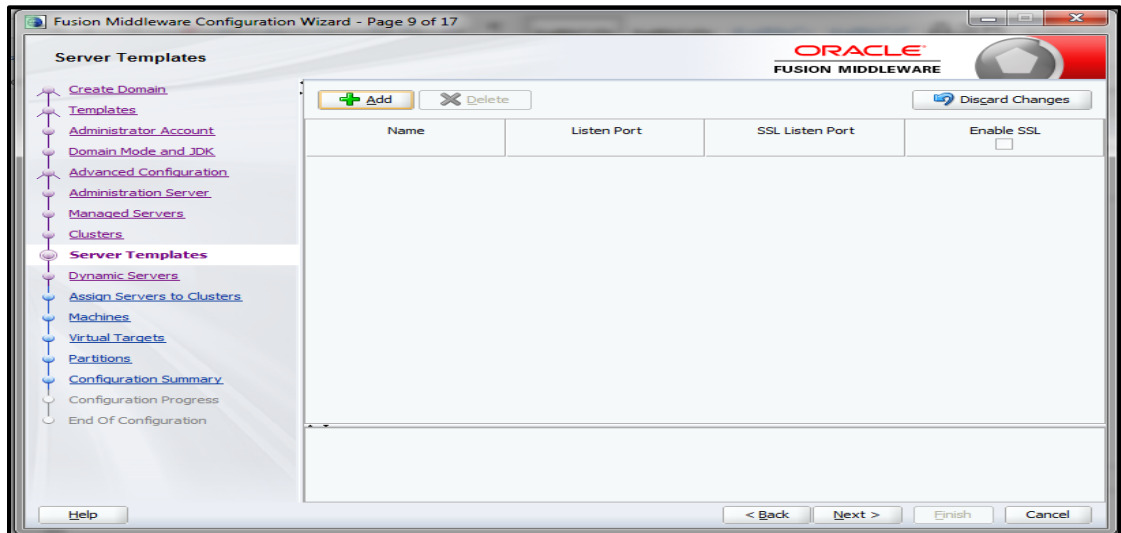


The screenshot shows the 'Clusters' page of the Fusion Middleware Configuration Wizard. The left sidebar contains a tree view with the following items: Create Domain, Templates, Administrator Account, Domain Mode and JDK, Advanced Configuration, Administration Server, Managed Servers, **Clusters** (selected), Server Templates, Dynamic Servers, Assign Servers to Clusters, Machines, Virtual Targets, Partitions, Configuration Summary, Configuration Progress, and End Of Configuration. The main area displays a table with the following columns: Cluster Name, Cluster Address, Frontend Host, Frontend HTTP Port, and Frontend HTTPS Port. The table contains the following data:

Cluster Name	Cluster Address	Frontend Host	Frontend HTTP Port	Frontend HTTPS Port
Config_Cluster			0	0
Discovery_Cluster			0	0
Api_Gateway_Cluster			0	0
Plato_UI_Config_Cluster			0	0
Plato_Orch_Cluster			0	0
Plato_Feed_Cluster			0	0
Plato_Batch_Cluster			0	0
Plato_Alerts_Managem			0	0

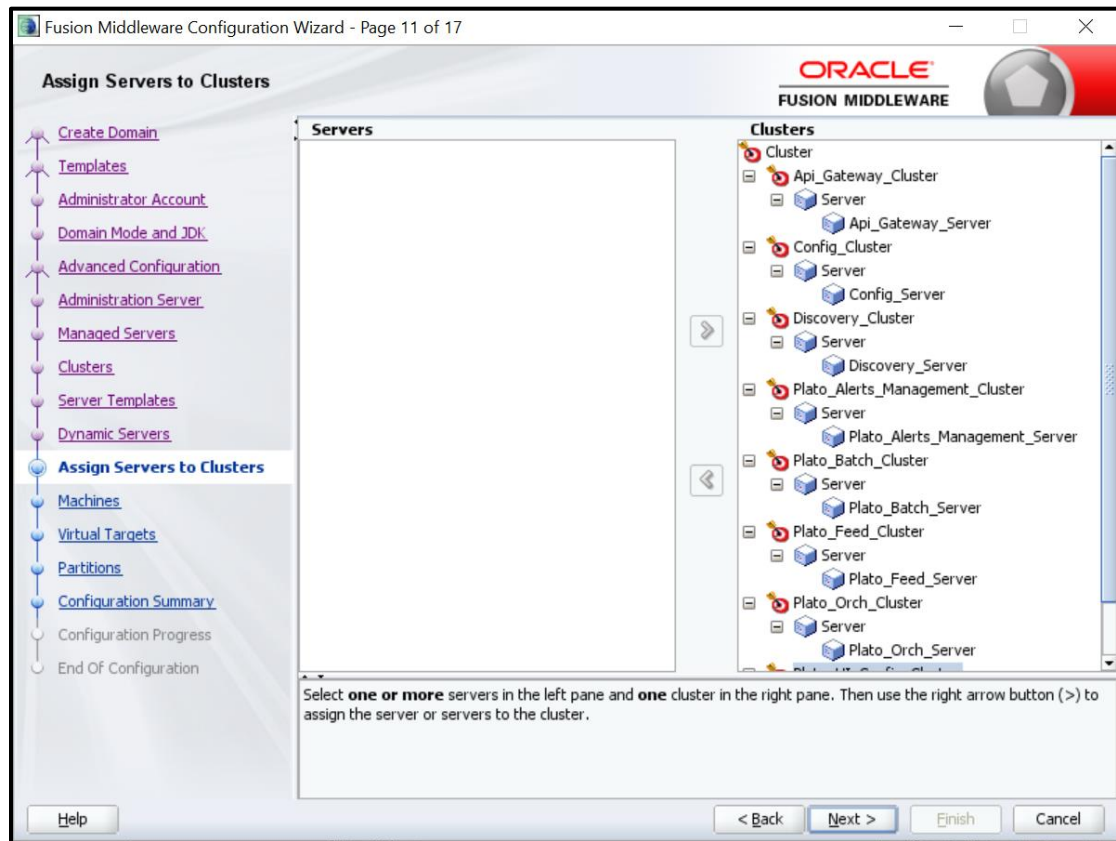
At the top of the main area, there are buttons for '+ Add', 'X Delete', and 'Discard Changes'. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. A 'Help' button is located in the bottom left corner.

10. Skip **Server Templates** and **Dynamic Servers**.

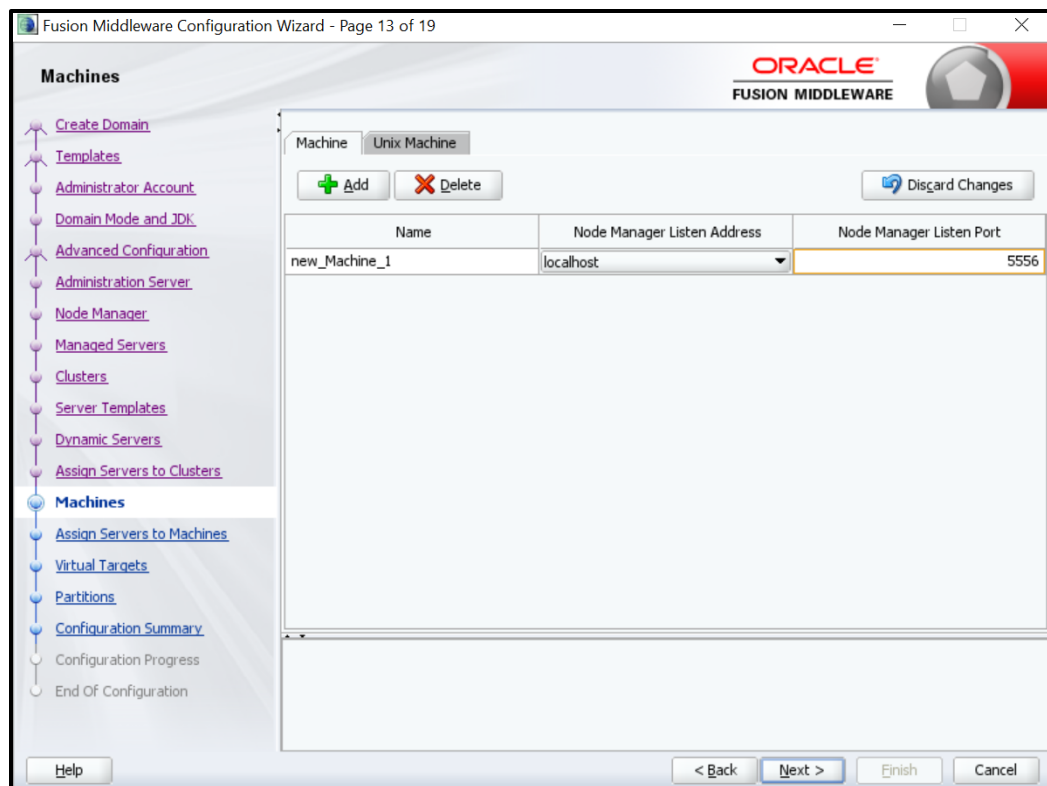


The screenshot shows the 'Server Templates' page of the Fusion Middleware Configuration Wizard. The left sidebar contains a tree view with the following items: Create Domain, Templates, Administrator Account, Domain Mode and JDK, Advanced Configuration, Administration Server, Managed Servers, Clusters, **Server Templates** (selected), Dynamic Servers, Assign Servers to Clusters, Machines, Virtual Targets, Partitions, Configuration Summary, Configuration Progress, and End Of Configuration. The main area displays a table with the following columns: Name, Listen Port, SSL Listen Port, and Enable SSL. The table is currently empty. At the top of the main area, there are buttons for '+ Add', 'X Delete', and 'Discard Changes'. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. A 'Help' button is located in the bottom left corner.

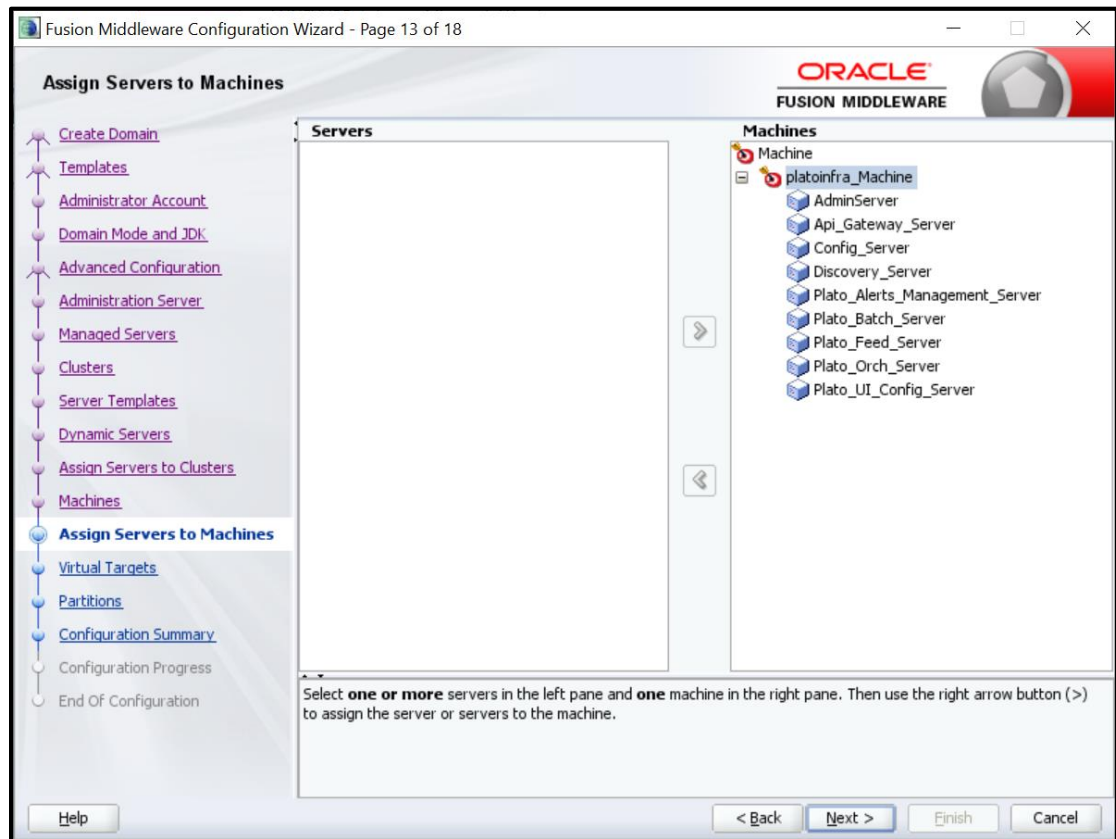
11. Assign clusters with servers.



