

ANNEXURE – 1
Oracle Banking
Release 14.5.3.0.0
Part Number F50590-01
November 2021



Table of Contents

1. ANNEXURE - 1	1-1
1.1 INTRODUCTION	1-1
1.2 PLACEHOLDER UPDATE FOR PLATO-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	1-5
1.2.3 Method 3 – Using env files and setUserOverrides.sh file	1-7
1.2.4 Method 4 – Workflow Configuration	1-14
1.3 HOW TO CREATE DOMAIN AND CLUSTER CONFIGURATION	1-17
1.3.1 Domain Creation Configuration	1-18
1.3.2 Post Domain Creation Configurations	1-27
1.4 HOW TO CREATE DATASOURCE	1-28
1.5 HOW TO DEPLOY APPLICATION	1-34
1.6 HOW TO UNDEPLOY APPLICATION:-	1-38
1.7 HOW TO RESTART SERVERS	1-39
1.8 HOW TO CHECK PORT NUMBER	1-42
1.9 WEBLOGIC EMBEDDED LDAP SETUP	1-43
1.9.1 Configuration of Weblogic LDAP	1-43
1.9.2 Creation of Users	1-44
1.9.3 Plato Security Config Table Entries	1-47
1.10 ORACLE ANALYTIC SERVER SETUP	1-48
1.10.1 Prerequisite	1-48
1.10.2 Start BI Server	1-48
1.10.3 Upload BI Reports	1-49
1.10.4 Test BI Reports	1-49

1.1 Introduction

This guide is a supporting document for the installation of PLATO applications. You can find the reference in the respective installation guides.

1.2 Placeholder Update for Plato-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 the plato 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>  
-Deureka.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>
```

//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_HOSTNAME>
-Dflyway.domain.placeholders.apigateway.port= <APIGATEWAY_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 >
//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 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=<EVETNHUB_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=<EVETNHUB_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>

//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

<input type="checkbox"/>	Name	Type	Cluster	Machine	State	Health	Listen Port
<input type="checkbox"/>	AdminServer(admin)	Configured		whf00dtx	RUNNING	OK	7001
<input type="checkbox"/>	managed1_server	Configured		whf00dtx	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: 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...](#)

Arguments: The arguments to use when starting this server. [More Info...](#)

```
-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
```

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
    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  
  
### 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-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/PLATOFEED
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-widiget-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=
```

```
### 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": {

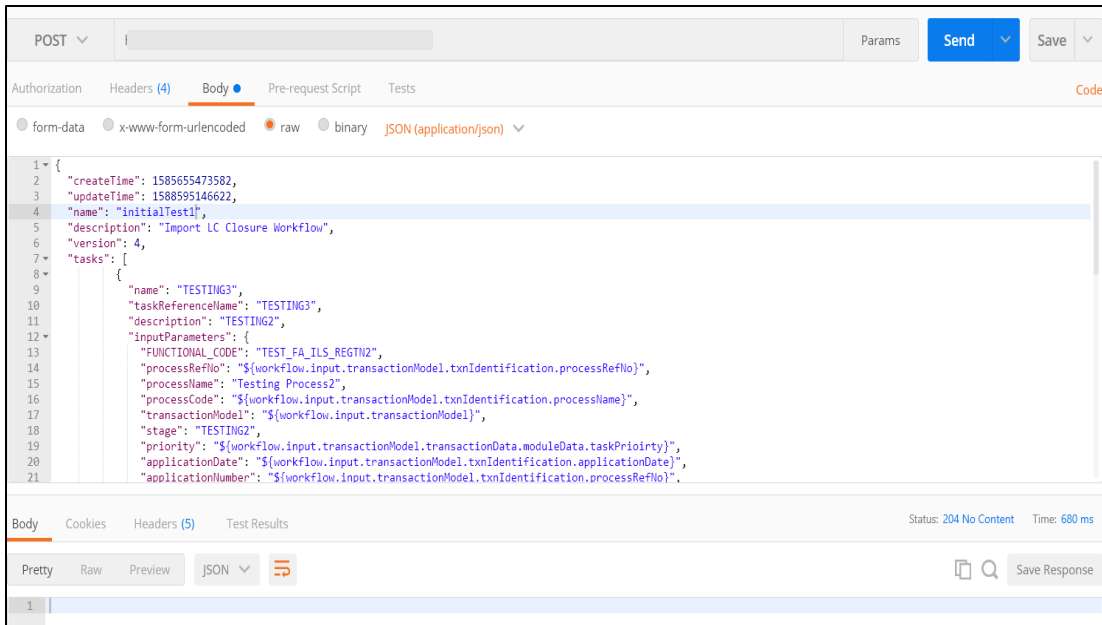
```

```

"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",

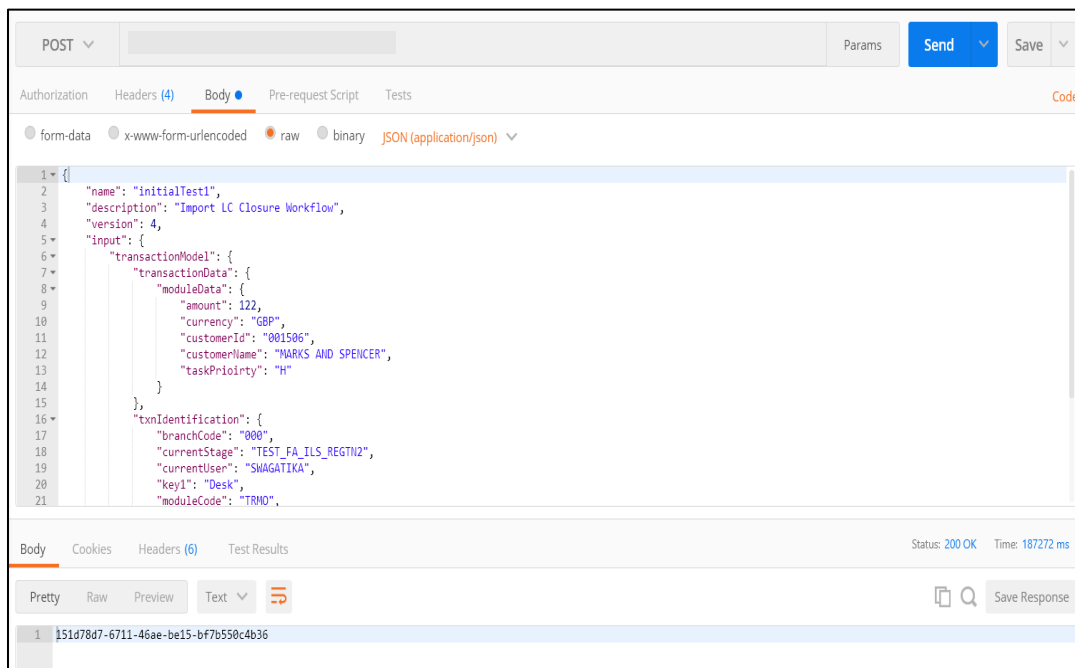
```

```

        "taskPriority": "H"
    }
},
    "txnIdentification": {
        "branchCode": "000",
        "currentStage": "TEST_FA_ILS_REGTN2",
        "currentUser": "SWAGATIKA",
        "key1": "Desk",
        "moduleCode": "TRMO",
        "processName": "Testing Process2",
        "processRefNo": "300ILC1012260",
        "applicationDate": 1588582461960,
        "taskOutcome": "PROCEED",
        "taskPriority": "H"
    }
}
}
}
}
}

```

The following screen depicts the sample workflow:



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

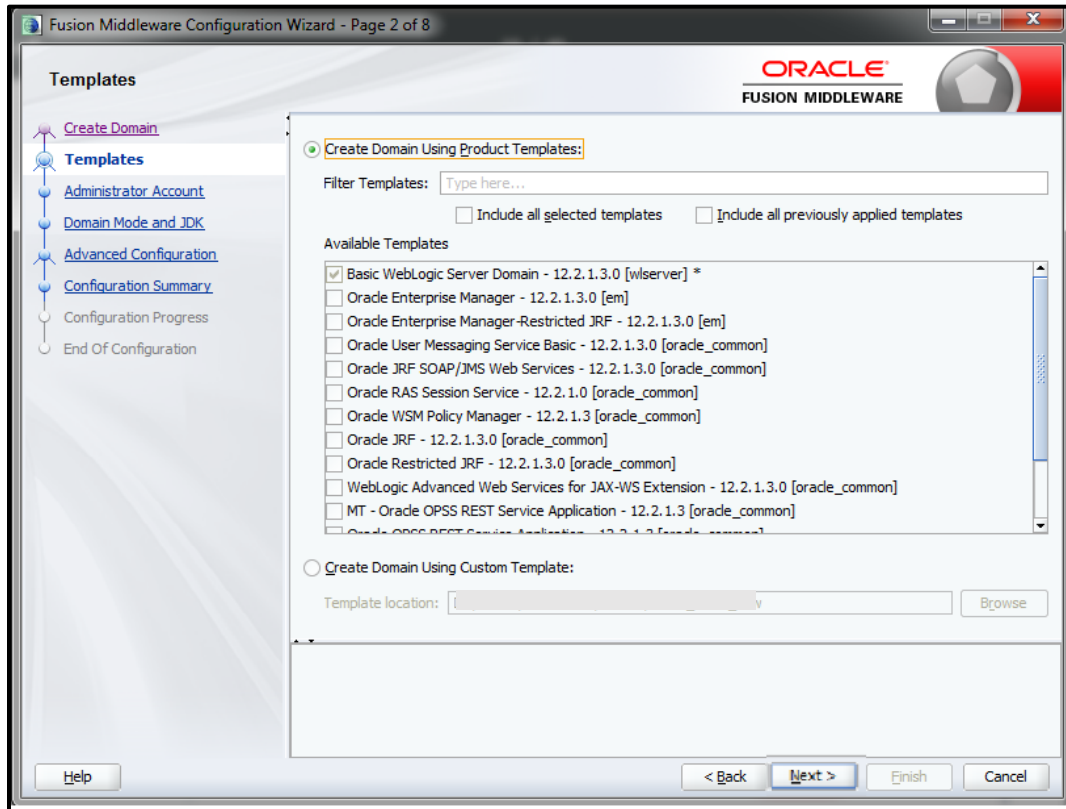
Perform the following steps for domain and cluster configuration:

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

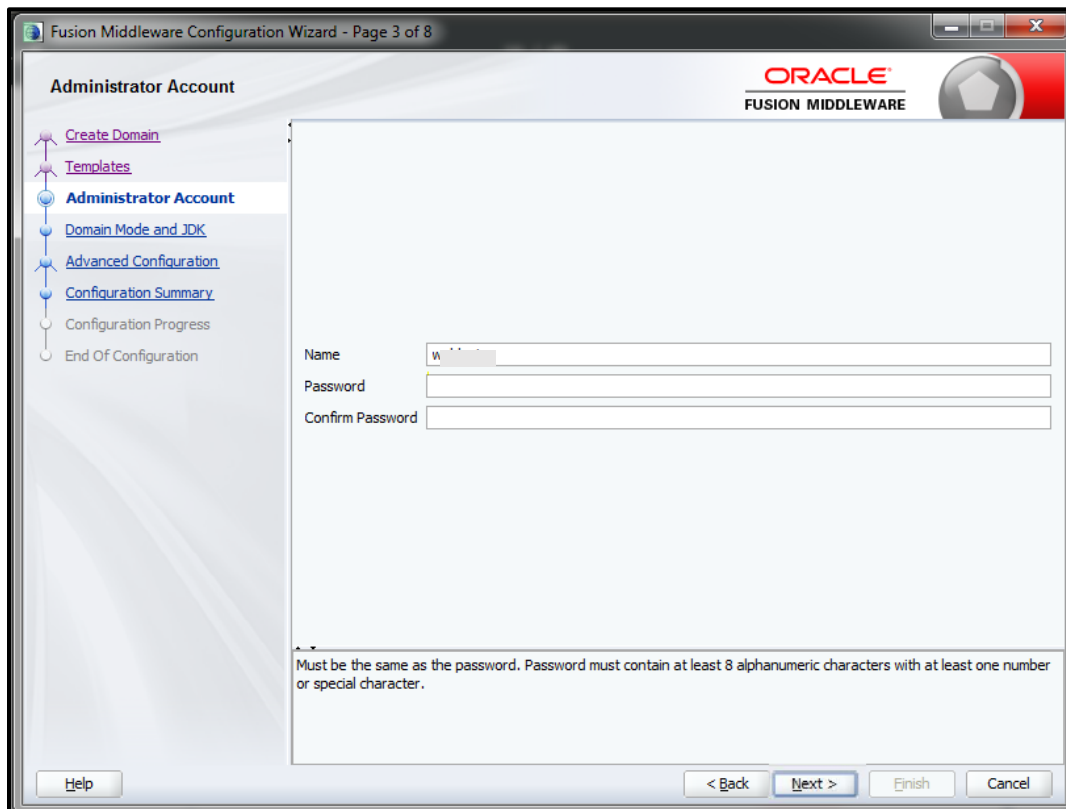
1. Open **/oracle_common/common/bin** and run **config.cmd** (or **.sh** if operating system is linux). Create domain with required cluster and server configurations. Refer to the screenshots below.
2. Select **Create a new domain** and provide domain name. For example, **platoinfra_domain**.



