# Oracle® Fusion Middleware

# Migrating Oracle Fusion Middleware On-Premises Database to an Autonomous Transaction Processing (Dedicated) Database

12c (12.2.1.4.0)
F40112-02
October 2021

ORACLE

Oracle Fusion Middleware Migrating Oracle Fusion Middleware On-Premises Database to an Autonomous Transaction Processing (Dedicated) Database, 12c (12.2.1.4.0)

F40112-02

# Contents

# Preface

This document describes how to migrate Oracle Fusion Middleware 12c (12.2.1.4.0) on-premises database to an Oracle Autonomous Database.



## Audience

This document is intended for users who need to migrate Oracle Fusion Middleware 12c (12.2.1.4.0) database on-premises to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database in Oracle Cloud Infrastructure.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

**Access to Oracle Support**

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

## Related Documents

You can access the Oracle Fusion Middleware documentation for additional information.

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware 12*c* Upgrade Documentation.
- For administration-related information, see Fusion Middleware 12*c* Administration Documentation.
- For release-related information, see Fusion Middleware 12*c* Release Notes.

# Conventions

The following text conventions are used in this document:

| Convention | Meaning |
|---|---|
| **boldface** | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary. |
| *italic* | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values. |
| `monospace` | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

# Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

# About Migrating 12c (12.2.1.4.0) Oracle Fusion Middleware to an Autonomous Transaction Processing-Dedicated Database

You can migrate the data from Oracle Fusion Middleware on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database in Oracle Cloud Infrastructure.

You can gain several advantages by moving to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database database.

- Why Migrate to an Autonomous Transaction Processing-Dedicated Database
  By moving your on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you can simplify your database provisioning, maintenance, and management operations.

## Why Migrate to an Autonomous Transaction Processing-Dedicated Database

By moving your on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you can simplify your database provisioning, maintenance, and management operations.

You can have your own dedicted infrastructure in the Oracle Cloud, a Private Database Cloud within the Oracle Public Cloud. You can run your cloud instance without sharing your hardware with other cloud users, and Oracle's cloud management software also can run on different hardware, further isolating it from security threats and malicious users.

Oracle Autonomous Database on Dedicated Infrastructure runs inside a hardware enforced virtual cloud network, offering the highest level of isolation from other tenants. You can easily configure one or more Container Databases on the dedicated Infrastructure, each of which can contain one or more Pluggable Databases. This setup delivers a self-driving, self-securing, self-repairing database service that can instantly scale to meet the demands of your mission-critical applications.

2

# Migrating 12c (12.2.1.4.0) Oracle Fusion Middleware On-Premises Database to an Autonomous Transaction Processing-Dedicated Database

Migrate an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for the Oracle Fusion Middleware products, Oracle WebCenter Content, Oracle WebCenter Portal, Oracle WebCenter Sites, Oracle SOA Suite, including the components, Oracle Enterprise Scheduler Services and Oracle Managed File Transfer, Oracle Forms, and Oracle Reports only.

The topics covered in this section describe the preparation steps, the migration tasks, and the postmigration tasks.

- Prepare to Migrate
  Before you begin with the migration of an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you must meet the prerequisites and perform the premigration tasks that are described in this section.
- Migration Tasks
  Compete the steps in the following sections to migrate an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:
- Postmigration Tasks
  After migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, perform the tasks described in this section. Some of these tasks apply to specific schemas.

## Prepare to Migrate

Before you begin with the migration of an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you must meet the prerequisites and perform the premigration tasks that are described in this section.

- Prerequisites to Migrate an On-Premises Database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database
  Perform the following prerequisites before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.
- Premigration Tasks
  Perform the following premigration tasks before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

# Prerequisites to Migrate an On-Premises Database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database

Perform the following prerequisites before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

> **Note:**
>
> The prerequisites for Oracle Reports is covered in *Prerequisites to Migrate an On-Premises Database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database for Oracle Reports*.

1. Install and configure Oracle Fusion Middleware version 12.2.1.4.0 domain on an on-premises database. If the Oracle Fusion Middleware version is 11g (11.1.1.9.0), perform an end-to-end upgrade to 12c (12.2.1.4.0) release version.

   > **Note:**
   >
   > In case of Oracle Forms, perform an end-to-end upgrade to 12c (12.2.1.4.0) release version, if the Oracle Fusion Middleware version is 11.1.1.7.0 or 11.1.2.2.0.

2. Install the Oracle Instant Client with SQL*Plus and impdp tools on the Oracle Cloud Infrastructure (OCI) machine.

   See Oracle Instant Client Downloads.