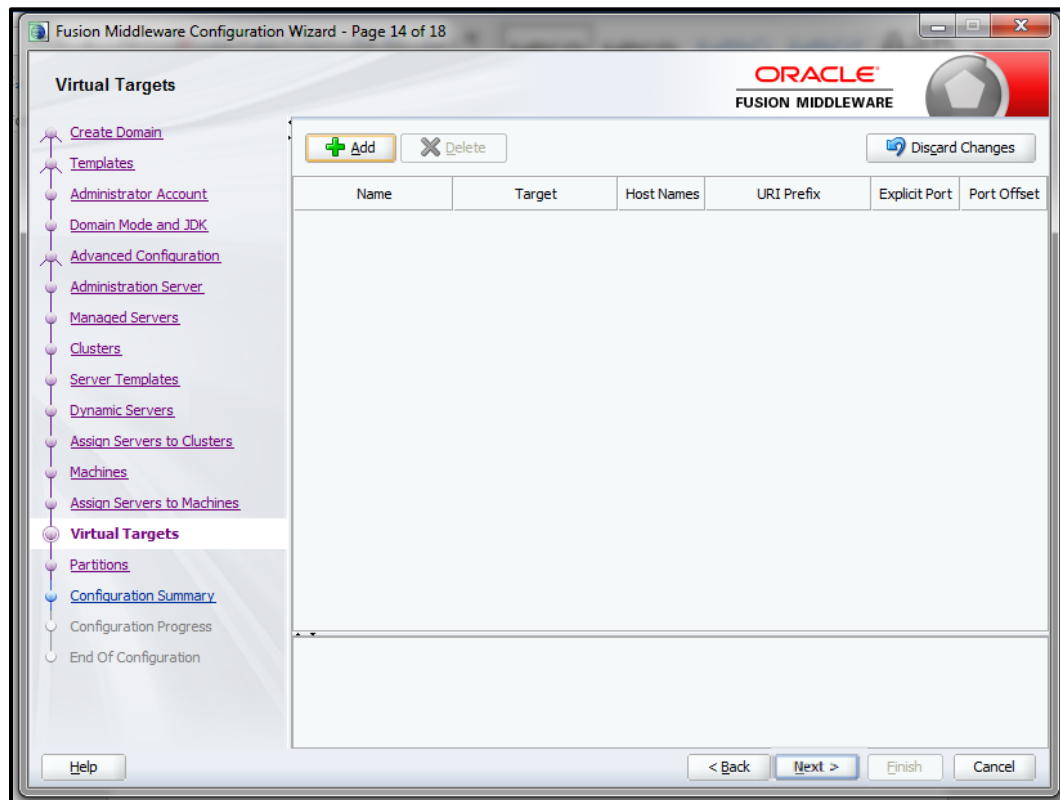
12. Add **Machine/Unix Machine** based on operating system and configure **Name**, **Node Manager Listen Address** and **Node Manager Listen Port** as required.

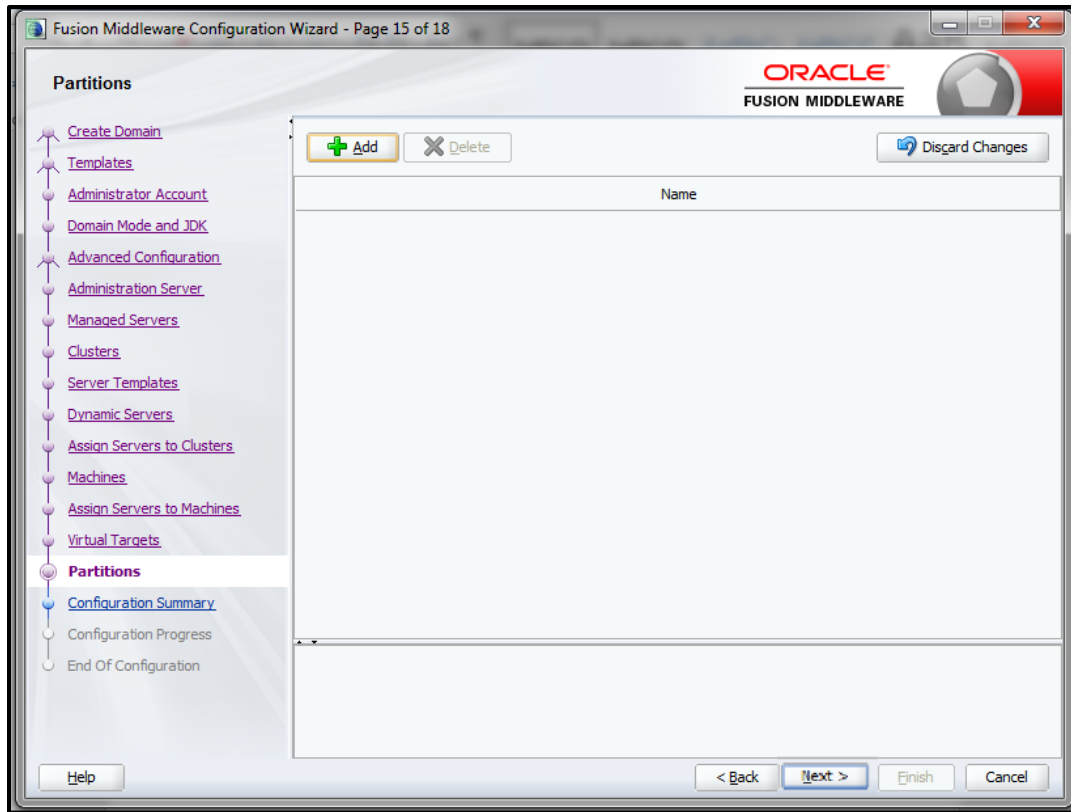


13. Map all managed servers under the machine created.

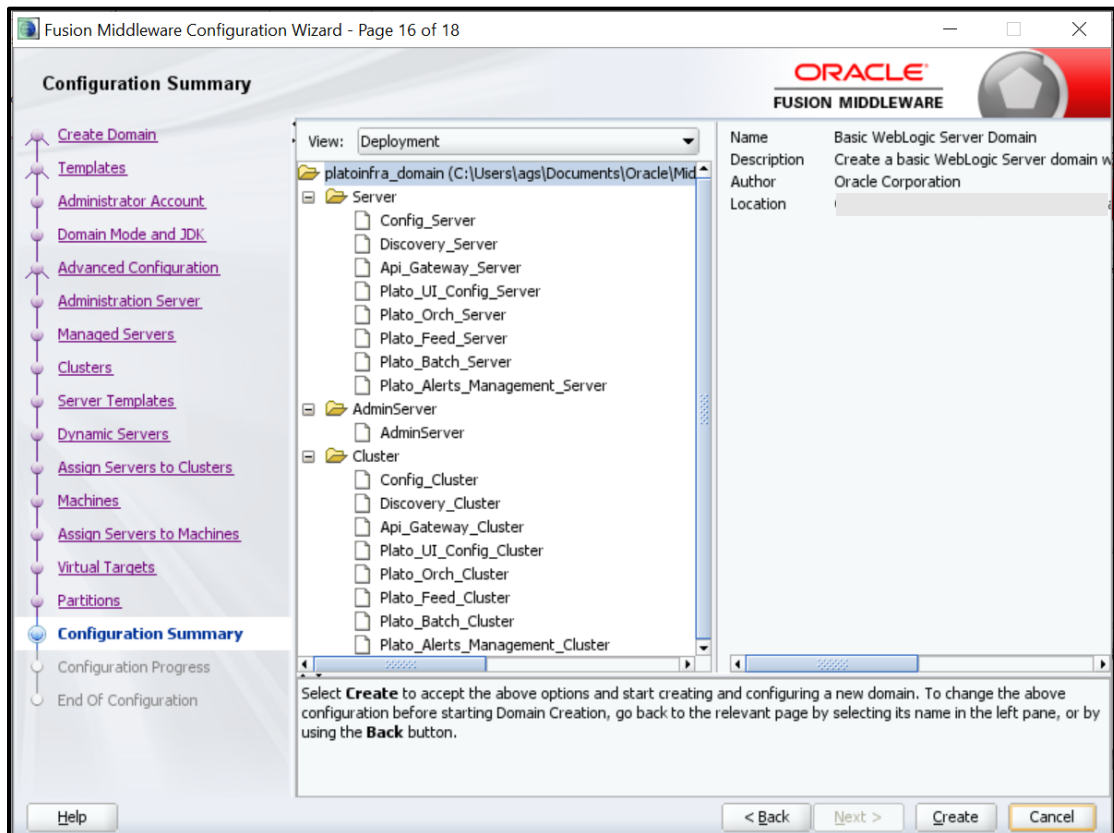


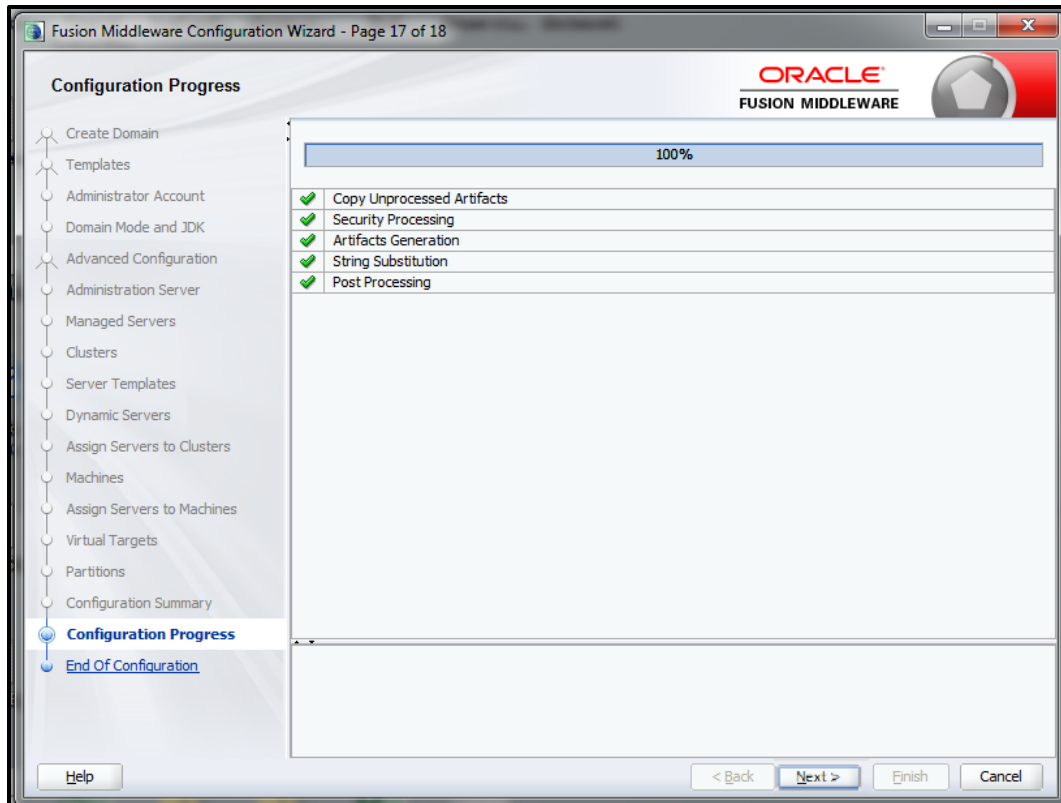
14. Skip or configure **Virtual Targets** and **Partitions** as required.