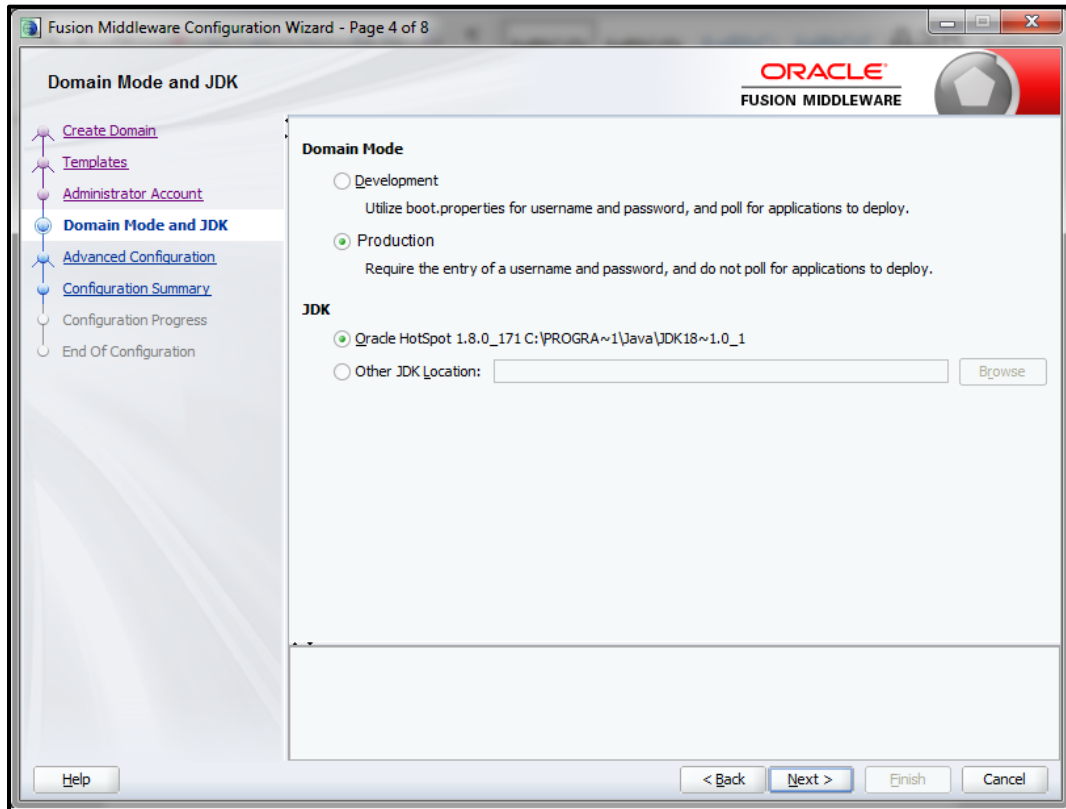
3. Click **Next** to create simple domain with default templates.



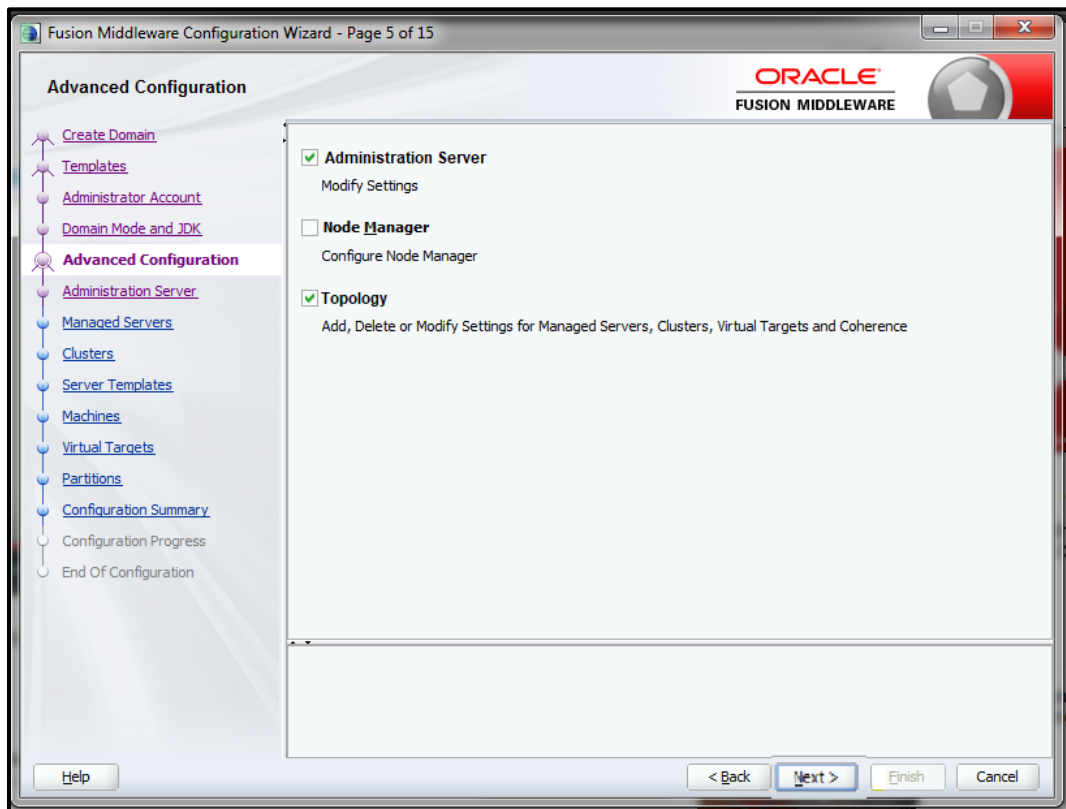
4. Set password and confirm, click **Next** to proceed.



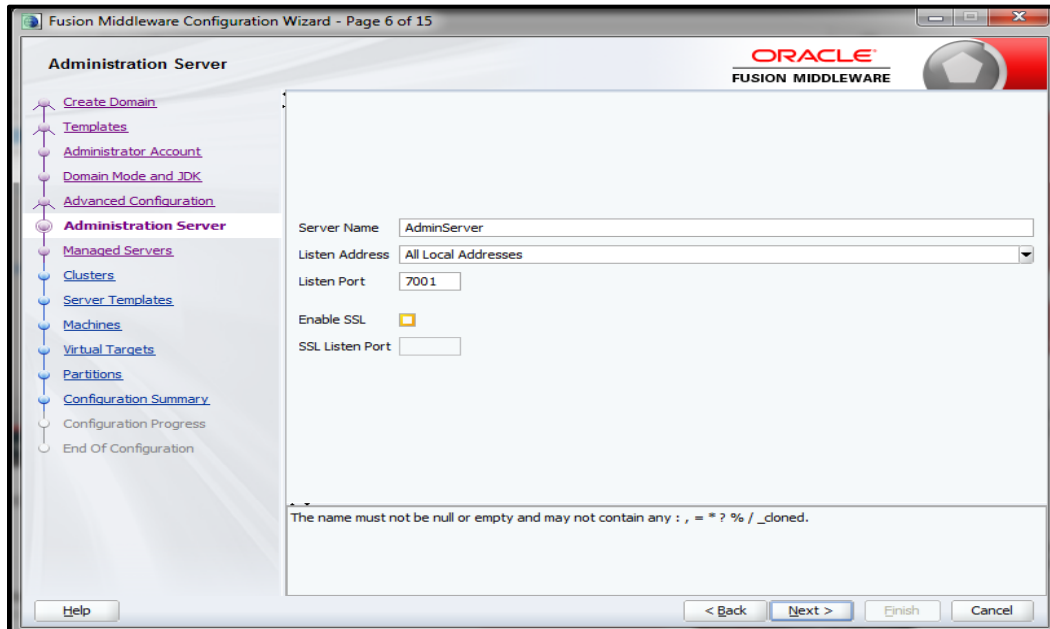
5. Select **Domain Mode** as **Production** and select **JDK**.



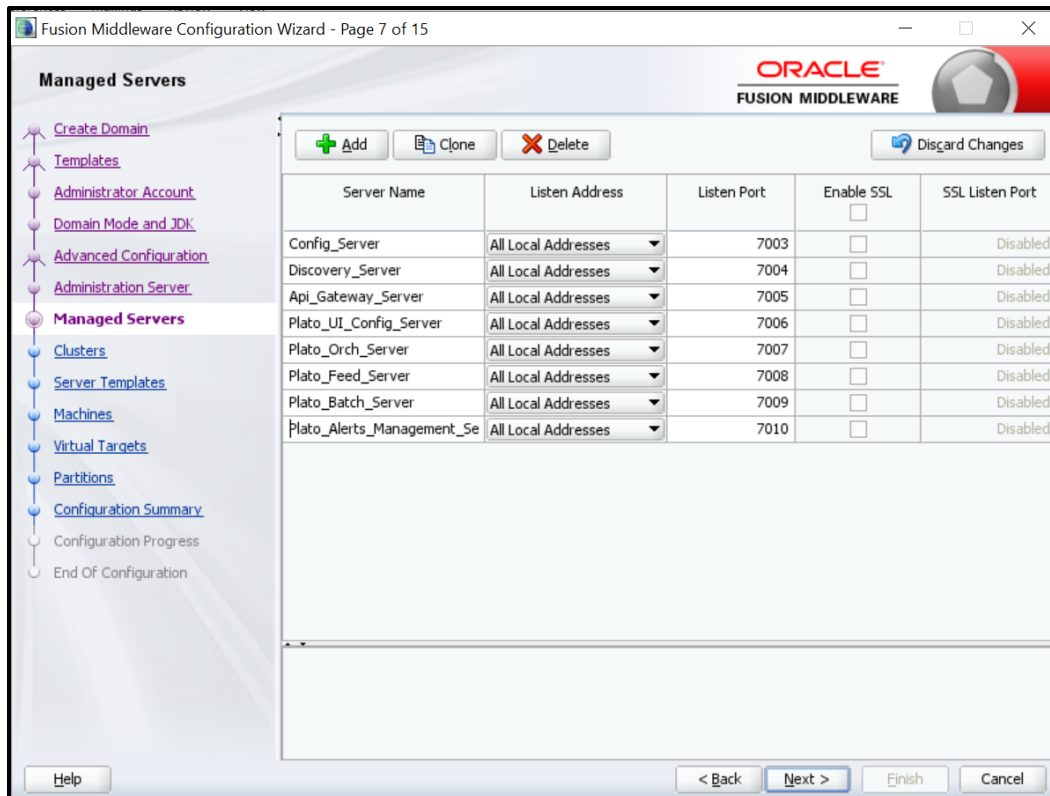
6. Select **Administration Server** and **Topology** in advanced configurations.



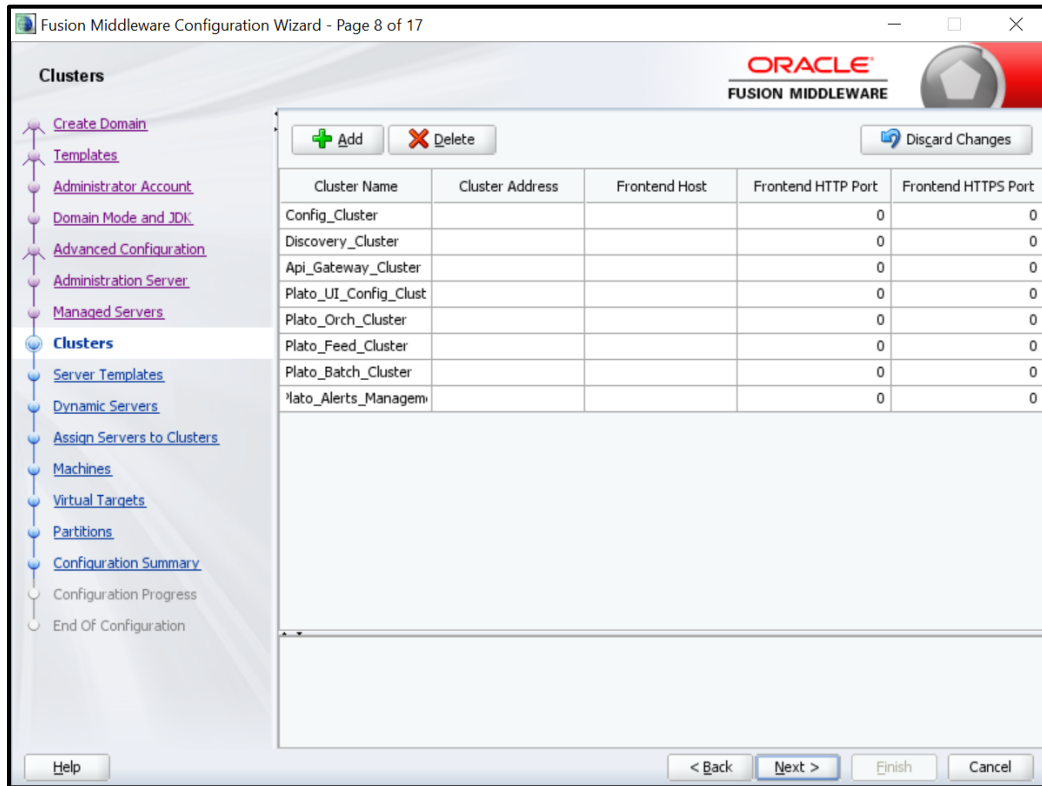
7. Edit the port and host configurations as required and click **Next**.