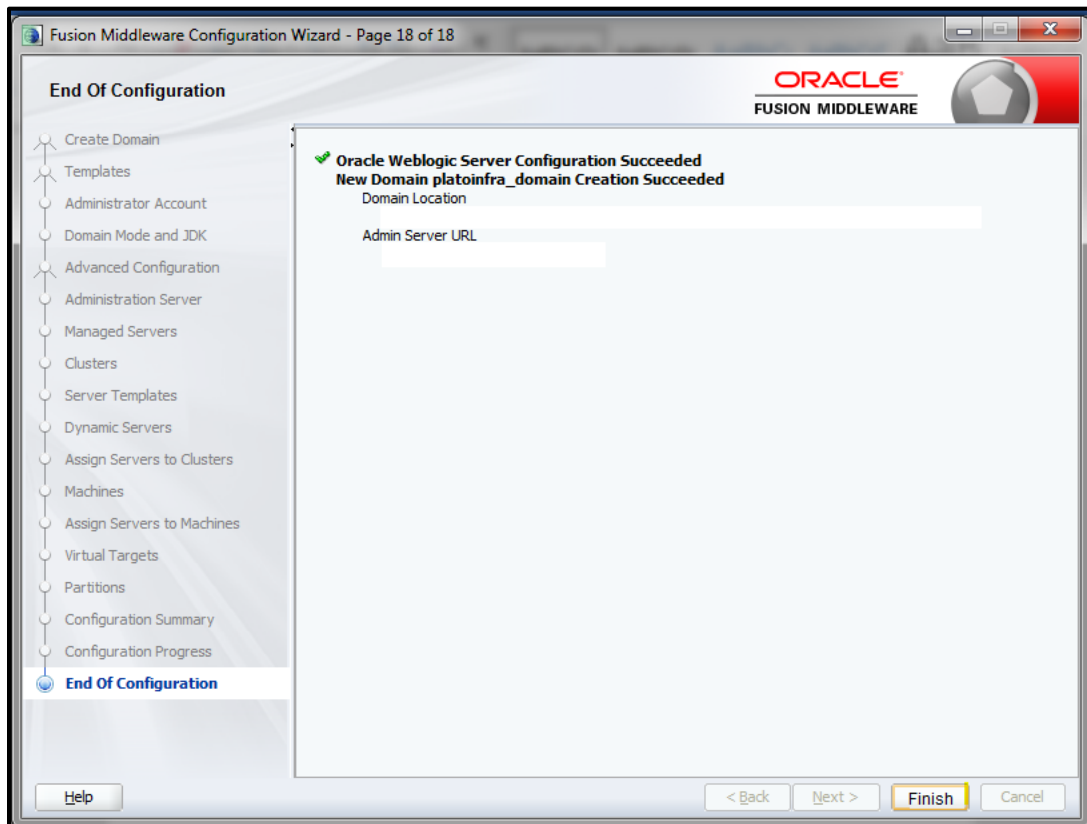


15. Check the **Configuration Summary** and confirm creating domain.





16. Click **Finish** to complete the procedure.



1.3.2 Post Domain Creation Configurations

Once finished, refer oracle fusion middleware documents for more details on how to start admin server, node manager and managed servers.

1. Open `/user_projects/domain/platoinfra_domain/bin`
2. Perform all the Environment Setup steps such as setting -D parameters, Embedded Weblogic Setup and changes required for OAuth.
3. Run `startWeblogic.cmd` (or `.sh` if operating system is linux).
4. Open `/user_projects/domains/platoinfra_domain/bin`.
5. Run `setNMJavaHome.cmd` (or `.sh` if operating system is linux).
6. Open `/user_projects/domains/platoinfra_domain/nodemanager`.
7. Edit `nodemanager.properties` as required (securelistner = false if ssl and keystore is not given).
8. In admin console, select the following options in sequential order:
 - a. **Machines**
 - b. **platoinfra_Machine**
 - c. **Node Manager**
 - d. **Type**
 - e. **Plain**
 - f. **Save**
9. Open `/user_projects/domains/platoinfra_domain/bin`.
10. Run `startNodeManager.cmd` (or `.sh` if operating system is linux)
11. Start all managed servers.
12. Login to console and verify servers and clusters. Refer to the screenshots below:

Configuration | Control

A server is an instance of WebLogic Server that runs in its own Java Virtual Machine (JVM) and has its own configuration. This page summarizes each server that has been configured in the current WebLogic Server domain.

[Customize this table](#)

Servers (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

Name	Type	Cluster	Machine	State	Health	Listen Port
AdminServer(admin)	Configured		platoinfra_Machine	RUNNING	OK	7001
Api_Gateway_Server	Configured	Api_Gateway_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7005
Config_Server	Configured	Config_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7003
Discovery_Server	Configured	Discovery_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7004
Plato_Alerts_Management_Server	Configured	Plato_Alerts_Management_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7010
Plato_Batch_Server	Configured	Plato_Batch_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7009
Plato_Feed_Server	Configured	Plato_Feed_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7008
Plato_Orch_Server	Configured	Plato_Orch_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7007
Plato_UI_Config_Server	Configured	Plato_UI_Config_Cluster	platoinfra_Machine	SHUTDOWN	Not reachable	7006

Change Center

View changes and restarts

Click the **Lock & Edit** button to modify, add or delete items in this domain.

Lock & Edit

Release Configuration

Domain Structure

platoinfra_domain

Domain Partitions

Environment

Servers

Clusters

Coherence Clusters

Resource Groups

Resource Group Templates

Machines

Virtual Hosts

Virtual Targets

Work Managers

Concurrent Templates

Resource Management

How do I...

Configure clusters

Assign server instances to clusters

Configure server migration in a cluster

Configure cross-cluster replication

Create dynamic clusters

Home Log Out Preferences Add Record Help

Welcome, Connected to: platoinfra_domain

Home > Summary of Deployments > Summary of Servers > Summary of Clusters

Summary of Clusters

This page summarizes the clusters that have been configured in the current WebLogic Server domain.

A cluster defines groups of WebLogic Server servers that work together to increase scalability and reliability.

Customize this table

Clusters (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

New Clone Delete

Showing 1 to 8 of 8 Previous Next

	Name	Cluster Address	Cluster Messaging Mode	Migration Basis	Default Load Algorithm	Replication Type	Cluster Broadcast Channel	S
	Api_Gateway_Cluster		Unicast	Database	Round Robin	(None)		A
	Config_Cluster		Unicast	Database	Round Robin	(None)		C
	Discovery_Cluster		Unicast	Database	Round Robin	(None)		D
	Plato_Alerts_Management_Cluster		Unicast	Database	Round Robin	(None)		P
	Plato_Batch_Cluster		Unicast	Database	Round Robin	(None)		P
	Plato_Feed_Cluster		Unicast	Database	Round Robin	(None)		P
	Plato_Orch_Cluster		Unicast	Database	Round Robin	(None)		P
	Plato_UI_Config_Cluster		Unicast	Database	Round Robin	(None)		P

New Clone Delete

Showing 1 to 8 of 8 Previous Next

Change Center

View changes and restarts

Click the **Lock & Edit** button to modify, add or delete items in this domain.

Lock & Edit

Release Configuration

Domain Structure

platoinfra_domain

Domain Partitions

Environment

Servers

Clusters

Coherence Clusters

Resource Groups

Resource Group Templates

Machines

Virtual Hosts

Virtual Targets

Work Managers

Concurrent Templates

Resource Management

How do I...

Configure clusters

Assign server instances to clusters

Configure server migration in a cluster

Configure cross-cluster replication

Create dynamic clusters

Home Log Out Preferences Add Record Help

Welcome, Connected to: platoinfra_domain

Home > Summary of Deployments > Summary of Servers > Summary of Clusters > Summary of Machines

Summary of Machines

A machine is the logical representation of the computer that hosts one or more WebLogic Server instances (servers). WebLogic Server uses configured machine names to determine the optimum server in a cluster to which certain tasks, such as HTTP session replication, are delegated. The Administration Server uses the machine definition in conjunction with Node Manager to start remote servers.

This page displays key information about each machine that has been configured in the current WebLogic Server domain.

Customize this table

Machines

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

New Clone Delete

Showing 1 to 1 of 1 Previous Next

	Name	Type
	platoinfra_Machine	Machine

New Clone Delete

Showing 1 to 1 of 1 Previous Next

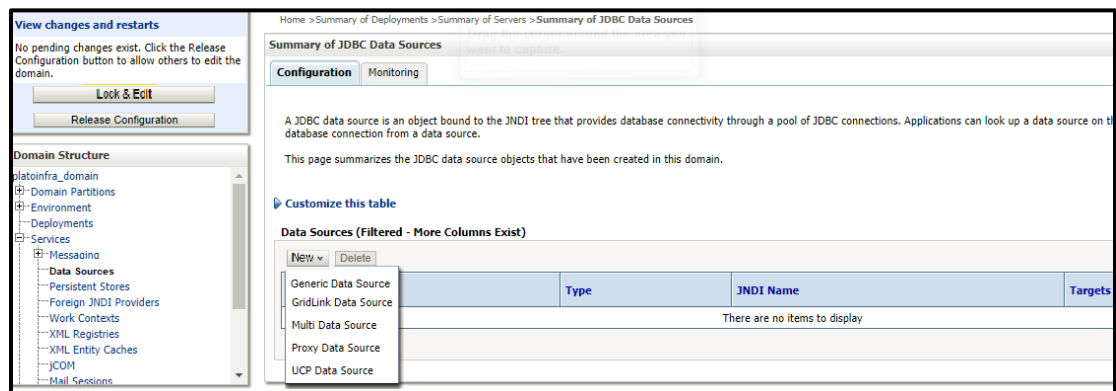
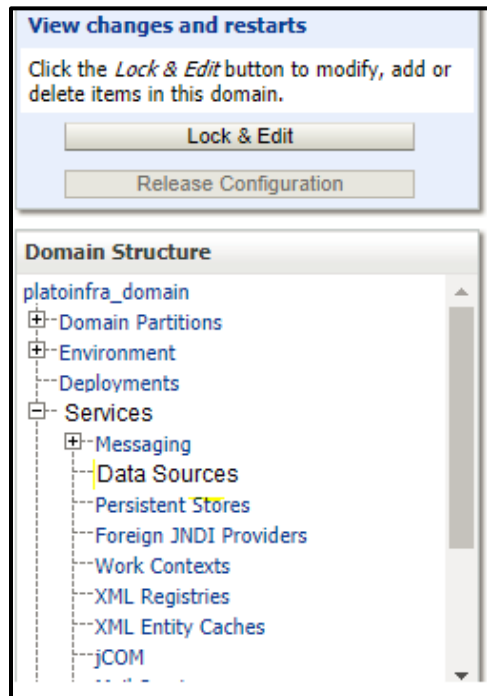
1-31

ORACLE

1.4 How to Create Datasource

Perform the following steps to create data source:

1. Start **AdminServer**, **Node Manager** and make sure all the **managed servers** (targets) are in running mode.
2. Select the following options in sequential order:
 - a. **Services**
 - b. **Datasources**
 - c. **New**
 - d. **Generic Datasource**



3. Give datasource **Name** and **JNDI Name**, and click **Next**.

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.