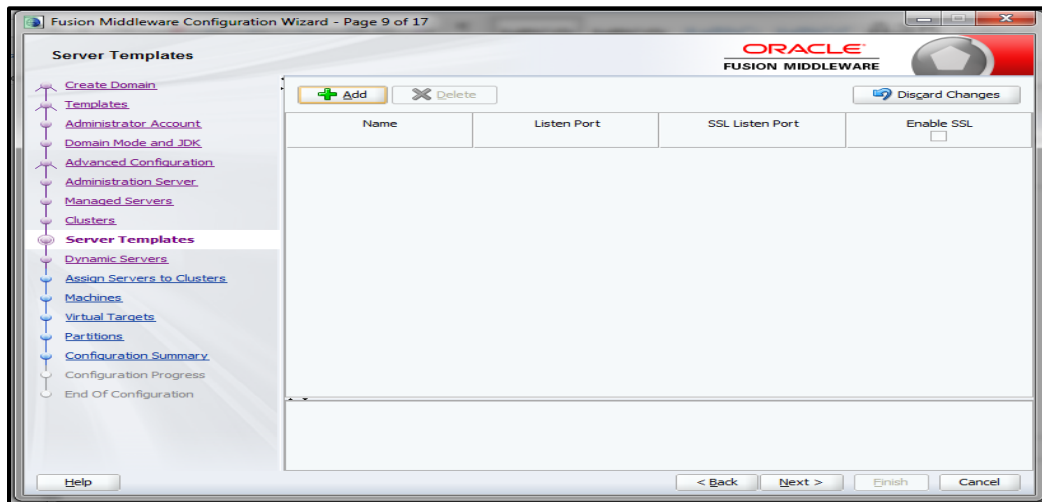
8. Add managed servers and provide meaningful **Server Name**, edit listen address and port as required.



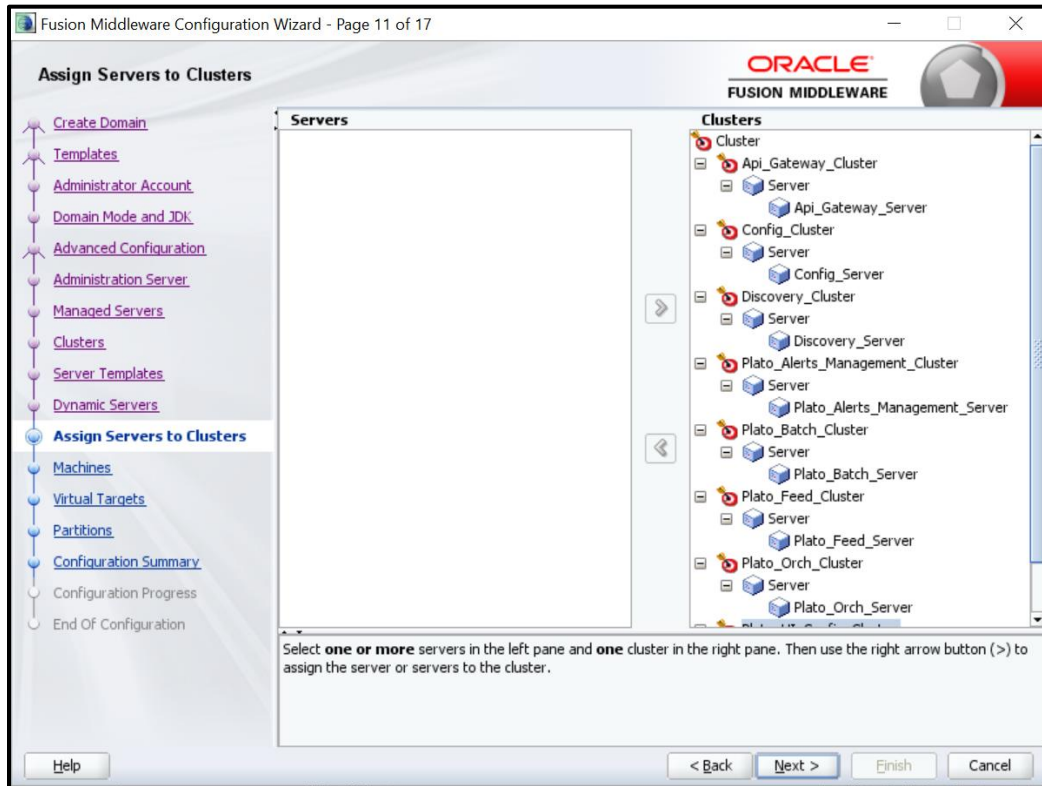
9. Add clusters one for each **managed servers**.



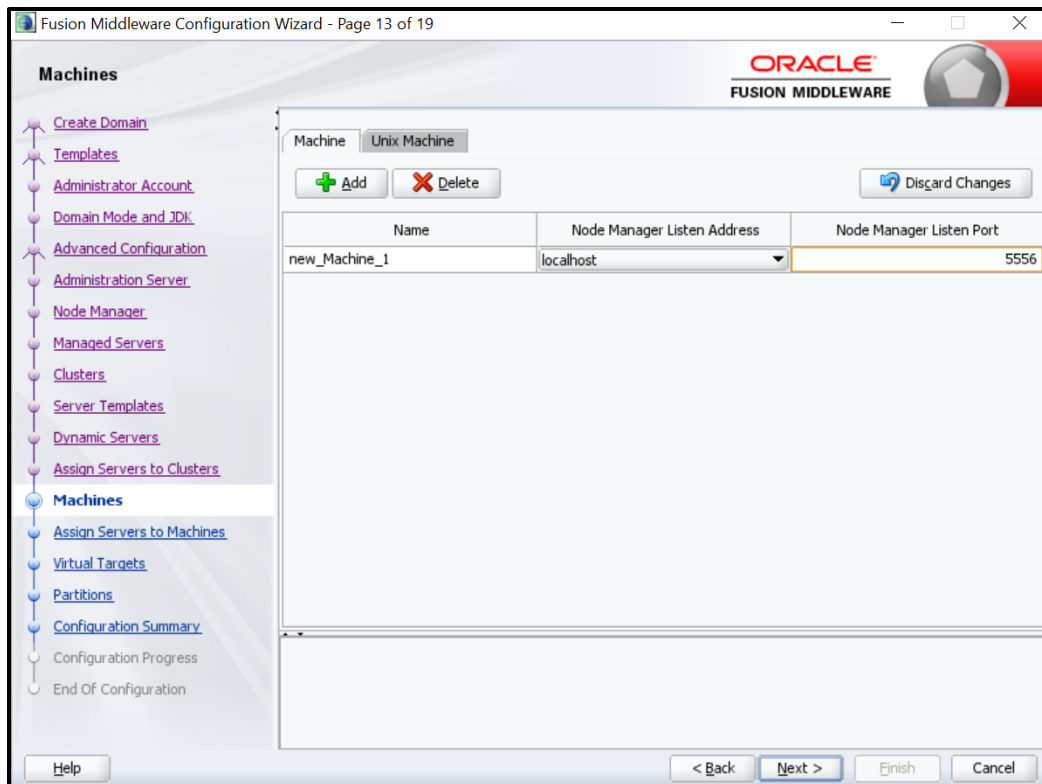
10. Skip **Server Templates** and **Dynamic Servers**.



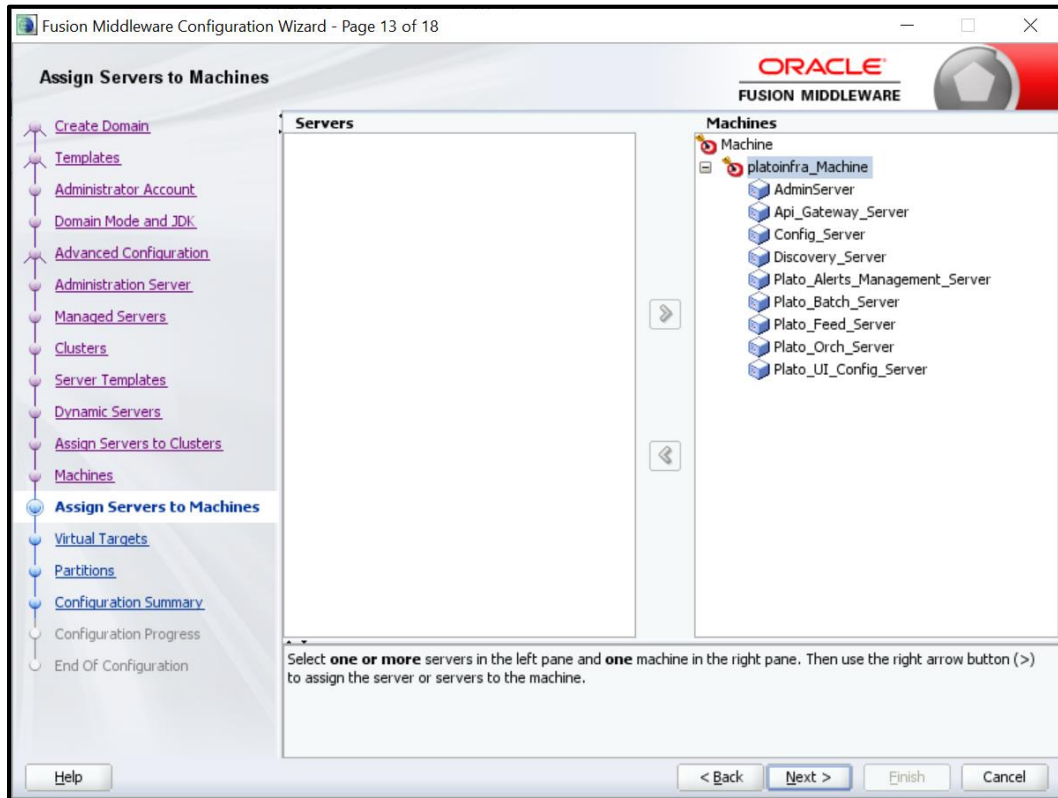
11. Assign clusters with servers.



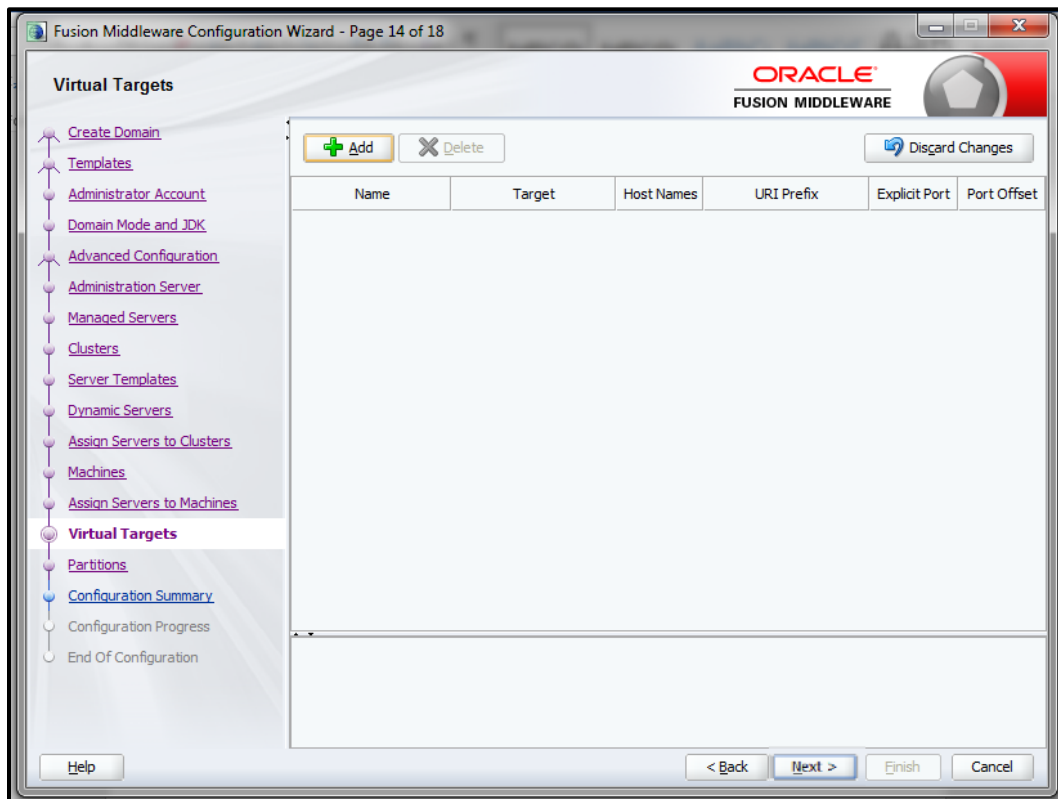
12. Add Machine/Unix Machine based on operating system and configure Name, Node Manager Listen Address and Node Manager Listen Port as required.