* Indicates required fields

What would you like to name your new JDBC data source?

* Name: PLATO

What scope do you want to create your data source in ?

Scope: Global

What JNDI name would you like to assign to your new JDBC Data Source?

JNDI Name: jdbc/PLATO

What database type would you like to select?

Database Type: Oracle

Back Next Finish Cancel

4. Select **Thin for Service Connections** (Instant) and click **Next**.

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.

* Indicates that the driver is explicitly supported by Oracle WebLogic Server.

Database Type: Oracle

What database driver would you like to use to create database connections? Note: * indicates that the driver is explicitly supported by Oracle WebLogic Server.

Database Driver: *Oracle's Driver (Thin) for Service connections; Versions:Any

Back Next Finish Cancel

5. Uncheck support for Global Transactions.

Home > Summary of Deployments > Summary of Servers > Summary of JDBC Data Sources

Create a New JDBC Data Source

Back Next Finish Cancel

Transaction Options

You have selected non-XA JDBC driver to create database connection in your new data source.

Does this data source support global transactions? If yes, please choose the transaction protocol for this data source.

☐ **Supports Global Transactions**

Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the *Logging Last Resource (LLR)* transaction option. Emulate Two-Phase Commit.

☐ **Logging Last Resource**

Select this option if you want to enable non-XA JDBC connections from the data source to emulate participation in global transactions using JTA. Select this option only if your application conditions.

☐ **Emulate Two-Phase Commit**

Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the one-phase commit transaction processing. With this option, the data source can participate in the global transaction.

☒ **One-Phase Commit**

Back Next Finish Cancel

6. Give database connection details and click **Next** to test connection.

Home > Summary of Deployments > Summary of Servers > Summary of JDBC Data Sources

Create a New JDBC Data Source

Back Next Finish Cancel

Connection Properties

Define Connection Properties.

What is the name of the database you would like to connect to?

Database Name:

What is the name or IP address of the database server?

Host Name:

What is the port on the database server used to connect to the database?

Port:

What database account user name do you want to use to create database connections?

Database User Name:

What is the database account password to use to create database connections?

Password:

Confirm Password:

Additional Connection Properties:

oracle.jdbc.DRCPConnectionClass:

Home > Summary of Deployments > Summary of Servers > Summary of JDBC Data Sources

Messages

✓ Connection test succeeded.

Create a New JDBC Data Source

Test Configuration | Back | Next | Finish | Cancel

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name:

7. Select targets to deploy data source.

Servers

☐ AdminServer

Clusters

☒ Api_Gateway_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Api_Gateway_Server

☒ Config_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Config_Server

☒ Discovery_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Discovery_Server

☒ Plato_Alerts_Management_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Plato_Alerts_Management_Server

☒ Plato_Batch_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Plato_Batch_Server

☒ Plato_Feed_Cluster
☐ All servers in the cluster
☒ Part of the cluster
☒ Plato_Feed_Server

8. View created Data Sources, verify JNDI Name and Targets.

Summary of JDBC Data Sources

Configuration | Monitoring

A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of JDBC connections. Applications can look up a data source on the JNDI tree and then borrow a database connection from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

[Customize this table](#)

Data Sources (Filtered - More Columns Exist)

New Delete Showing 1 to 1 of 1 Previous Next

<input type="checkbox"/>	Name	Type	JNDI Name	Targets
<input type="checkbox"/>	PLATO	Generic	jdbc/PLATO	Api_Gateway_Server, Config_Server, Discovery_Server, Plato_Alerts_Management_Server, Plato_Batch_Server, Plato_Feed_Server, ...

New Delete Showing 1 to 1 of 1 Previous Next

- Click **Activate Changes** after confirming details.

Change Center

View changes and restarts

Pending changes exist. They must be activated to take effect.

Activate Changes

Undo All Changes

Domain Structure

platoinfra_domain

Domain Partitions

Home Log Out Preferences Record Help

Home > Summary of Deployments > Summary of Servers > Summary of JDBC Data Sources

Summary of JDBC Data Sources

Configuration Monitoring

A JDBC data source is an object bound to the JNDI tree that provides database connectivity from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

Change Center

View changes and restarts

Click the **Lock & Edit** button to modify, add or delete items in this domain.

Lock & Edit

Release Configuration

Domain Structure

Coherence Clusters

Resource Groups

Resource Group Templates

Machines

Virtual Hosts

Virtual Targets

Work Managers

Concurrent Templates

Resource Management

Startup and Shutdown Classes

Deployments

Services

Messaging

Data Sources

How do I...

Create JDBC generic data sources

Create JDBC GridLink data sources

Home Log Out Preferences Record Help

Welcome, Connected to: platoinfra_domain

Home > Summary of Deployments > Summary of Servers > Summary of Clusters > Summary of Machines > Summary of JDBC Data Sources

Messages

All changes have been activated. No restarts are necessary.

Summary of JDBC Data Sources

Configuration Monitoring

A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of JDBC connections. Applications can look up a data source on the JNDI tree and then borrow a database connection from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

Customize this table

Data Sources (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

Name	Type	JNDI Name	Targets
PLATO	Generic	jdbc/PLATO	Api_Gateway_Server, Config_Server, Discovery_Server, Plato_Alerts_Management_Server, Plato_Batch_Server, Plato_Feed_Server, ...

Showing 1 to 1 of 1 Previous Next

- Post creating Data Source, click **Data source** under **Services** menu.
- Select the specific Data sources one by one.
- Navigate to **Connection Pool** tab under **Configuration** section and update the initial capacity, Minimum capacity and Max capacity as explained in the below screenshot

Configuration Targets Monitoring Control Security Notes

Connection Pool Oracle JDBC Transaction Diagnostics Identity Options

URL: jdbc:oracle:thin:@//uhf05g2:1521:O8TFPM15CPDB

Driver Class Name: oracle.jdbc.OracleDriver

Properties:

System Properties:

Encrypted Properties:

Password:

Confirm Password:

Initial Capacity: 5

Maximum Capacity: 12

Minimum Capacity: 1

Statement Cache Type: LRU

The URL of the database to connect to. The format of the URL varies by JDBC driver. [More info...](#)

The full package name of JDBC driver class used to create the physical database connections in the connection pool. (Note that this driver class must be in the classpath of any server to which it is deployed). [More info...](#)

The list of properties passed to the JDBC driver that are used to create physical database connections. For example: server=obsemer1. List each property-value pair on a separate line. [More info...](#)

The list of system properties passed to the JDBC driver that are used to create physical database connections. For example: server=obsemer1. List each property-value pair on a separate line. [More info...](#)

The list of encrypted properties passed to the JDBC driver that are used to create physical database connections. For example: password=obsemer1. [More info...](#)

The password attribute passed to the JDBC driver when creating physical database connections. [More info...](#)

The number of physical connections to create when creating the connection pool in the data source. If unable to create this number of connections, a subset of the data source will fail. [More info...](#)

The maximum number of physical connections that this connection pool can contain. [More info...](#)

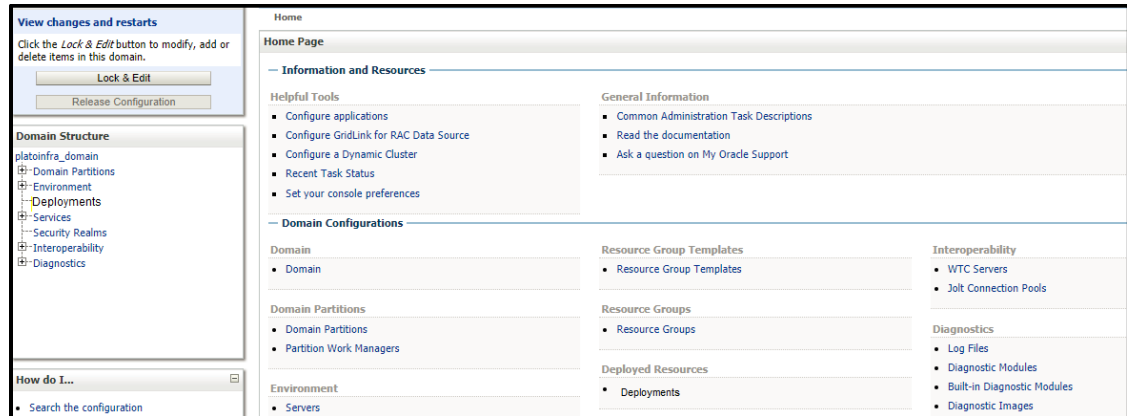
The minimum number of physical connections that this connection pool can contain after it is initialized. [More info...](#)

The algorithm used for maintaining the prepared statements stored in the statement cache. [More info...](#)

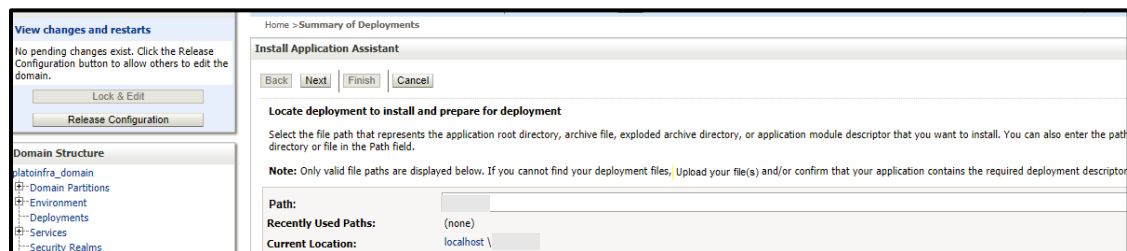
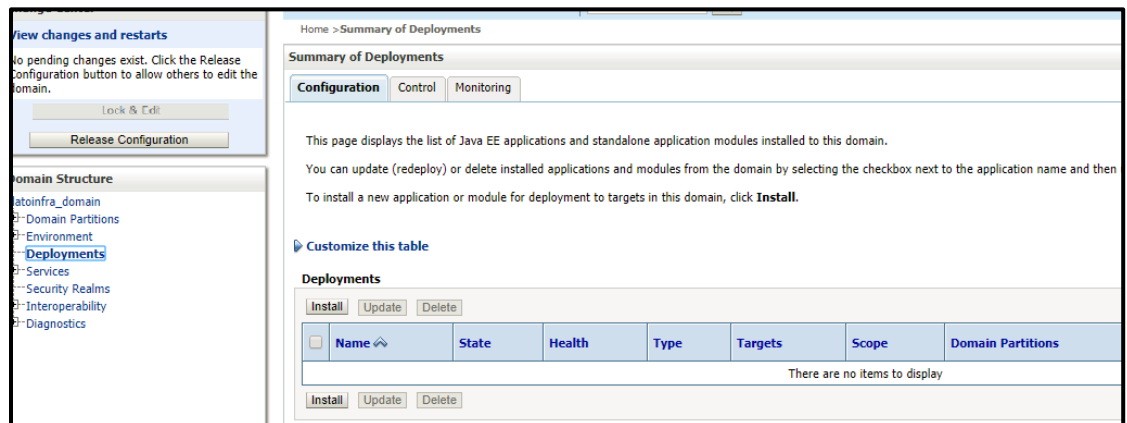
1.5 How to Deploy Application

Steps to Deploy archives as application on weblogic is same for all the above except for managed server and domain, where we deploy will differ. Perform the following steps to see how deployment of archive as application is done on weblogic:

1. Navigate to left menu and select **Deployments**.



2. Click on **Lock and Edit** and then click **Install**.



- Click **Upload your file(s)** to select archive, **Choose File** and click **Next**.

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit
Release Configuration

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Security Realms
 - Interoperability
 - Diagnostics

Install Application Assistant

Back Next Finish Cancel

Upload a deployment to the Administration Server

Click the Browse button below to select an application or module on the machine from which you are currently browsing. When you have located the file, click the Next button to upload this deployment to the Administration Server.

Deployment Archive: Choose File No file chosen

Upload a deployment plan (this step is optional)

A deployment plan is a configuration which can supplement the descriptors included in the deployment archive. A deployment will work without a deployment plan. This deployment plan archive will be a directory of configuration information packaged as a .jar file. See related links for additional information about deployment plans.

Deployment Plan Archive: Choose File No file chosen

Back Next Finish Cancel

- After archive is uploaded, click **Next**.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit
Release Configuration

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

Home Log Out Preferences Record Help

Welcome, Connected to: platoinfra_domain

Home > Summary of Deployments > Summary of Servers > Summary of Clusters > Summary of Machines > Summary of JDBC Data Sources > Summary of Deployments

Install Application Assistant

Back Next Finish Cancel

Upload a deployment to the Administration Server

Click the Browse button below to select an application or module on the machine from which you are currently browsing. When you have located the file, click the Next button to upload this deployment to the Administration Server.

Deployment Archive: Browse... plato-discovery-service-5.1.0.war

Upload a deployment plan (this step is optional)

A deployment plan is a configuration which can supplement the descriptors included in the deployment archive. A deployment will work without a deployment plan, but you can also upload a deployment plan archive now. This deployment plan archive will be a directory of configuration information packaged as a .jar file. See related links for additional information about deployment plans.

Deployment Plan: Browse... No file selected.

Back Next Finish Cancel

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit
Release Configuration

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

Home Log Out Preferences Record Help

Welcome, Connected to: platoinfra_domain

Home > Summary of Deployments > Summary of Servers > Summary of Clusters > Summary of Machines > Summary of JDBC Data Sources > Summary of Deployments

Messages

✓ The file plato-discovery-service-5.1.0.war has been uploaded successfully to C:\Users\ags\Documents\Oracle\Middleware\Oracle_Home\user_projects\domains\platoinfra_domain\servers\AdminServer\upload

Install Application Assistant

Back Next Finish Cancel

Locate deployment to install and prepare for deployment

Select the file path that represents the application root directory, archive file, exploded archive directory, or application module descriptor that you want to install. You can also enter the path of the application directory or file in the Path field.

Note: Only valid file paths are displayed below. If you cannot find your deployment files, Upload your file(s) and/or confirm that your application contains the required deployment descriptors.

Path:

Recently Used Paths: (none)

Current Location:

plato-discovery-service-5.1.0.war

Back Next Finish Cancel

- Select option **Install this deployment as an application** and click **Next**.

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- temp_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Security Realms
 - Interoperability
 - Diagnostics

How do I...

Install Application Assistant

Back Next Finish Cancel

Choose installation type and scope

Select if the deployment should be installed as an application or library. Also decide the scope of this deployment.

The application and its components will be targeted to the same locations. This is the most common usage.

☒ **Install this deployment as an application**

Application libraries are deployments that are available for other deployments to share. Libraries should be available on all of the targets running their referencing applications.

☐ **Install this deployment as a library**

Select a scope in which you want to install the deployment.

Scope: Global

Back Next Finish Cancel

6. Select target servers/clusters on which application has to be deployed and the **Next**.

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

How do I...

- Start and stop a deployed enterprise application
- Configure an enterprise application
- Create a deployment plan
- Target an enterprise application to a server instance
- Test the modules in an enterprise application

System Status

Health of Running Servers as of 6:34 PM

Failed (0)

Available targets for plato-discovery-service-5.1.0 :

Servers

- ☐ AdminServer

Clusters

- ☐ Apl_Gateway_Cluster
 - ☐ All servers in the cluster
 - ☐ Part of the cluster
 - ☐ Apl_Gateway_Server
- ☐ Config_Cluster
 - ☐ All servers in the cluster
 - ☐ Part of the cluster
 - ☐ Config_Server
- ☒ Discovery_Cluster
 - ☐ All servers in the cluster
 - ☒ Part of the cluster
 - ☒ Discovery_Server
- ☐ Plato_Alerts_Management_Cluster
 - ☐ All servers in the cluster
 - ☐ Part of the cluster
 - ☐ Plato_Alerts_Management_Server

7. Click **Finish** and then click **Save and Activate Changes**.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

How do I...

- Start and stop a deployed enterprise application
- Configure an enterprise application
- Create a deployment plan
- Target an enterprise application to a server instance
- Test the modules in an enterprise application

System Status

Health of Running Servers as of 6:34 PM

Failed (0)

Summary of Deployments

Configuration | **Control** | **Monitoring**

This page displays the list of Java EE applications and standalone application modules installed to this domain. You can update (redeploy) or delete installed applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page. To install a new application or module for deployment to targets in this domain, click **Install**.

Customize this table

Deployments

<input type="checkbox"/>	Name	State	Health	Type	Targets	Scope	Domain Partitions	Deployment Order
<input type="checkbox"/>	plato-discovery-service-5.1.0	distribute Initializing		Web Application	Discovery_Server	Global		100

Showing 1 to 1 of 1 Previous | Next

Change Center

View changes and restarts

Click the **Lock & Edit** button to modify, add or delete items in this domain.

Domain Structure

- Environment
 - Servers
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management
 - Startup and Shutdown Classes
- Deployments

How do I...

- Install an enterprise application
- Configure an enterprise application

Summary of Deployments

Configuration | Control | Monitoring

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can update (redeploy) or delete installed applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

To install a new application or module for deployment to targets in this domain, click **Install**.

Customize this table

Deployments

Install | Update | Delete

Name	State	Health	Type	Targets	Scope	Domain Partitions	Deployment Order
<input type="checkbox"/> plato-discovery-service-5.1.0	New		Web Application	Discovery_Server	Global		100

Showing 1 to 1 of 1 Previous | Next

- Click **Deployments** and then **Control** to changes the state of application from prepared to active status.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

Summary of Deployments

Configuration | **Control** | Monitoring

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can start and stop applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

Customize this table

Deployments

Start | Stop

Name	State	Health	Type	Targets	Scope	Domain Partitions
<input type="checkbox"/> plato-discovery-service-5.1.0	Prepared	OK	Web Application	Discovery_Server	Global	

Showing 1 to 1 of 1 Previous | Next

- Under **Deployment**, click **Start** dropdown and select **Start all requests**.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

Summary of Deployments

Configuration | **Control** | Monitoring

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can start and stop applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

Customize this table

Deployments

Start | Stop

Name	State	Health	Type	Targets	Scope	Domain Partitions
<input type="checkbox"/> plato-discovery-service-5.1.0	Prepared	OK	Web Application	Discovery_Server	Global	

Showing 1 to 1 of 1 Previous | Next

- Click **Yes**.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Domain Structure

- platoinfra_domain
 - Domain Partitions
 - Environment
 - Servers

Start Application Assistant

Start Deployments

You have selected the following deployments to be started. Click 'Yes' to continue, or 'No' to cancel.

- plato-discovery-service-5.1.0

Yes | **No**

11. The status is displayed as **Active** in the state column.

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit

Release Configuration

Domain Structure

- Environment
 - Servers
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management
 - Startup and Shutdown Classes
- Deployments**

Summary of Deployments

Home > Summary of Clusters > Summary of Machines > Summary of JDBC Data Sources > Summary of Deployments > Summary of Servers > Summary of Deployments > Summary of Servers > Discovery_Server > Summary of Servers > Summary of Deployments

Configuration Control Monitoring

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can update (redeploy) or delete installed applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

To install a new application or module for deployment to targets in this domain, click **Install**.

Customize this table

Deployments

Install Update Delete

Showing 1 to 1 of 1 Previous Next

<input type="checkbox"/>	Name	State	Health	Type	Targets	Scope	Domain Partitions	Deployment Order
<input type="checkbox"/>	plato-discovery-service-5.1.0	Active	OK	Web Application	Discovery_Server	Global		100

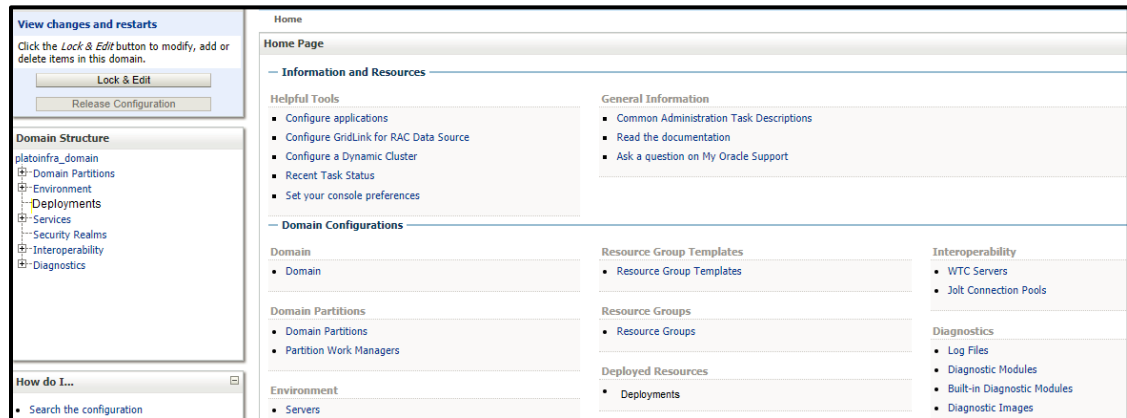
Install Update Delete

Showing 1 to 1 of 1 Previous Next

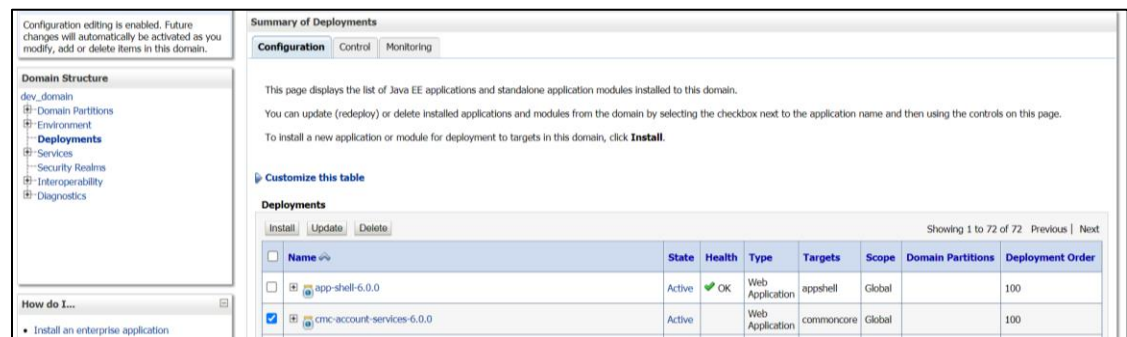
1.6 How to Undeploy Application

Login into weblogic server with the proper credentials.