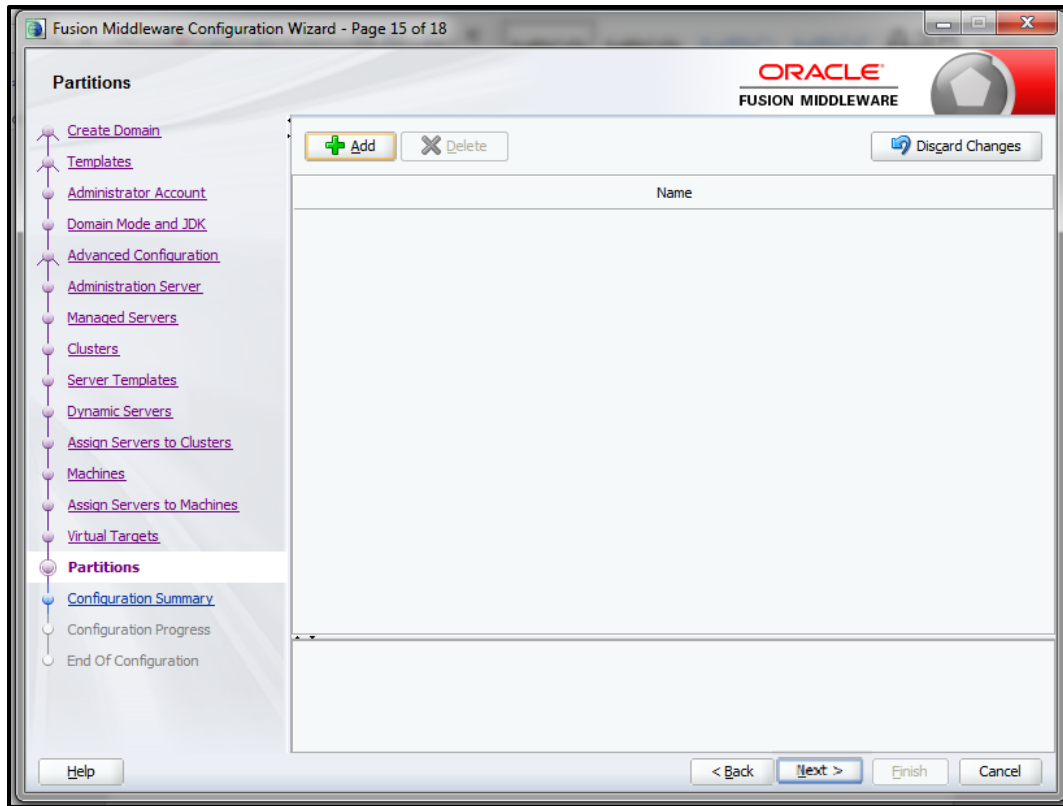


13. Map all managed servers under the machine created.

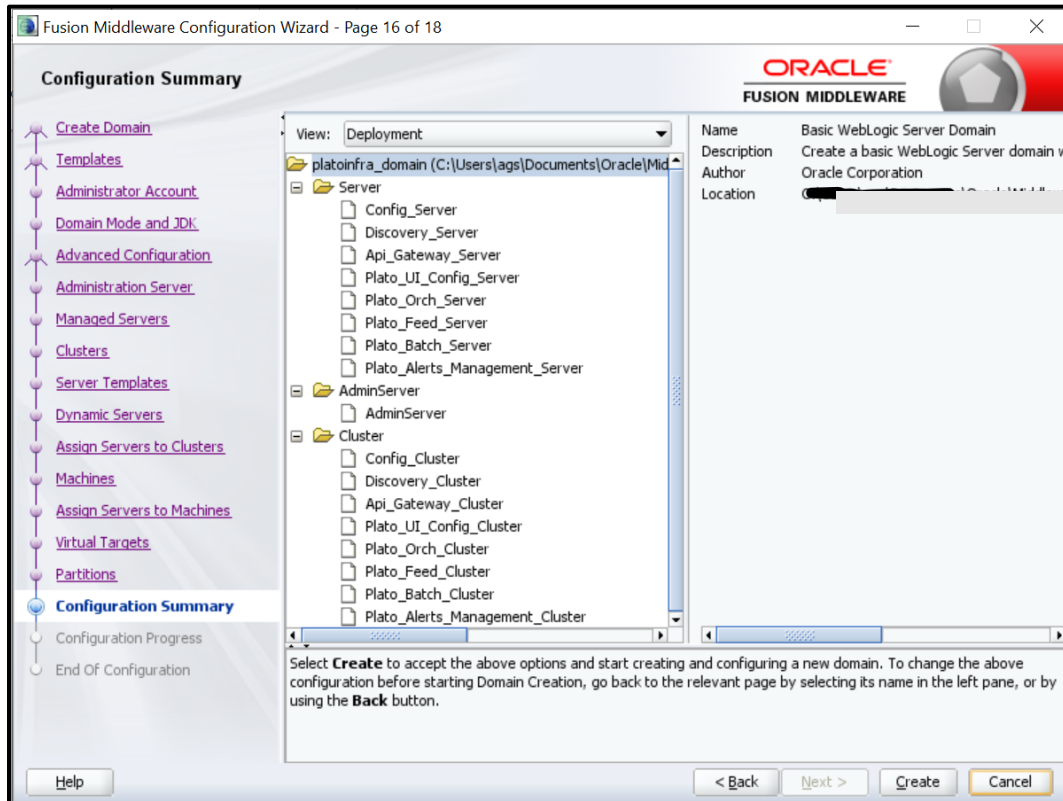


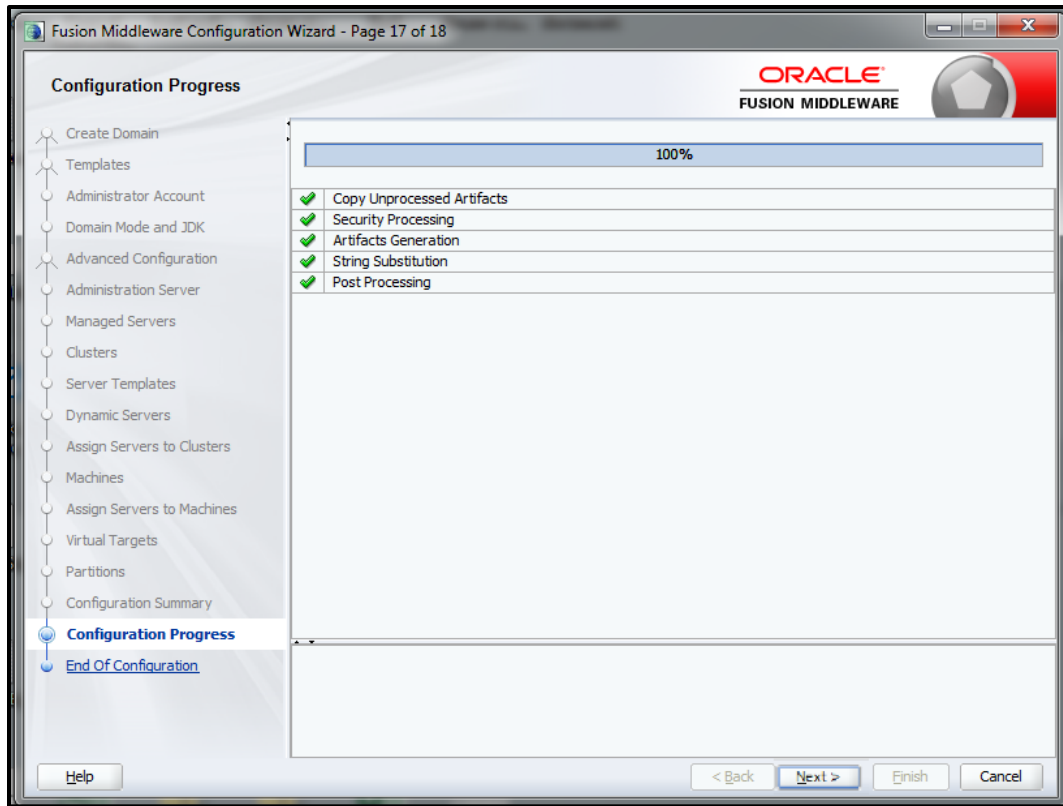
14. Skip or configure **Virtual Targets** and **Partitions** as required.



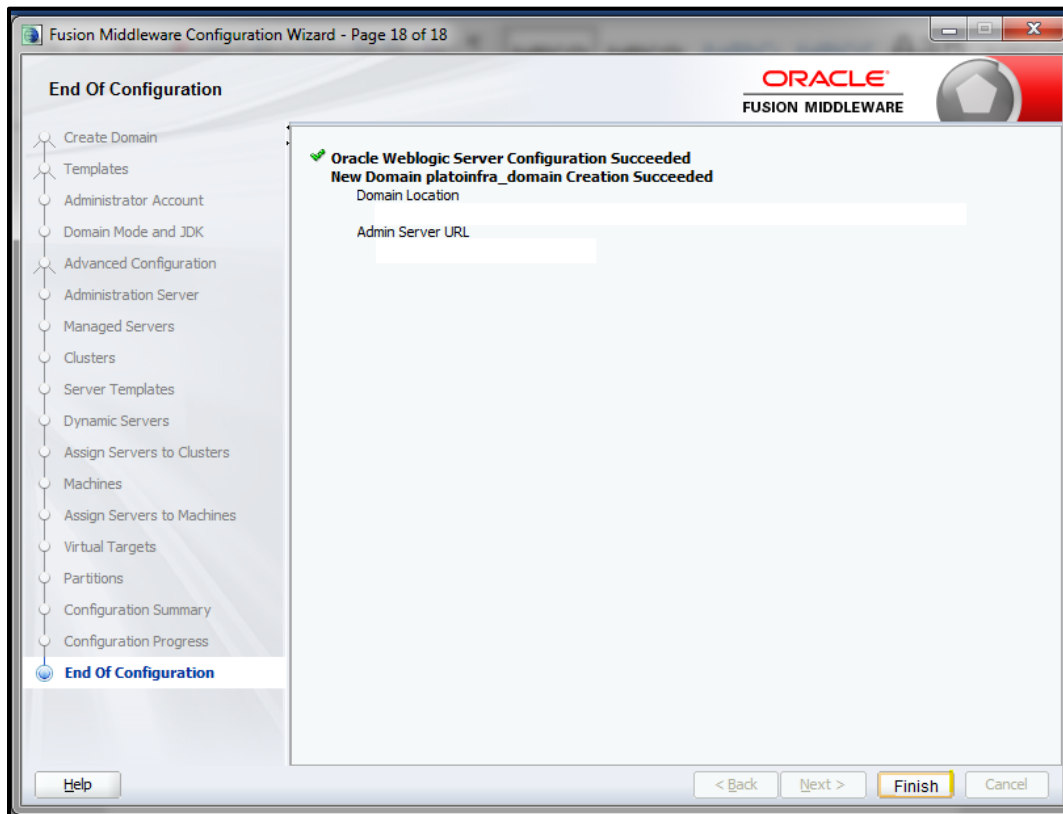


15. Check the **Configuration Summary** and confirm creating domain.





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:

The screenshot shows the Oracle WebLogic Administration Console configuration page for servers. The page includes a navigation pane on the left with 'Domain Structure' and 'How do I...'. The main content area shows a table of servers with the following data:

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.

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

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.

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_UL_Config_Cluster		Unicast	Database	Round Robin	(None)		P

Change Center
View changes and restarts
Click the **Lock & Edit** button to modify, add or delete items in this 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...
• Configure clusters
• Assign server instances to clusters
• Configure server migration in a cluster
• Configure cross-cluster replication
• Create dynamic clusters

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.

Name	Type
platoinfra_Machine	Machine

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**

View changes and restarts

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

Domain Structure

- platoinfra_domain
 - [-] Domain Partitions
 - [-] Environment
 - [-] Deployments
 - [-] Services
 - [-] Messaging
 - [-] Data Sources
 - [-] Persistent Stores
 - [-] Foreign JNDI Providers
 - [-] Work Contexts
 - [-] XML Registries
 - [-] XML Entity Caches
 - [-] jCOM

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 through a pool of JDBC connections. Applications can look up a data source on the 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)

Generic Data Source	Type	JNDI Name	Targets
There are no items to display			
GridLink Data Source			
Multi Data Source			
Proxy Data Source			
UCP Data Source			

Domain Structure

- platoinfra_domain
 - [-] Domain Partitions
 - [-] Environment
 - [-] Deployments
 - [-] Services
 - [-] Messaging
 - [-] Data Sources
 - [-] Persistent Stores
 - [-] Foreign JNDI Providers
 - [-] Work Contexts
 - [-] XML Registries
 - [-] XML Entity Caches
 - [-] jCOM
 - [-] Mail Sessions

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.

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. We 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.

The screenshot shows the 'Servers' configuration page for a JDBC data source. On the left, a tree view shows the domain structure. The main area is divided into 'Servers' and 'Clusters' sections. The 'Clusters' section is expanded, showing the following clusters and their target servers:

- Api_Gateway_Cluster:** Part of the cluster (selected), with Api_Gateway_Server.
- Config_Cluster:** Part of the cluster (selected), with Config_Server.
- Discovery_Cluster:** Part of the cluster (selected), with Discovery_Server.
- Plato_Alerts_Management_Cluster:** Part of the cluster (selected), with Plato_Alerts_Management_Server.
- Plato_Batch_Cluster:** Part of the cluster (selected), with Plato_Batch_Server.
- Plato_Feed_Cluster:** Part of the cluster (selected), with 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](#)

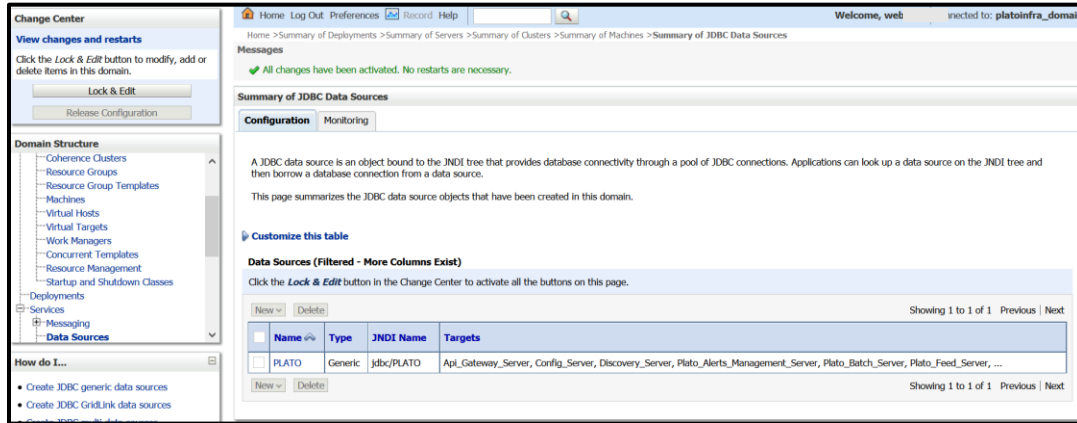
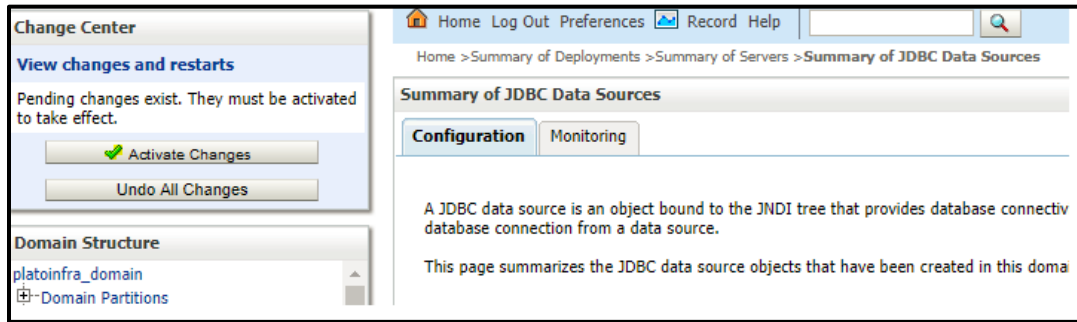
Data Sources (Filtered - More Columns Exist)