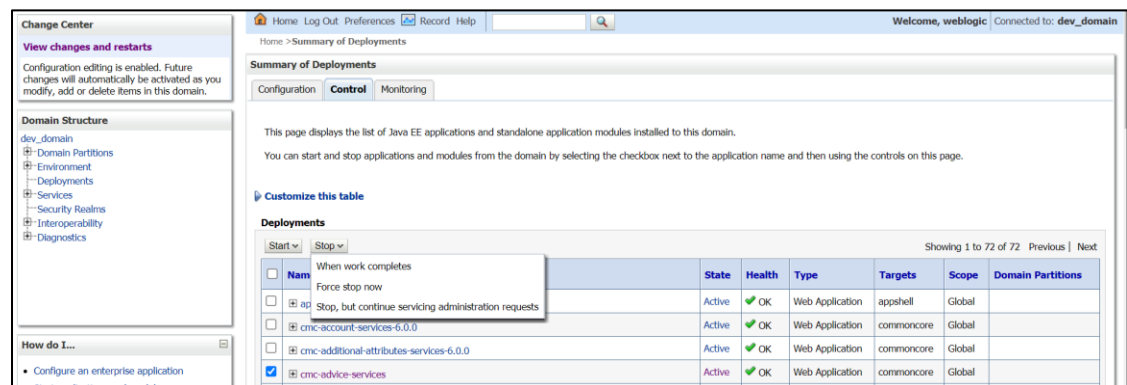
1. Navigate to left menu and select **Deployments**.



2. Click **Lock and Edit** and then select the service that needs to be undeployed in Deployments.



3. On **Control** tab, click **Stop**, and select **Force stop now** from the dropdown list.



- Once it changes to prepared state, click **Configuration** tab.

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- dev_domain
 - Domain Partitions
 - Environment
 - Deployments**
 - Services
 - Security Realms
 - Interoperability
 - Diagnostics

How do I...

- Install an enterprise application
- Configure an enterprise application
- Update (redeploy) an enterprise application
- Monitor the modules of an enterprise application

Summary of Deployments

Configuration | Control | Monitoring

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can update (redeploy) or delete installed applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

To install a new application or module for deployment to targets in this domain, click **Install**.

Customize this table

Deployments

Install | Update | Delete

Showing 1 to 72 of 72 Previous | Next

<input type="checkbox"/>	Name	State	Health	Type	Targets	Scope	Domain Partitions	Deployment Order
<input type="checkbox"/>	app-shell-6.0.0	Active	OK	Web Application	appshell	Global		100
<input type="checkbox"/>	omc-account-services-6.0.0	Active		Web Application	commoncore	Global		100
<input type="checkbox"/>	omc-additional-attributes-services-6.0.0	Active		Web Application	commoncore	Global		100
<input checked="" type="checkbox"/>	omc-advice-services	Prepared		Web Application	commoncore	Global		100

- Select the service again and click on Delete to undeploy the service.

1.7 How to Restart Servers

Perform the following steps to restart servers:

- Navigate to left menu and select **Environment**, and then click **Servers**.

Change Center

View changes and restarts

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- temp_domain
 - Domain Partitions
 - Environment
 - Servers**
 - Clusters
 - Coherence Clusters
 - Resource Groups
 - Resource Group Templates
 - Machines
 - Virtual Hosts
 - Virtual Targets
 - Work Managers
 - Concurrent Templates
 - Resource Management

2. Click **Control** tab.

The screenshot shows the 'Summary of Servers' page in the Oracle WebLogic Control Center. The 'Control' tab is selected. The page displays a table of servers with columns: Server, Machine, State, and Status of Last Action. The 'Discovery_Server' is highlighted.

Server	Machine	State	Status of Last Action
AdminServer(admin)	platoinfra_Machine	RUNNING	None
Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
Config_Server	platoinfra_Machine	SHUTDOWN	None
Discovery_Server	platoinfra_Machine	RUNNING	None
Plato_Alerts_Management_Server	platoinfra_Machine	SHUTDOWN	None
Plato_Batch_Server	platoinfra_Machine	SHUTDOWN	None
Plato_Feed_Server	platoinfra_Machine	SHUTDOWN	None
Plato_Orch_Server	platoinfra_Machine	SHUTDOWN	None
Plato_UI_Config_Server	platoinfra_Machine	SHUTDOWN	None

3. Select servers to **Shutdown**

The screenshot shows the 'Summary of Servers' page in the Oracle WebLogic Control Center. The 'Control' tab is selected. The 'Shutdown' button is highlighted in the table's header row. A tooltip is visible over the 'Shutdown' button, indicating the action.

Server	Machine	State	Status of Last Action
AdminServer(admin)	platoinfra_Machine	RUNNING	None
Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
Config_Server	platoinfra_Machine	SHUTDOWN	None
Discovery_Server	platoinfra_Machine	RUNNING	None

4. Click **Yes** to confirm shutdown.

The screenshot shows the 'Server Life Cycle Assistant' dialog box. It prompts the user to confirm the shutdown of the selected servers. The 'Discovery_Server' is listed as the server to be shut down. The 'Yes' button is highlighted.

Forcibly Shutdown Servers

You have selected the following servers to be immediately shut down. Press 'Yes' to continue or 'No' to cancel.

- Discovery_Server

5. The status displayed as shown below:

The screenshot shows the 'Summary of Servers' page in the Oracle WebLogic Control Center. The 'Control' tab is selected. The 'Discovery_Server' is now in the 'FORCE_SHUTTING_DOWN' state, and its status of last action is 'TASK IN PROGRESS'.

Server	Machine	State	Status of Last Action
AdminServer(admin)	platoinfra_Machine	RUNNING	None
Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
Config_Server	platoinfra_Machine	SHUTDOWN	None
Discovery_Server	platoinfra_Machine	FORCE_SHUTTING_DOWN	TASK IN PROGRESS

- Once shutdown is completed, navigate to **Control**, select the servers to **Start**, and click **Yes** to confirm action.

Change Center
View changes and restarts
No pending changes exist. Click the Release Configuration button to allow others to edit the domain.
Lock & Edit
Release Configuration

Domain Structure
platoinfra_domain
Domain Partitions
Environment
Servers
Clusters
Coherence Clusters
Resource Groups
Resource Group Templates
Machines
Virtual Hosts
Virtual Targets
Work Managers
Concurrent Templates
Resource Management

Summary of Servers
Configuration **Control**

Use this page to change the state of the servers in this WebLogic Server domain. Control operations on Managed Servers require starting the Node Manager. Starting Managed Servers in Standby mode requires the domain-wide administration port.

Customize this table

Servers (Filtered - More Columns Exist)
Start Resume Suspend Shutdown Restart SSL Showing 1 to 9 of 9 Previous Next

Server	Machine	State	Status of Last Action
<input type="checkbox"/> AdminServer(admin)	platoinfra_Machine	RUNNING	None
<input type="checkbox"/> Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
<input type="checkbox"/> Config_Server	platoinfra_Machine	SHUTDOWN	None
<input checked="" type="checkbox"/> Discovery_Server	platoinfra_Machine	SHUTDOWN	TASK COMPLETED

Change Center
View changes and restarts
No pending changes exist. Click the Release Configuration button to allow others to edit the domain.
Lock & Edit
Release Configuration

Domain Structure
platoinfra_domain
Domain Partitions
Environment
Servers

Server Life Cycle Assistant
Yes No
Start Servers
You have selected the following servers to be started. Press 'Yes' to continue or 'No' to cancel.
Discovery_Server
Yes No

Change Center
View changes and restarts
No pending changes exist. Click the Release Configuration button to allow others to edit the domain.
Lock & Edit
Release Configuration

Domain Structure
platoinfra_domain
Domain Partitions
Environment
Servers
Clusters
Coherence Clusters
Resource Groups
Resource Group Templates
Machines
Virtual Hosts
Virtual Targets
Work Managers
Concurrent Templates
Resource Management

Summary of Servers
Configuration **Control**

Use this page to change the state of the servers in this WebLogic Server domain. Control operations on Managed Servers require starting the Node Manager. Starting Managed Servers in Standby mode requires the domain-wide administration port.

Customize this table

Servers (Filtered - More Columns Exist)
Start Resume Suspend Shutdown Restart SSL Showing 1 to 9 of 9 Previous Next

Server	Machine	State	Status of Last Action
<input type="checkbox"/> AdminServer(admin)	platoinfra_Machine	RUNNING	None
<input type="checkbox"/> Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
<input type="checkbox"/> Config_Server	platoinfra_Machine	SHUTDOWN	None
<input type="checkbox"/> Discovery_Server	platoinfra_Machine	SHUTDOWN	TASK IN PROGRESS

- When all requested servers are running, navigate to **Summary of Deployments**, and check if deployments are in active state.

Change Center
View changes and restarts
No pending changes exist. Click the Release Configuration button to allow others to edit the domain.
Lock & Edit
Release Configuration

Domain Structure
platoinfra_domain
Domain Partitions
Environment
Servers
Clusters
Coherence Clusters
Resource Groups
Resource Group Templates
Machines
Virtual Hosts
Virtual Targets
Work Managers
Concurrent Templates
Resource Management

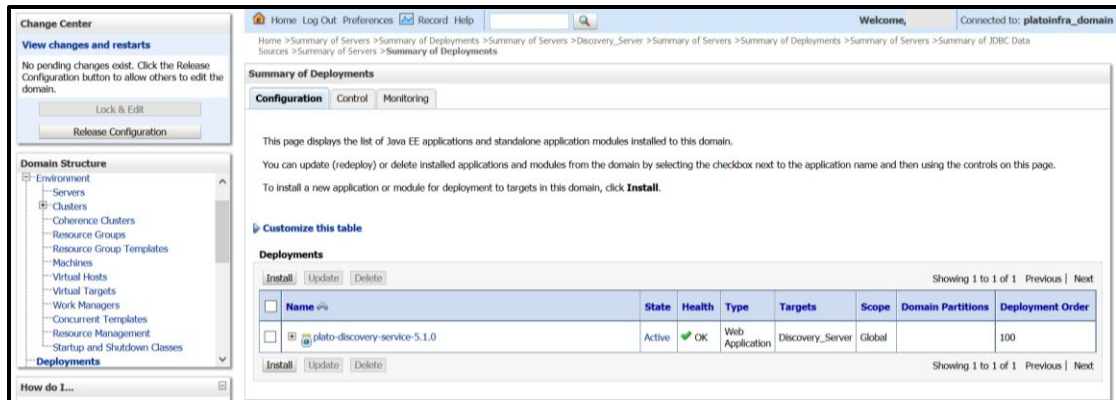
Summary of Servers
Configuration **Control**

Use this page to change the state of the servers in this WebLogic Server domain. Control operations on Managed Servers require starting the Node Manager. Starting Managed Servers in Standby mode requires the domain-wide administration port.