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

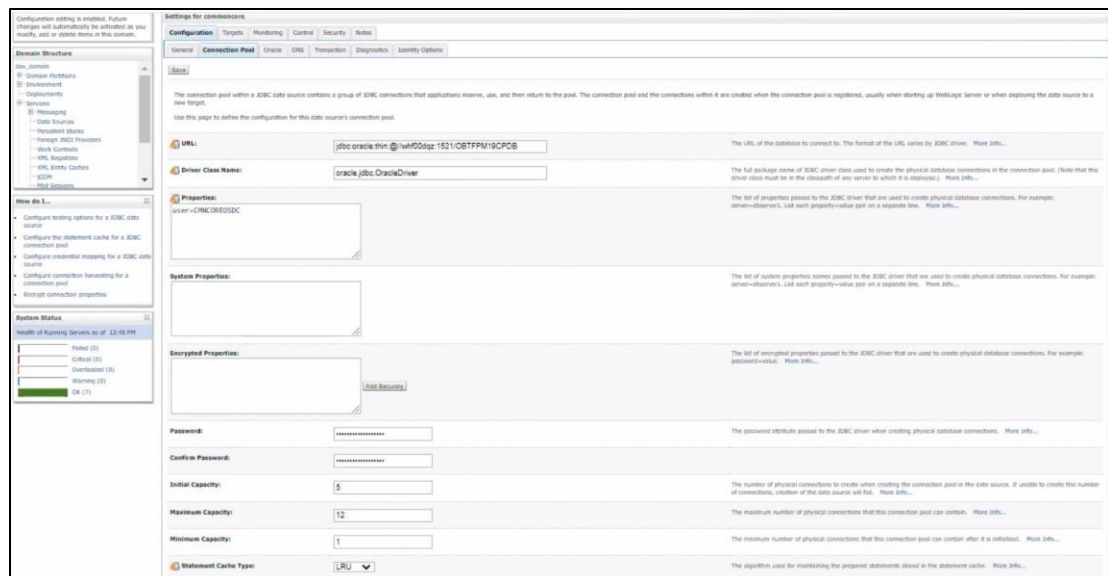
<input type="checkbox"/>	Name	Type	JNDI Name	Targets
<input type="checkbox"/>	PLATO	Generic	jdbc/PLATO	ApiGateway_Server, Config_Server, Discovery_Server, Plato_Alerts_Management_Server, Plato_Batch_Server, Plato_Feed_Server, ...

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

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



- Post creation of Data source, click on “Data source” under “Services” menu. Select the specific Data sources one by one and go to “Configuration” → “Connection Pool” section and update the initial capacity, Minimum capacity and Max capacity as explained in the below screenshot.



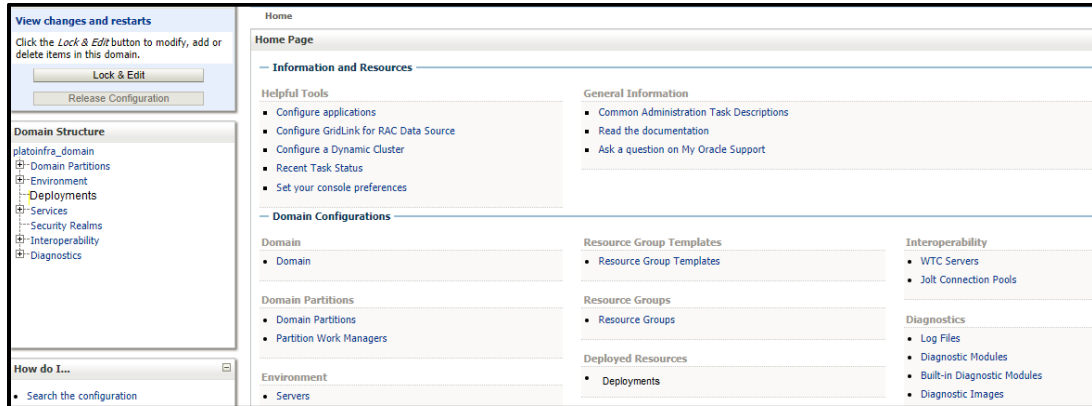
Note :

It is the sample configuration. The Max capacity may increase based on the concurrent user and transaction volume

1.5 How to Deploy Application

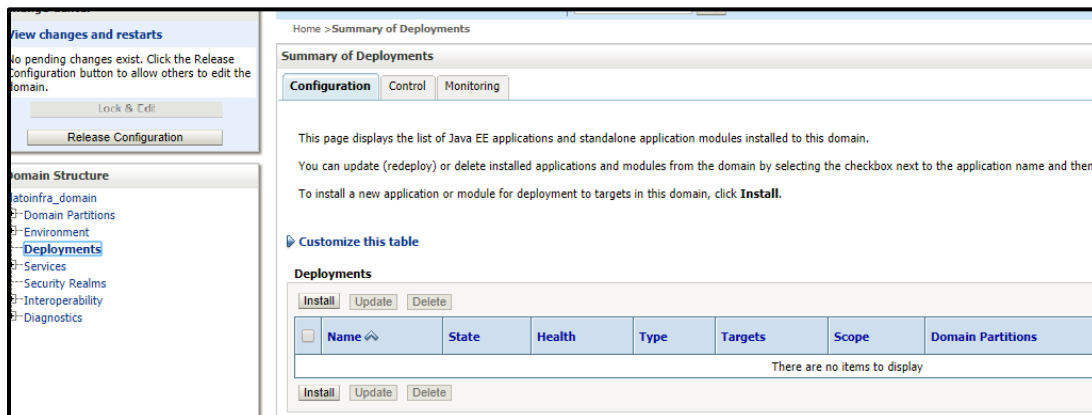
Steps to Deploy archives as application on weblogic is same for all the above except for managed server and domain, where we deploy will differ. Perform the following steps to see how deployment of archive as application is done on weblogic:

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

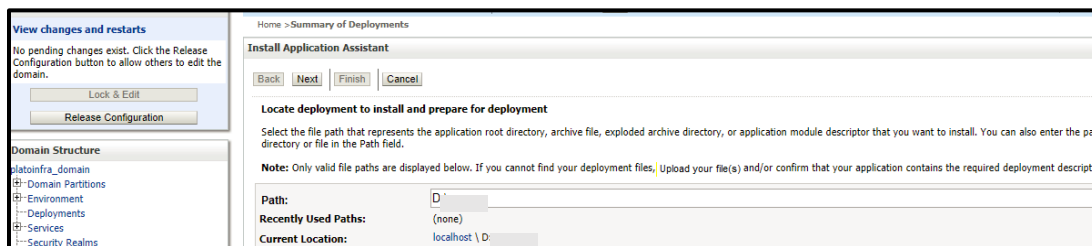


The screenshot shows the Oracle WebLogic Administration Console. On the left, the 'Domain Structure' tree is expanded to 'Deployments'. The main content area is titled 'Home Page' and contains several sections: 'Information and Resources' with helpful tools like 'Configure applications' and 'Recent Task Status'; 'Domain Configurations' with links for 'Domain', 'Domain Partitions', and 'Environment'; and 'Deployed Resources' with a link for 'Deployments'. There are also sections for 'Resource Group Templates', 'Resource Groups', and 'Diagnostics'.

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

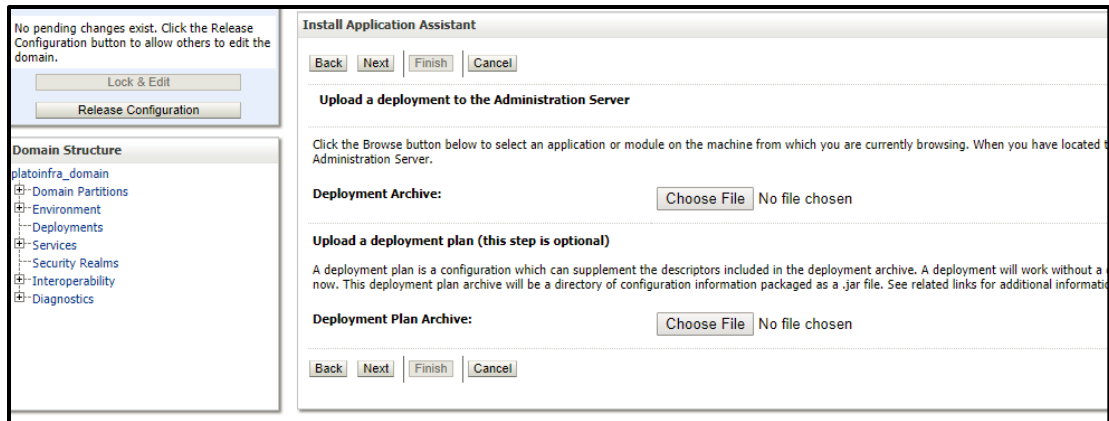


The screenshot shows the 'Summary of Deployments' page. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Summary of Deployments' and has tabs for 'Configuration', 'Control', and 'Monitoring'. It contains instructions: 'This page displays the list of Java EE 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 To install a new application or module for deployment to targets in this domain, click **Install**.' Below this is a table with columns: Name, State, Health, Type, Targets, Scope, and Domain Partitions. The table is currently empty, with the text 'There are no items to display' below it. There are 'Install', 'Update', and 'Delete' buttons above and below the table.

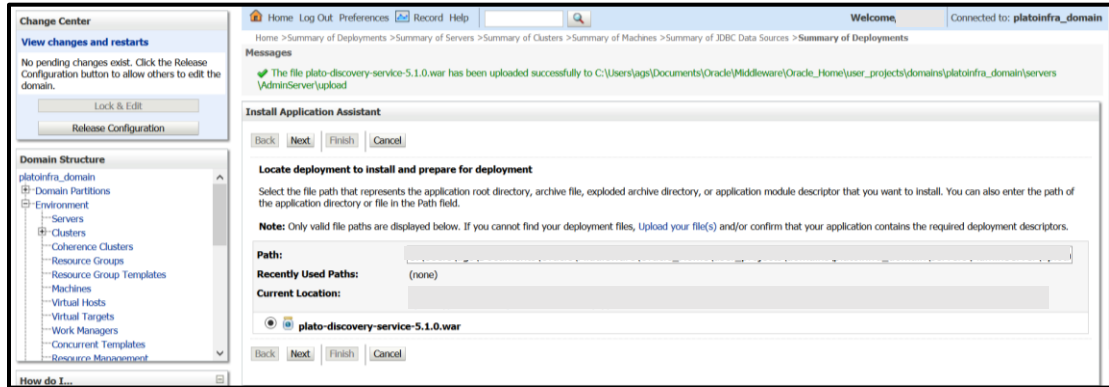
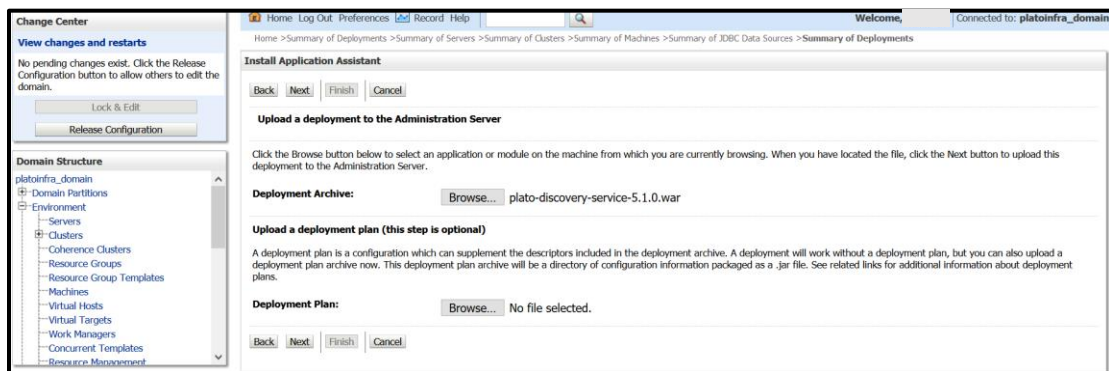


The screenshot shows the 'Install Application Assistant' dialog. The left sidebar is the same as in the previous screenshots. The main content area is titled 'Install Application Assistant' and has buttons for 'Back', 'Next', 'Finish', and 'Cancel'. It contains instructions: 'Locate deployment to install 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 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 descriptor.' Below this is a 'Path' field with a text input box containing 'D:\'. There are also 'Recently Used Paths' and 'Current Location' sections.

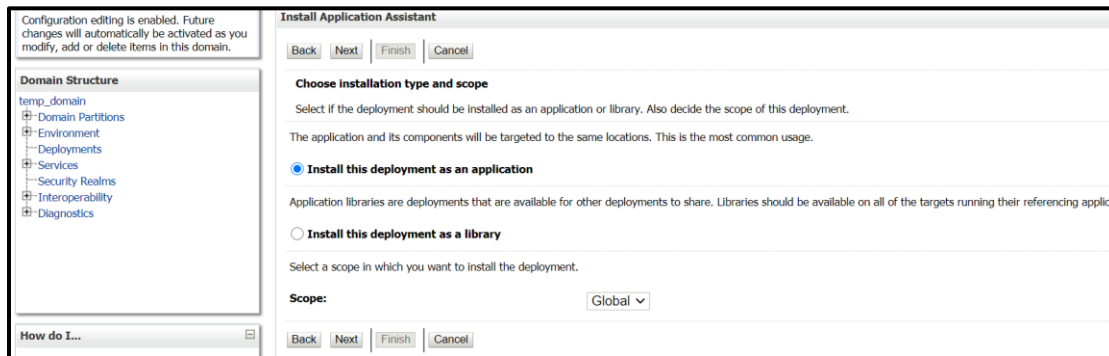
3. Click **Upload your file(s)** to select archive, **Choose File** and click **Next**.