Customize this table

Servers (Filtered - More Columns Exist)
Start Resume Suspend Shutdown Restart SSL Showing 1 to 9 of 9 Previous Next

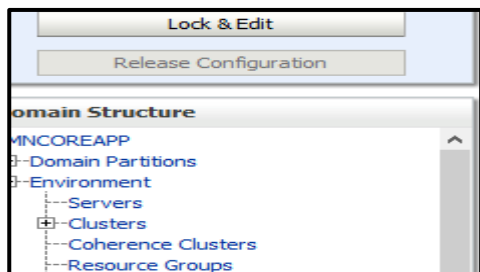
Server	Machine	State	Status of Last Action
<input type="checkbox"/> AdminServer(admin)	platoinfra_Machine	RUNNING	None
<input type="checkbox"/> Apl_Gateway_Server	platoinfra_Machine	SHUTDOWN	None
<input type="checkbox"/> Config_Server	platoinfra_Machine	SHUTDOWN	None
<input type="checkbox"/> Discovery_Server	platoinfra_Machine	RUNNING	TASK COMPLETED



1.8 How to Check Port Number

Perform the following steps to check port numbers:

1. Specify the **User id** and **Password**, and login to **WebLogic console**.
2. Click **Environment** and then click **Server**.



3. Under Servers (Filtered - More Columns Exist) section, you will be able to see all the server listed.

	Name	Type	Cluster	Machine	State	Health	Listen Port
<input type="checkbox"/>	AdminServer(admin)	Configured			RUNNING	OK	7020
<input type="checkbox"/>	managed_server1	Configured		Machine1	RUNNING	OK	7023

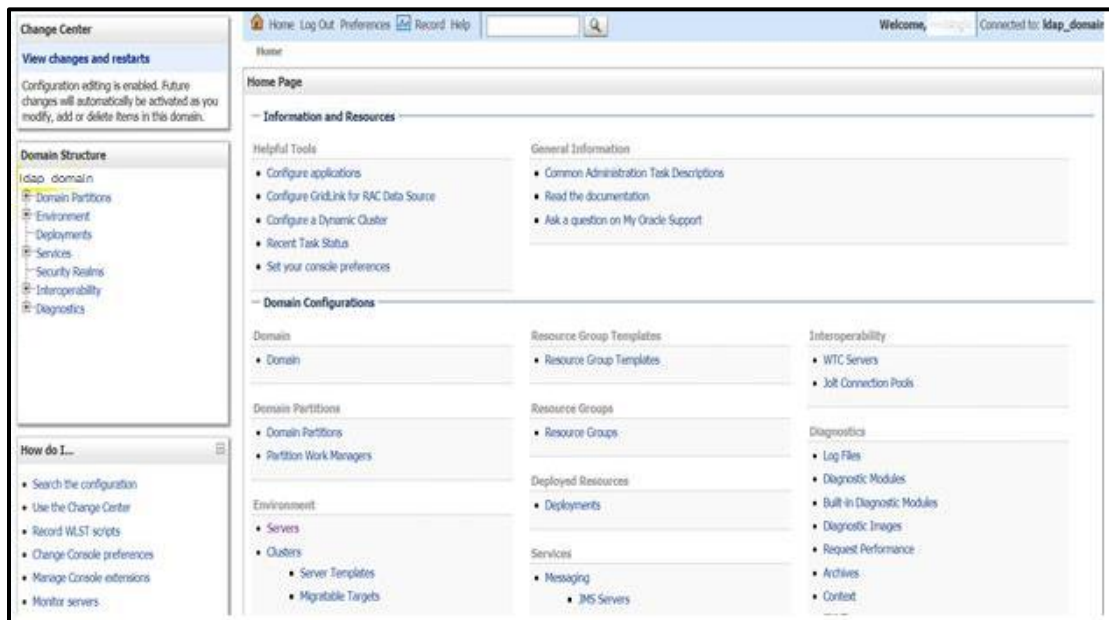
1.9 Weblogic Embedded LDAP Setup

The following changes are to be made for configuring the Weblogic-Embedded LDAP server for Oracle Banking Microservices Architecture:

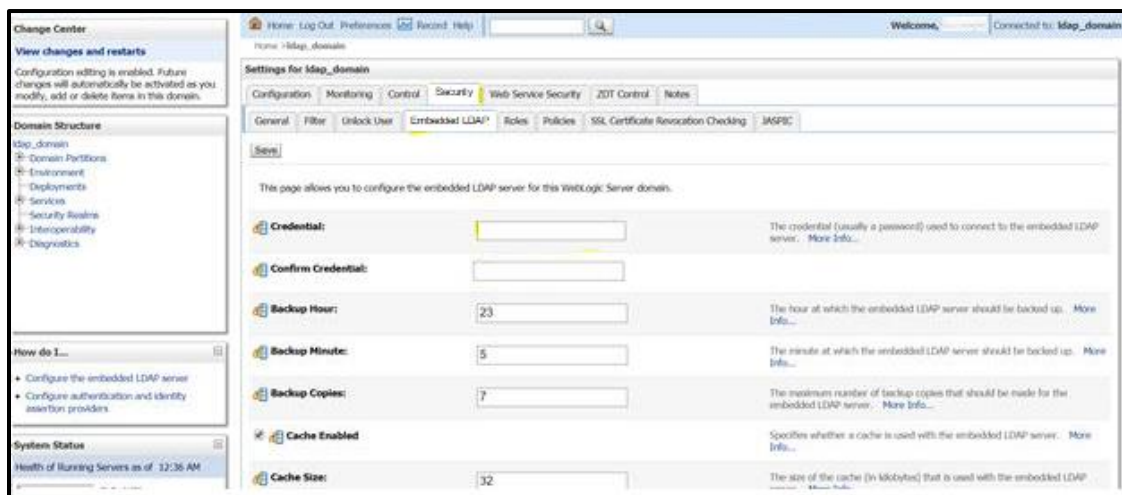
- Configuration of Weblogic LDAP
- Creation of Users
- Plato Security Config Table Entries

1.9.1 Configuration of Weblogic LDAP

1. Open the **Administration Console** for the **Weblogic**, and click domain name in left panel.



2. Under Settings for ldap_domain, click **Security** tab, and then click **Embedded LDAP** tab.



- Set the **Credential** for Weblogic Embedded LDAP store. This is needed in the **Security Config** table.

Change Center

View changes and restarts

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- ldap_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Security Realms
 - Interoperability
 - Diagnostics

How do I...

- Configure the embedded LDAP server
- Configure authentication and identity assertion providers

System Status

Health of Running Servers as of 12:38 AM

Home Log Out Preferences Record Help

Home > ldap_domain

Settings for ldap_domain

Configuration Monitoring Control **Security** Web Service Security ZOT Control Notes

General Filter Unlock User **Embedded LDAP** Roles Policies SSL Certificate Revocation Checking JASPER

Save

This page allows you to configure the embedded LDAP server for this WebLogic Server domain.

Credential: The credential (usually a password) used to connect to the embedded LDAP server. [More Info...](#)

Confirm Credential:

Backup Hour: The hour at which the embedded LDAP server should be backed up. [More Info...](#)

Backup Minute: The minute at which the embedded LDAP server should be backed up. [More Info...](#)

Backup Copies: The minimum number of backup copies that should be made for the embedded LDAP server. [More Info...](#)

☒ **Cache Enabled** Specifies whether a cache is used with the embedded LDAP server. [More Info...](#)

Cache Size: The size of the cache (in kilobytes) that is used with the embedded LDAP server. [More Info...](#)

1.9.2 Creation of Users

- Navigate to left menu, and click **Security Realms**.
- In the **Summary of Security Realms** window, click **myrealm**.

Change Center

View changes and restarts

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- temp_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Security Realms**
 - Interoperability
 - Diagnostics

How do I...

- Manage users and groups
- Create groups
- Modify groups
- Delete groups

System Status

Home Log Out Preferences Record Help

Home > temp_domain > Summary of Security Realms

Summary of Security Realms

A security realm is a container for the mechanisms—including users, groups, security roles, security policies, and security providers—that are in a WebLogic Server domain, but only one can be set as the default security realm, which is reserved for domain administrative purposes.

This Security Realms page lists each security realm that has been configured in this WebLogic Server domain. Click the name of the realm.

[Customize this table](#)

Realms (Filtered - More Columns Exist)

[New](#) [Delete](#)

<input type="checkbox"/> Name	Default Realm
<input type="checkbox"/> myrealm	true

[New](#) [Delete](#)

- Under **Settings for myrealm**, click **Users and Groups**.
- Click **Groups** tab. Click **New** to make a new group.

Change Center

View changes and restarts

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- temp_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Security Realms
 - Interoperability
 - Diagnostics

How do I...

- Manage users and groups
- Create groups
- Modify groups
- Delete groups

System Status

Home Log Out Preferences Record Help

Home > temp_domain > Summary of Security Realms > myrealm > Users and Groups

Settings for myrealm

Configuration **Users and Groups** Roles and Policies Credential Mappings Providers Migration

[Users](#) **[Groups](#)**

This page displays information about each group that has been configured in this security realm.

[Customize this table](#)

Groups

[New](#) [Delete](#)

<input type="checkbox"/> Name	Description
<input type="checkbox"/> AdminChannelUsers	AdminChannelUsers can access the admin channel.
<input type="checkbox"/> Administrators	Administrators can view and modify all resource attributes and start and stop servers.
<input type="checkbox"/> AppTesters	AppTesters group.
<input type="checkbox"/> CrossDomainConnectors	CrossDomainConnectors can make inter-domain calls from foreign domains.
<input type="checkbox"/> Deployers	Deployers can view all resource attributes and deploy applications.
<input type="checkbox"/> Monitors	Monitors can view and modify all resource attributes and perform operations not restricted by roles.
<input type="checkbox"/> Operators	Operators can view and modify all resource attributes and perform server lifecycle operations.
<input type="checkbox"/> OracleSystemGroup	Oracle application software system group.

[New](#) [Delete](#)

5. Add the relevant details and click **OK**. The new group will be created.

The screenshot shows the 'Create a New Group' dialog box. On the left, there is a 'Domain Structure' tree with 'temp_domain' selected. Below it, a 'How do I...' section lists 'Create groups', 'Modify groups', and 'Delete groups'. The main area of the dialog is titled 'Create a New Group' and contains the following fields:

- Group Properties:** A section with a note: 'The following properties will be used to identify your new Group. * Indicates required fields'.
- Name:** A text field containing 'Developers'.
- Description:** A text field containing 'Group for Developers'.
- Provider:** A dropdown menu showing 'DefaultAuthenticator'.

At the bottom of the dialog are 'OK' and 'Cancel' buttons.

6. Click **Users** tab, and click **New** to create user.

The screenshot shows the 'Users and Groups' tab in the Oracle WebLogic Server console. The left sidebar is the same as in the previous screenshot. The main area is titled 'Settings for myrealm' and has tabs for 'Configuration', 'Users and Groups', 'Roles and Policies', 'Credential Mappings', 'Providers', and 'Migration'. The 'Users and Groups' tab is active, and the 'Users' sub-tab is selected. Below the tabs, there is a table of users:

Name	Description
LCMUser	This is the default service account for WebLogic Server Lifecycle Manager configuration updates.
OracleSystemUser	Oracle application software system user.
weblogic	This user is the default administrator.