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



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



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

- 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

Change Center

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

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

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

Install Application Assistant

Back Next Finish Cancel

Optional Settings

You can modify these settings or accept the defaults.

* Indicates required fields

General

What do you want to name this deployment?

* Name: plato-discovery-service-5.1.0

Security

What security model do you want to use with this application?

- DD Only: Use only roles and policies that are defined in the deployment descriptors.**
- Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor.
- Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console.
- Advanced: Use a custom model that you have configured on the realm's configuration page.

Source Accessibility

How should the source files be made accessible?

- Use the defaults defined by the deployment's targets

Deploy Desktop Update

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

Change Center

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

View changes and restarts

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

Activate Changes
Undo All Changes

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

How do I...

- Install an enterprise application
- Configure an enterprise application
- Update (redeploy) an enterprise application

Messages

- The deployment has been successfully installed.
- You must also activate the pending changes to commit this, and other updates, to the active system.

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**.

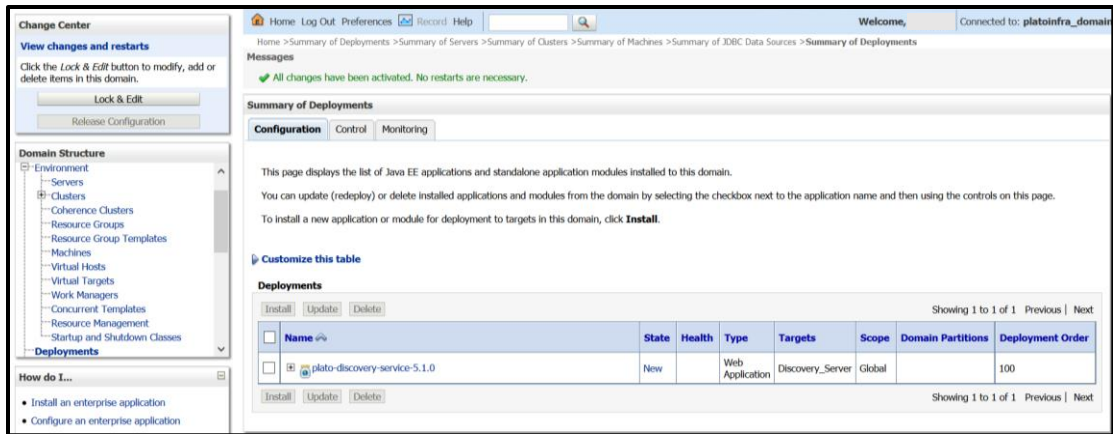
Customize this table

Deployments

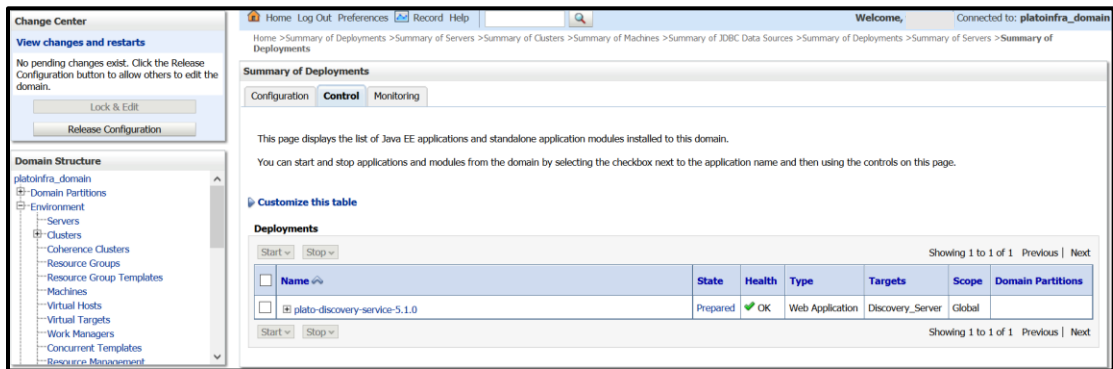
Install Update Delete Showing 1 to 1 of 1 Previous Next

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

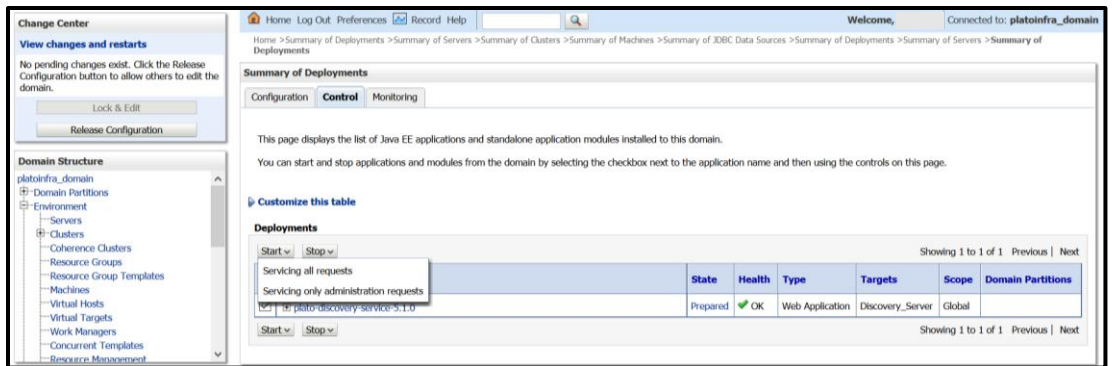
Install Update Delete Showing 1 to 1 of 1 Previous Next



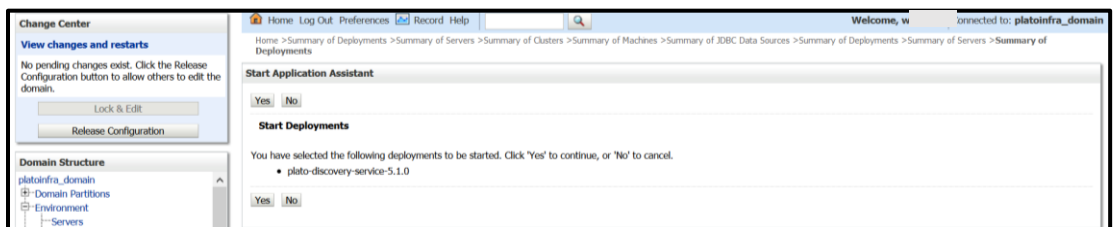
8. Click **Deployments** and then **Control** to changes the state of application from prepared to active status.



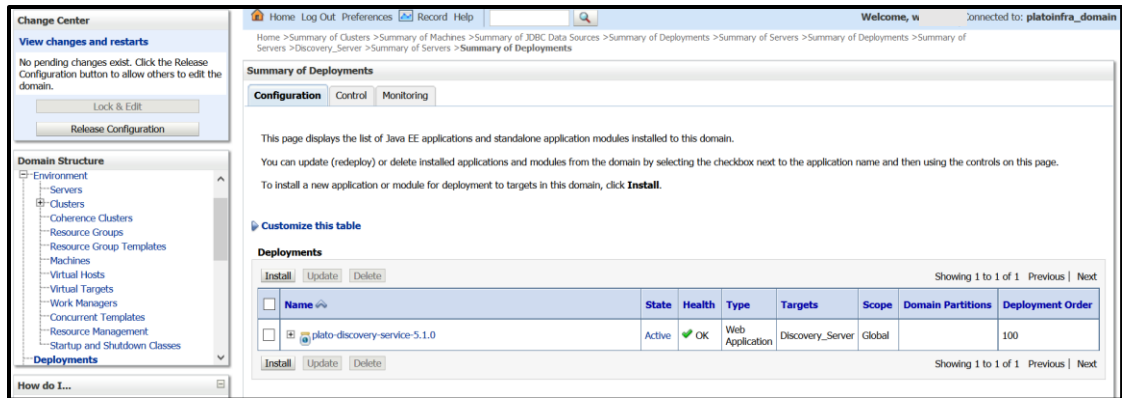
9. Under **Deployment**, click **Start** dropdown and select **Start all requests**.



10. Click **Yes**.

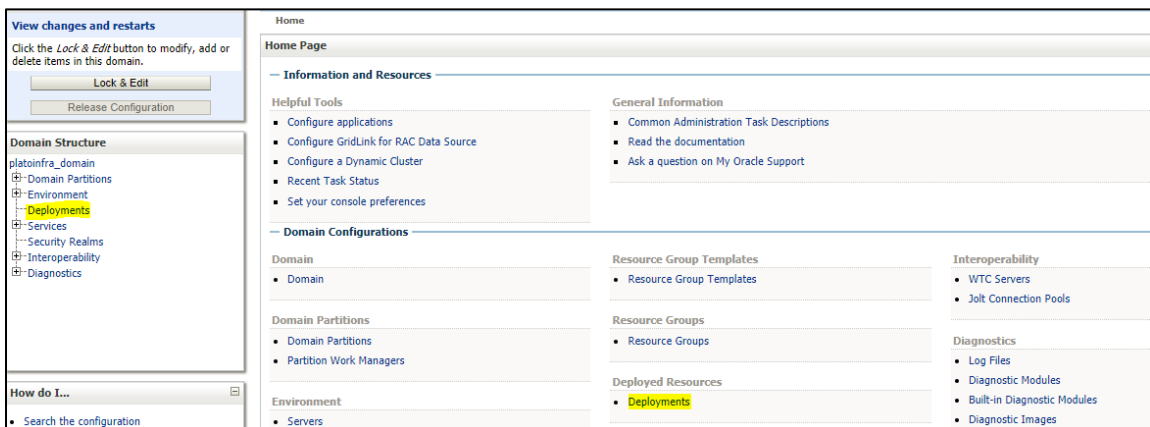


11. The status is displayed as **Active** in the state column.

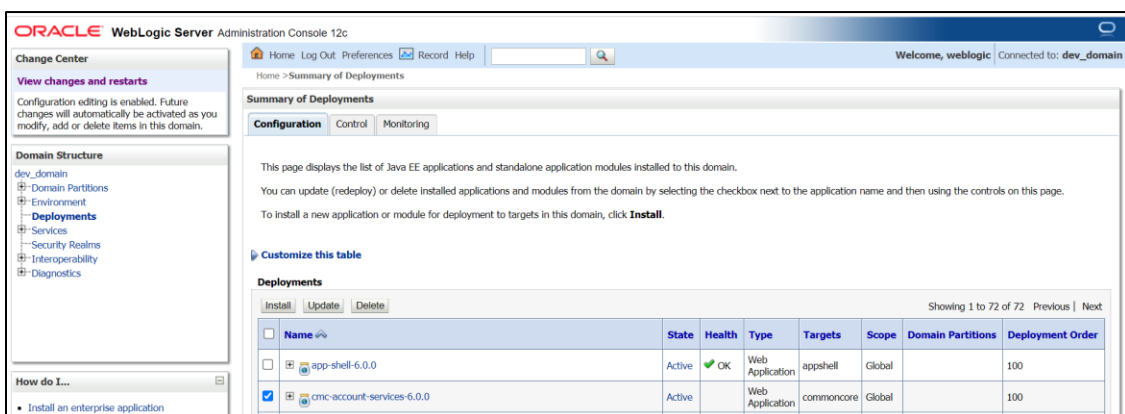


1.6 How to Undeploy application:-

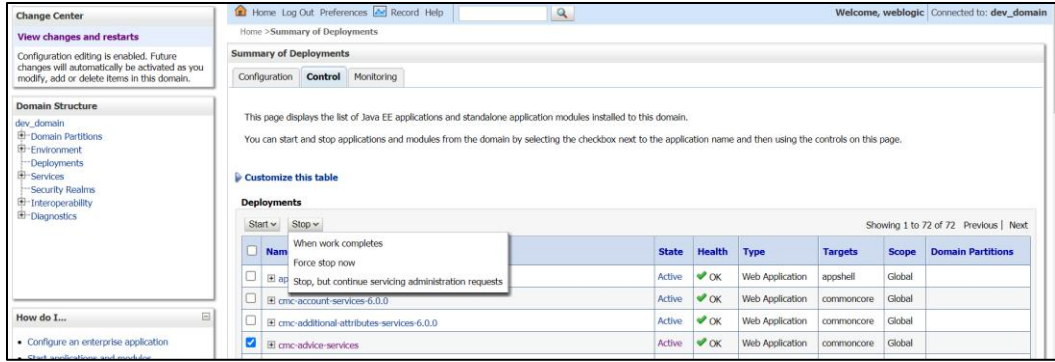
1. Login into with weblogic server with proper credentials. Post login, go to **Deployments** section



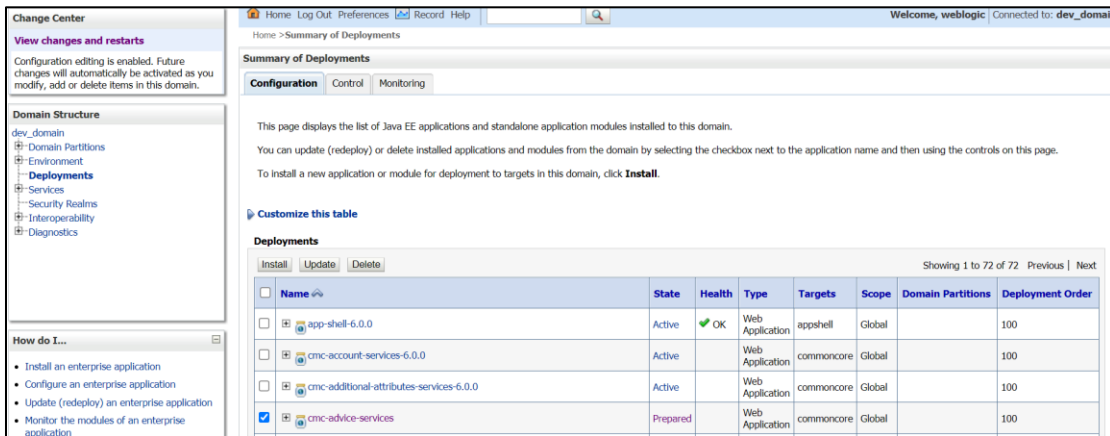
2. Click on **Lock and Edit** and then select the service that needs to be undeployed in Deployments.