At the bottom of the table are 'New' and 'Delete' buttons.

7. Enter the required details for the user. After completing click **OK**. The user will be created.

The screenshot shows the 'Create a New User' dialog box. On the left, there is a 'Domain Structure' tree with 'temp_domain' selected. Below it, a 'How do I...' section lists 'Create users', 'Modify users', 'Delete users', 'Create groups', and 'Manage users and groups'. The main area of the dialog is titled 'Create a New User' and contains the following fields:

- User Properties:** A section with a note: 'The following properties will be used to identify your new User. * Indicates required fields'.
- Name:** A text field containing 'testuser'.
- Description:** A text field containing 'user for testing'.
- Provider:** A dropdown menu showing 'DefaultAuthenticator'.
- Password:** A text field with masked characters (dots).
- Confirm Password:** A text field with masked characters (dots).

At the bottom of the dialog are 'OK' and 'Cancel' buttons.

8. Click on the newly created user to assign the newly created user to some group.

View changes and restarts
Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure
temp_domain
├─ Domain Partitions
├─ Environment
├─ Deployments
├─ Services
├─ Security Realms
├─ Interoperability
└─ Diagnostics

How do I...
• Manage users and groups
• Create users
• Modify users
• Delete users

Home > temp_domain > Summary of Security Realms > myrealm > Users and Groups

Messages
✔ User created successfully

Settings for myrealm
Configuration Users and Groups Roles and Policies Credential Mappings Providers Migration

Users Groups

This page displays information about each user that has been configured in this security realm.

Customize this table

Users (Filtered - More Columns Exist)

New Delete

<input type="checkbox"/>	Name	Description
<input type="checkbox"/>	ADMINUSER1	ADMINUSER1
<input type="checkbox"/>	LCMUser	This is the default service account for WebLogic Server Lifecycle Manager configuration updates.
<input type="checkbox"/>	OracleSystemUser	Oracle application software system user.
<input type="checkbox"/>	weblogic	This user is the default administrator.

New Delete

9. Under **Setting for ADMINUSER1** (or whatever your user's name is) window, click **Groups** tab.

View changes and restarts
Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure
temp_domain
├─ Domain Partitions
├─ Environment
├─ Deployments
├─ Services
├─ Security Realms
├─ Interoperability
└─ Diagnostics

How do I...
• Create users

Home > temp_domain > Summary of Security Realms > myrealm > Users and Groups > ADMINUSER1

Settings for ADMINUSER1
General Passwords Attributes Groups

Save

Use this page to configure group membership for this user.

Parent Groups:

Available:

- ☐ AdminChannelUsers
- ☐ Administrators
- ☐ AppTesters
- ☐ CrossDomainConnectors
- ☐ Deployers
- ☐ Monitors
- ☐ Operators
- ☐ OracleSystemGroup

Chosen:

Save

10. Select the groups you want to assign to the user and click single right button as shown below.

View changes and restarts
Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure
temp_domain
├─ Domain Partitions
├─ Environment
├─ Deployments
├─ Services
├─ Security Realms
├─ Interoperability
└─ Diagnostics

Home > temp_domain > Summary of Security Realms > myrealm > Users and Groups > ADMINUSER1

Settings for ADMINUSER1
General Passwords Attributes Groups

Save

Use this page to configure group membership for this user.

Parent Groups:

Available:

- ☐ AdminChannelUsers
- ☐ Administrators
- ☐ AppTesters
- ☐ CrossDomainConnectors
- ☒ Deployers
- ☐ Monitors
- ☒ Operators
- ☐ OracleSystemGroup

Chosen:

11. Click **Save**.

The screenshot shows the 'Settings for ADMINUSER1' page in the Oracle WebLogic console. The 'Groups' tab is selected, showing a list of 'Available' groups and a 'Chosen' list. The 'Available' list includes AdminChannelUsers, Administrators, AppTesters, CrossDomainConnectors, Monitors, and OracleSystemGroup. The 'Chosen' list includes Deployers and Operators. The 'Save' button is visible at the bottom of the page.

1.9.3 Oracle Banking Microservices Architecture Security Config Table Entries

Connection details for the embedded LDAP of weblogic (assuming the admin server is running on 10.99.99.10:7001) are given below:

Connection Details:

URL: ldap:// 10.99.99.10:7001

Server Base: dc={DOMAIN_NAME} (in our case it would be dc=ldap_domain)

User Search Base: ou=people,ou=myrealm

Server User: cn=admin

Server Credentials: As setup in step Point 3 under 1.8.1

Security Config Table Entries:

ID	VALUE	Description
LDAP_URL	ldap:// 10.99.99.10:7001	Valid LDAP Server address with port.
LDAP_SERVER_USER	cn=admin	LDAP server login username
LDAP_SERVER_BASE	dc=ldap_domain	LDAP Server Base
LDAP_SERVER_CREDENTIAL	ylksiMFfjVbfcpA7Qheh8Q==	LDAP server credentials in encrypted form(For Encryption steps, refer to Encrypted Utility section below)
LDAP_USER_SEARCH_BASE	ou=people,ou=myrealm	LDAP User Search Base
LDAP_PROVIDER	EMBEDDED_WEBLOGIC	Which LDAP Provider to be used. Also, if this row is not present in this table, then In-House Spring Plato LDAP will be used.

1.10 Oracle Analytic Server Setup

This section contains the following sub-sections:

- Prerequisite
- Start BI Server
- Upload BI Reports
- Test BI Reports

1.10.1 Prerequisite

Perform the following steps:

- Machine should have Java JDK1.8.0_271 has installed
- Oracle Analytics Server 5.5.0

1.10.2 Start BI Server

Perform the following steps to start BI server:

1. Start the weblogic server and analytics server.
2. Check the weblogic console whether analytics server is running.

1.10.3 Upload BI Reports

Perform the following steps to upload BI reports:

1. Login to the Analytics server console.
2. Open the OSDC and check for the report Catalog object **\{unzip folder\}\REP\{reportfilename}.xdrz** or any other Catalog objects listed below:

Catalog Object	Extensions Supported
Data Model	.xdmz
Folder	.xdrz
Report	.xdoz
Style Template	.xssz
Subtemplate	.xsbz

3. Upload the catalog object to Analytics Server.

1.10.4 Test BI Reports

Perform the following steps to generate BI reports:

1. Open the application, and go the **Reports** section of the application.
2. Choose the report generation criteria. For example, **Start Date** or **End Date**.
3. Choose the format of the report.
4. Generate the report.

NOTE: If the format of the report selected is PDF, a PDF report will be generated.

1.11 How to deploy Plato-Apigateway Router

1.11.1 Router deployment steps

Follow services must be deployment in below order to setup router service:

1. **Deploy plato-config-service**

- a. Set placeholder -Dflyway.domain.placeholders.plato-apigateway-router.server.port=<new server port for plato-apigateway-router>

2. **Deploy plato-ui-config-service**

- a. set -Dflyway.domain.placeholders.apigateway.port=<new server port for plato-apigateway-router>
- b. set -Dflyway.domain.placeholders.apigateway.host=<server host for plato-apigateway-router>

3. **Deploy plato-api-gateway**

- a. Migrate existing OAuth users:

API for migration - /api-gateway/migrateOAuthUsers

Example: http://hostname:8080/api-gateway/migrateOAuthUsers

Authorization - jwtToken

Headers:

applId,userId,entityId

Body (Json): ["client1", "client2"] - Migrate selected list of clients

or

Body (Json): ["ALL"] -Migrate all clients.

4. **Deploy plato-apigateway-router**

```
java -jar plato-apigateway-router.jar --plato.services.config.uri=http://hostname:8001 --plato.service.logging.path=/logfilePath
```

--plato.services.config.uri - Config server URI which is referred by all other services.

--plato.service.logging.path - Path where log file(plato-apigateway-router.log) must be created. Specify the same path as that of other services.

We can enable SSL for plato-apigateway-router by providing:

--server.ssl.enabled=true

--server.ssl.key-store=C:/Users/KEYS/keytool/keystore.jks

--key-store-password=xxxx

--server.ssl.trust-store=C:/Users/KEYS/keytool/truststore.jks

--trust-store-password=xxxxx

--salt=xxxxx

Note: Passwords and salt must be encrypted value generated using respective toolkits.

Provide ssl certs of plato-api-gateway required for validation call when plato-api-gateway is deployed in different server.:

--apigateway.useServerSSLKeys=false

--apigateway.ssl.key-store=C:/Users/KEYS/keytool/keystore.jks

--apigateway.ssl.key-store-password=xxxx

--apigateway.ssl.trust-store=C:/Users/KEYS/keytool/truststore.jks

--apigateway.ssl.trust-store-password=xxxxx

Note: Above certificates can be different than that of plato-apigateway-route

we must also provide trust certificates as

--spring.cloud.gateway.httpclient.ssl.trusted-x509-certificates=C:/Users/KEYS/keytool/keystore1.pem, C:/Users/KEYS/keytool/keystore2.pem

Note: Run this service with nohup command to that process will run on background

App-shell must point to plato-apigateway-router service. Update 'apigateway.url' by correcting it to "http://hostname:8080" - here 8080 is the port is configured for plato-apigateway-router.

1.11.2 Generation pem file and encryption of secrets:

Use plato-security-toolkit to encrypt secrets ---key-store-password, --trust-store-password, --apigateway.ssl.key-store-password, --apigateway.ssl.trust-store-password and these encrypted values must be passed to router service.

Encryption of secrets:

To encrypt the passwords as per Oracle Standards, we recommend toolkit - plato-security-toolkit

Usage: java -jar plato-security-toolkit-9.1.0.jar

Enter pass phrase: Test123

Enter Salt: 0.9412345671234567

Encrypted Password: m4Q1rbtegkWse2s7D2jKfw==

Encryption of salt: to encrypt --salt value used while generating encrypted secret. This encrypt salt must be passed to router service.

To encrypt the salt as per Oracle Standards, we recommend toolkit - plato-security-salt-encryption-toolkit

Usage: java -jar plato-security-salt-encryption-toolkit-9.1.0.jar

Enter Salt: 0.9412345671234567

Encrypted Password:

VmtjMWQxTnJOVlpPV0VaWFZrVndUMWxYTVU1bFJsSlpZMFZLYTFaVVZrWldWbWgzVkRGS
1JsWnFVVDA9

PEM file from keystore

keytool -exportcert -alias localhost -keystore keystore.jks -rfc -file keystore.pem

1.11.3 Timeout parameters

These parameters are similar to earlier ribbon timeout params

spring.cloud.gateway.httpclient.connect-timeout= 3000 //seconds

spring.cloud.gateway.httpclient.response-timeout= 360s

spring.cloud.gateway.httpclient.pool.acquire-timeout=6000 //milliseconds

spring.cloud.gateway.httpclient.pool.max-connections=10000

#Properties used webclient call is made to plato-api-gateway for validation

webclient.http.max.connections=1000

webclient.http.acquire.timeout.millisec=5000

webclient.http.connection.timeout.millisec=20000

webclient.http.read.timeout.seconds=20000

webclient.http.write.timeout.seconds=20000



Configuration and Deployment Guide

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax: +91 22 6718 3001

<https://www.oracle.com/industries/financial-services/index.html>

Copyright © 2018, 2023, Oracle and/or its affiliates. All rights reserved.

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

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 failsafe, 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.

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.

This software or hardware and documentation may provide access to or information on 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. 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.