3. Click on **Control**. Click on Force stop now under Stop.



4. Once it changes to prepared state, click on Configuration.

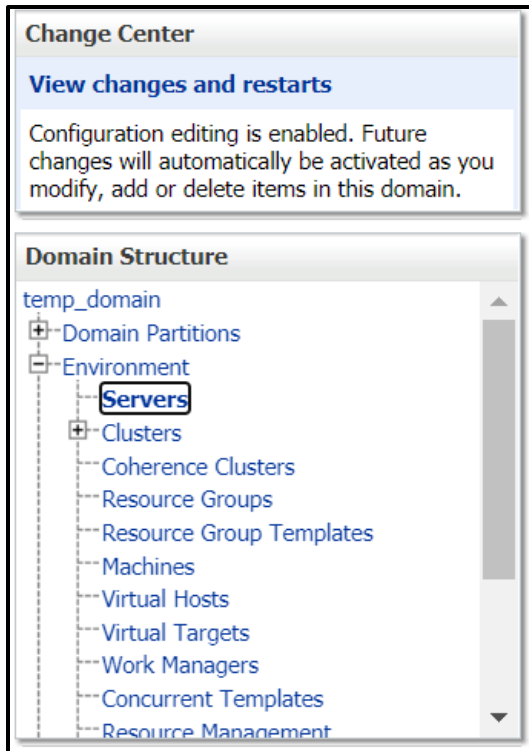


5. Select the service again and click on Delete to undeploy the service.

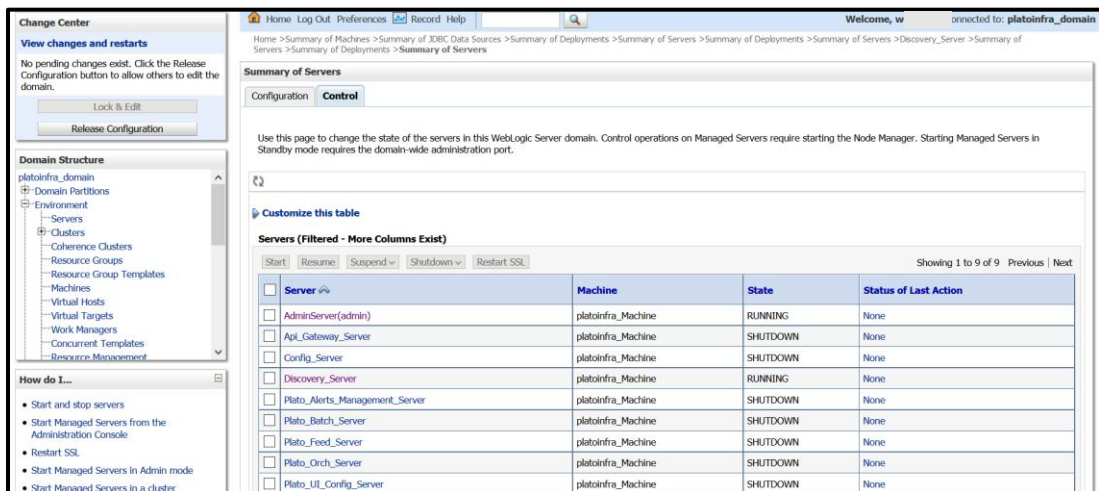
1.7 How to Restart Servers

Perform the following steps to restart servers:

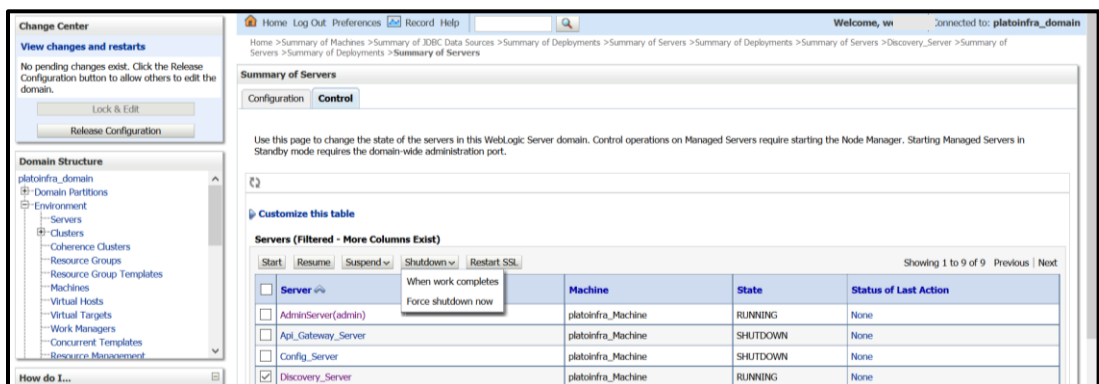
1. Navigate to left menu and select **Environment**, and then click **Servers**.



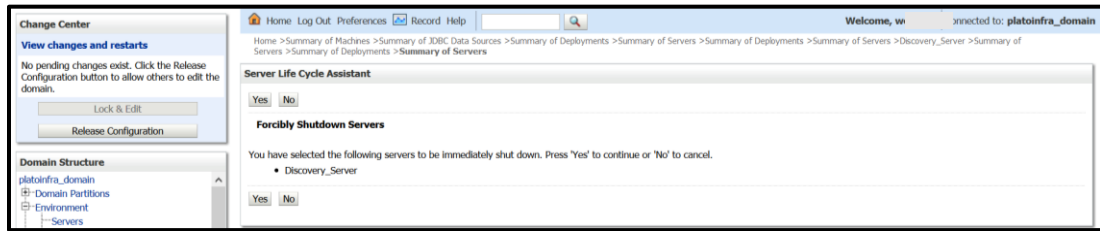
2. Click **Control** tab.



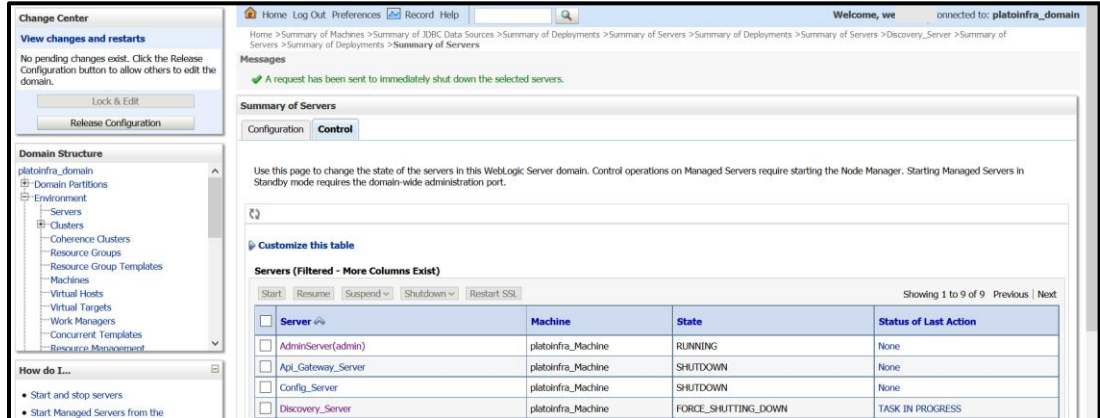
3. Select servers to **Shutdown**



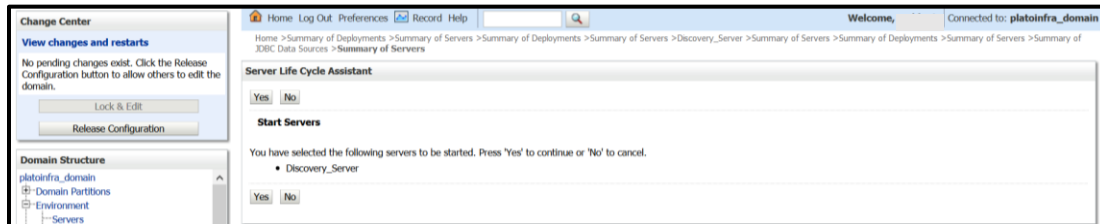
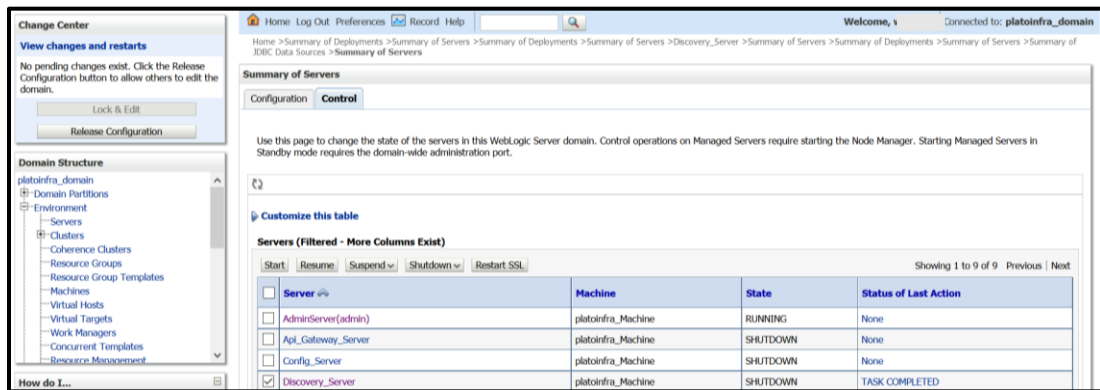
4. Click **Yes** to confirm shutdown.

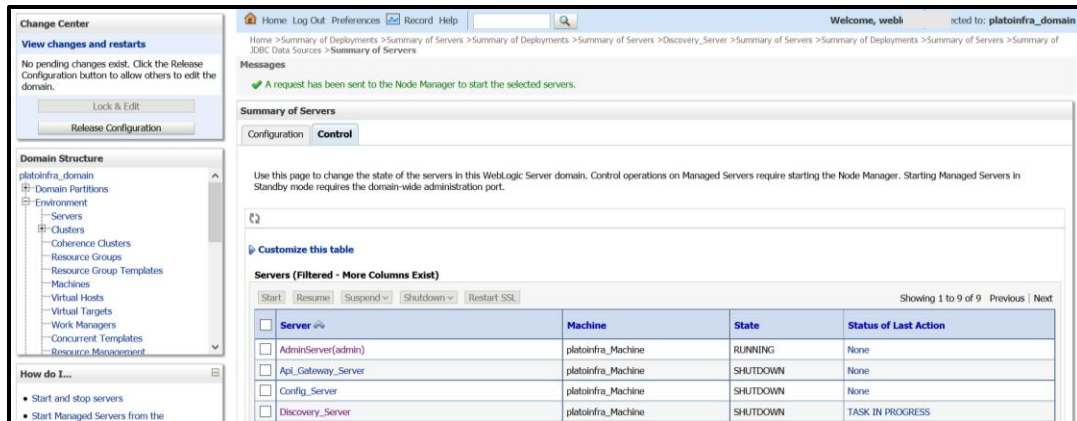


5. The status displayed as shown below:

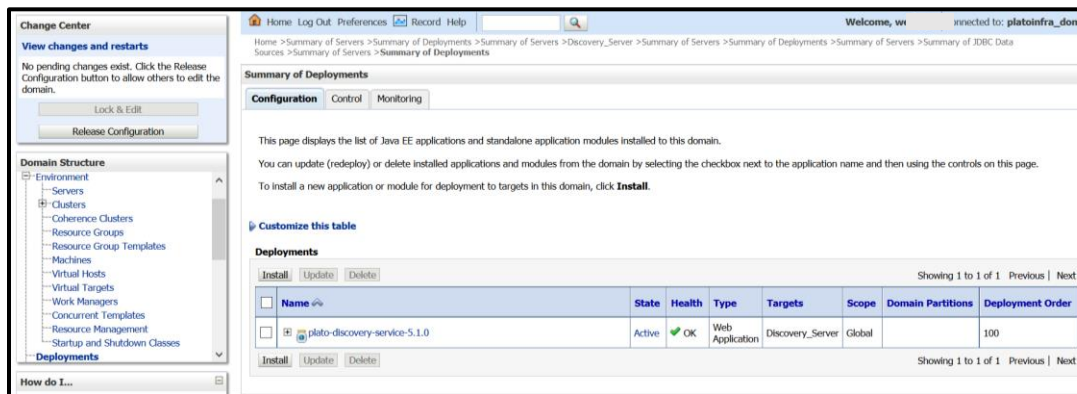
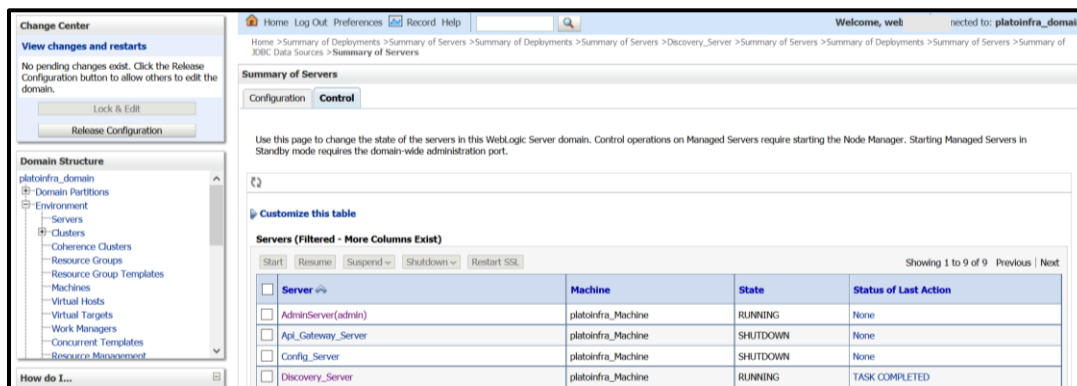


6. Once shutdown is completed, navigate to **Control**, select the servers to **Start**, and click **Yes** to confirm action.





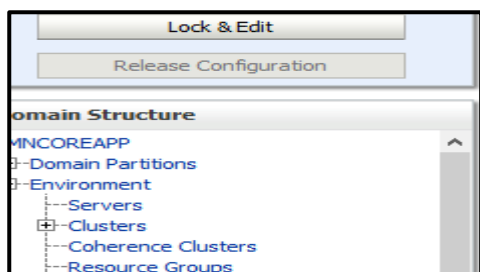
- When all requested servers are running, navigate to **Summary of Deployments**, and check if deployments are in active state.



1.8 How to Check Port Number

Perform the following steps to check port numbers:

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



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

<input type="checkbox"/>	Name	Type	Cluster	Machine	State	Health	Listen Port
<input type="checkbox"/>	AdminServer(admin)	Configured			RUNNING	✔ OK	7020
<input type="checkbox"/>	managed_server1	Configured		Machine 1	RUNNING	✔ OK	7023

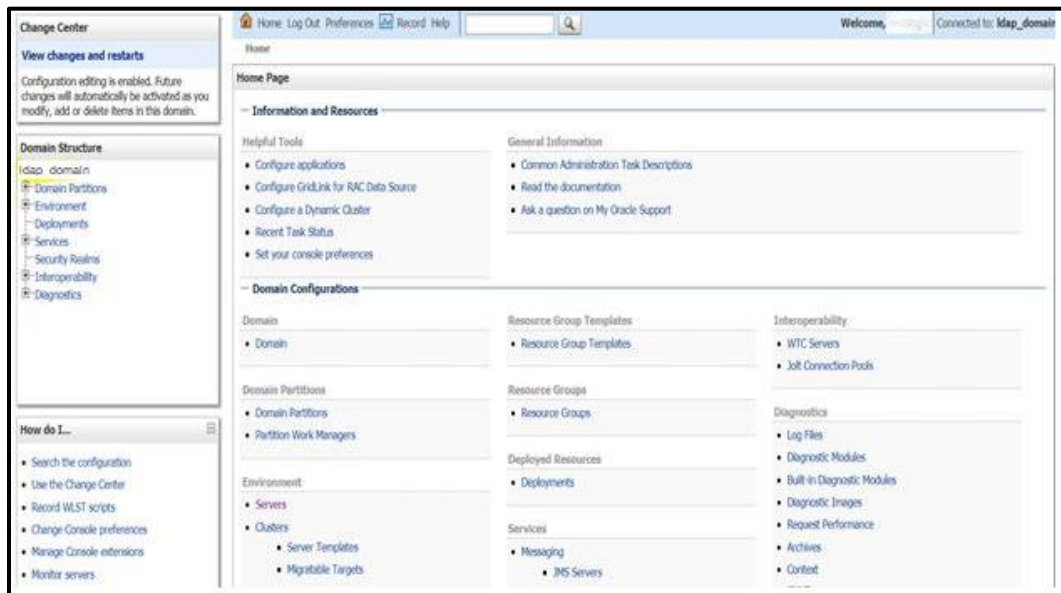
1.9 Weblogic Embedded LDAP Setup

The following changes are to be made for configuring the Weblogic-Embedded LDAP server for PLATO:

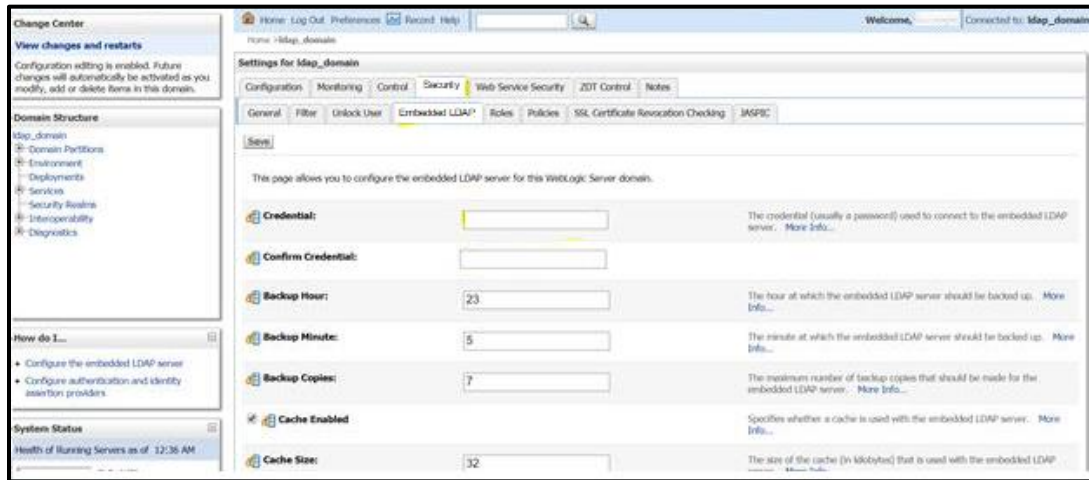
- Configuration of Weblogic LDAP
- Creation of Users
- Plato Security Config Table Entries

1.9.1 Configuration of Weblogic LDAP

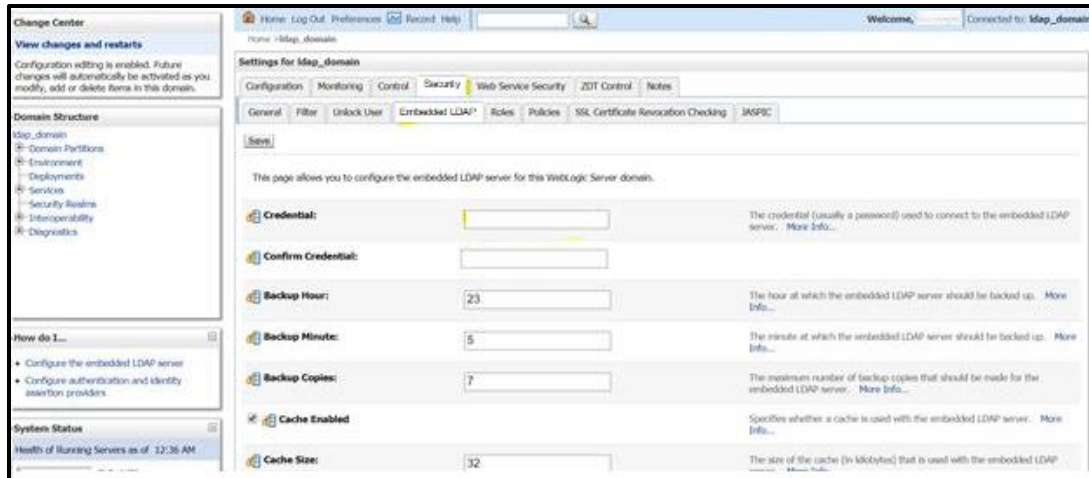
- Open the **Administration Console** for the **Weblogic**, and click domain name in left panel.



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

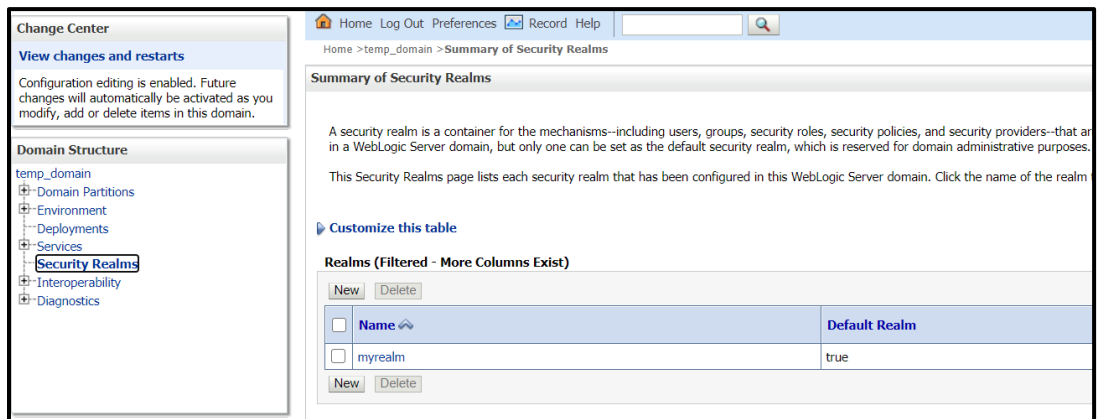


- Set the **Credential** for Weblogic Embedded LDAP store. This is needed in the **Security Config** table.



1.9.2 Creation of Users

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



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

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

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

New Delete

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

Create a New Group

OK Cancel

Group Properties

The following properties will be used to identify your new Group.

* Indicates required fields

What would you like to name your new Group?

* Name: Developers

How would you like to describe the new Group?

Description: Group for Developers

Please choose a provider for the group.

Provider: DefaultAuthenticator

OK Cancel

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

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...

- Create groups
- Modify groups
- Delete groups

Home Log Out Preferences Record Help

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

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.

New Delete

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

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

Create a New User

OK | Cancel

User Properties

The following properties will be used to identify your new User.
* Indicates required fields

What would you like to name your new User?

* **Name:**

How would you like to describe the new User?

Description:

Please choose a provider for the user.

Provider:

The password is associated with the login name for the new User.

* **Password:**

* **Confirm Password:**

OK | Cancel

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

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

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

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

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:

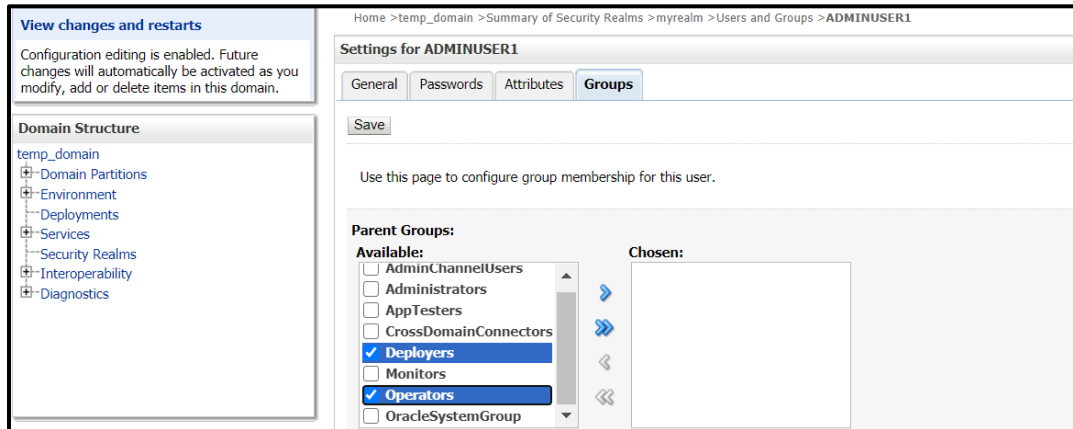
Available:

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

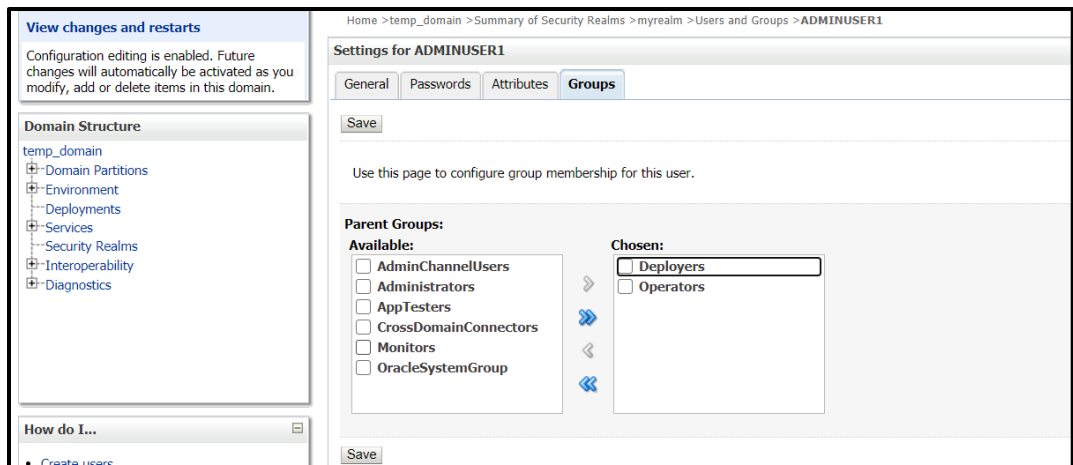
Chosen:

Save

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



11. Click **Save**.



1.9.3 Plato 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

ID	VALUE	Description
LDAP_SERVER_BASE	dc=ldap_domain	LDAP Server Base
LDAP_SERVER_CREDENTIAL	ylksiMFjVbfcpA7Qheh8Q==	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.



ANNEXURE - 1

November 2021

Version 14.5.3.0.0

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 © 2021, 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.