- Prerequisites to Migrate an On-Premises Database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database for Oracle Reports
  Perform the following prerequisites before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

# Prerequisites to Migrate an On-Premises Database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database for Oracle Reports

Perform the following prerequisites before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

> **Note:**
>
> The Oracle Cloud Infrastructure (OCI) VM should be in the same tenancy as your dedicated autonomous database.

1. Configure the OCI VM with at least 15 GB of memory (RAM) and 60 GB of disk space (total storage).

2. Install the environment group "Server with GUI" in the OCI VM to enable GUI after reboot:

```
sudo su - root
yum groupinstall "Server with GUI"
ln -sf '/lib/systemd/system/runlevel5.target' '/etc/systemd/system/
default.target'
exit
```

3. Install Tiger VNC using the following commands:

```
sudo su - root
yum install tigervnc-server
exit
```

   You must manually start the VNC server after the machine reboots.

4. Resize the system disk partition to allocate more disk space to your volume as follows:

   a. Determine the volume files system that should be increased:

```
lsblk
```

   b. Extend the root partition, and reboot the machine:

```
sudo growpart /dev/sda 3
sudo reboot
```

5. Start the VNC server manually as current user with a screen resolution as follows:

```
vncserver -geometry 1280x1024
```

> **Note:**
>
> If you are starting the VNC server for the first time, you must create a password. The resolution should not be larger than the monitor size.

6. Check the currently installed Oracle Fusion Middleware OS packages and kernel are up to date:

```
sudo yum -y update
```

7. Determine if the Oracle Fusion Middleware OS packages are installed.

   You can manually check the packages using information in Operating System Requirements in *System Requirements and Specifications*.

   If any of the packages are not installed, for Oracle Linux 7, run the following command:

```
sudo yum install <PACKAGE_NAME_WITHOUT_VERSION>*.x86_64
```

# Premigration Tasks

Perform the following premigration tasks before you migrate an on-premises database to Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

- Creating Credentials
  Create a database credential to authenticate between the Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database and the Oracle Cloud Infrastructure (OCI) object storage.
- Installing 12c (12.2.1.4.0) Product Binaries in Oracle Cloud Infrastructure VM
  Ensure that the 12c (12.2.1.4.0) product binaries are installed in the new `ORACLE_HOME`.
- Creating a Backup of the Schema Version Registry
  Use the Upgrade Assistant on the on-premises host to create a backup of the existing schema version registry on the on-premises database.

## Creating Credentials

Create a database credential to authenticate between the Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database and the Oracle Cloud Infrastructure (OCI) object storage.

To create credentials for OCI authentication:

1. Generate an authentication for your user account. For information on how to generate a token, see Generate an Authentication token.
2. Set the following environment variables on the Oracle Cloud Infrastructure (OCI) host:

   ```
   export LD_LIBRARY_PATH=/usr/lib/oracle/18.5/client64/
   lib:$LD_LIBRARY_PATH
   export PATH=/usr/lib/oracle/18.5/client64/bin:$PATH
   export TNS_ADMIN=<path_of_the_wallet_files>
   export ORACLE_SID=<Database name> (Optional)
   ```

   where, `TNS_ADMIN` is the location where you downloaded the database wallet on your OCI host and `ORACLE_SID` is the database name.
3. Go to the directory where Oracle Instant Client is installed:

   ```
   cd /usr/lib/oracle/18.5/client64/bin
   ```

4. Connect to sqlplus:

   ```
   connect admin/
   <admin_password>@<database_service_name_found_in_tnsnames.ora>
   ```

   For example:

   ```
   connect ADMIN/<admin_password>@fmwatpdedic2_tp
   ```

5. Run the following procedure. In this example, replace username and password with your own cloud credentials:

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL(
  credential_name => '<userXX_cred>',
  username =>  '<OCI_Username>',
  password =>  '<Your_Auth_Token_Here>');
END;
/
```

## Installing 12c (12.2.1.4.0) Product Binaries in Oracle Cloud Infrastructure VM

Ensure that the 12c (12.2.1.4.0) product binaries are installed in the new `ORACLE_HOME`.

Copy the 12c (12.2.1.4.0) Oracle Fusion Middleware installers from the on-premises VM to the OCI VM. To copy the installer files from on-premises to the OCI host, you can also upload the file to Oracle Cloud Infrastructure (OCI) Object Storage and download this file to the OCI VM. See Uploading Files to Object Storage in *Using Oracle Autonomous Database on Dedicated Exadata Infrastructure*.

After copying the installer files, install the products. See the Oracle Fusion Middleware library page.

## Creating a Backup of the Schema Version Registry

Use the Upgrade Assistant on the on-premises host to create a backup of the existing schema version registry on the on-premises database.

To create a backup of the schema version registry :

1. On the on-premises host, navigate to the `ORACLE_COMMON/upgrade/bin` directory and run the following command on your on-premises database:

```
./ua -backupRegistry

Oracle Fusion Middleware Upgrade Assistant 12.2.1.4.0
Enter the Database Connect String(host:port/service or host:port:SID or
TNS connect string):
myhost.us.example.com:1521/myservice.us.example.com
Enter the DBA User Name: sys as sysdba
Enter the DBA Password: <DBA Password>
```

   The schema version registry is saved to `./registry.xml`.

2. Upload the registry file to Oracle Cloud Infrastructure Object Storage. See Uploading Files to Object Storage in *Using Oracle Autonomous Database on Dedicated Exadata Infrastructure*.

   This registry file can now be downloaded from cloud object storage to the ATP-D Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database Oracle Cloud Infrastructure (OCI) VM.

# Migration Tasks

Compete the steps in the following sections to migrate an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

Exporting an On-Premises Database and Creating a Domain Template steps are performed on an on-premises host and the remaining steps are performed on the OCI host.

- Exporting an On-Premises Database to a Data Dump File
  Use Oracle Data Pump to export data from an on-premises database to your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.
- Creating a Domain Template for an On-Premises Domain
  Use the Domain Template Builder to create a custom domain template.
- Uploading the Data Dump File and Template to Oracle Cloud Infrastructure Object Storage
  Use this task to create a new storage bucket and upload the export files to it.
- Creating Users and Tablespaces
- Importing the Data Dump File to an Oracle Autonomous Transaction Processing-Dedicated Database
  Use this task to import the data for the schemas.
- Restoring the Schema Version Registry
  To migrate schema version registry from an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you must restore the schema version registry on your dedicated autonomous database.
- Creating a New Domain Using the Configuration Wizard
  Use the Configuration Wizard to create a new domain in your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database Oracle Cloud Infrastructure (OCI) VM with the Domain Template you created for an on-premises domain.
- Updating the Configuration Files in the Oracle Cloud Infrastructure Domain Host
  Follow the steps in this section to update the `config.xml` configuration file, and the Oracle Platform Security Services (OPSS) configuration files, `jps-config.xml` and `jps-config-jse.xml` files in the Oracle Cloud Infrastructure (OCI) domain host.

## Exporting an On-Premises Database to a Data Dump File

Use Oracle Data Pump to export data from an on-premises database to your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

To export data from an on-premises database to a data dump file:

1. Set the environment variables on the on-premises database host:

   ```
   setenv ORACLE_HOME <DB_$OH>
   setenv ORACLE_SID <SID>
   setenv PATH $ORACLE_HOME/bin:$PATH
   ```

   where `DB_$OH` is the Oracle_Home of the database and `SID` is the service ID of the database.

2. Start SQL*Plus:

   ```
   sqlplus
   ```

3. Connect to the database as *SYS* user with the *SYSDBA* privilege:

   ```
   CONNECT / AS SYSDBA
   ```

4. Check the schema users for your product:

   ```
   Select username,default_tablespace, temporary_tablespace from dba_users
   where username like '<Schema_Prefix>%';
   ```

   > **Note:**
   >
   > You must select the users only that are applicable to your product. Ensure that the user names have the schema prefixes that you specified when creating the schemas.

5. Unlock the schemas for the schema users and commit the changes:

   ```
   ALTER USER <Schema_Prefix_User> IDENTIFIED BY <Schema_Password> account
   unlock;
   commit;
   ```

   You can use the results from step 4 to determine the schemas to be unlocked.

   An example command to unlock the schemas:

   ```
   ALTER USER upg_stb identified by <Schema_Password> account unlock;
   ```

6. Create a new directory on the database server:

   For example, create a directory `/scratch/DP/soa`:

   ```
   CREATE OR REPLACE DIRECTORY test_dir AS '/scratch/DP/soa';
   ```

   > **Note:**
   >
   > Ensure that this folder structure exists on the on-premises database host.

7. Grant access to all users you identified in step 4 to the database directory:

   ```
   GRANT read, write ON DIRECTORY test_dir TO <Schema_Prefix_User>;
   ```

For example, to grant user access to the directory for *SOAINFRA*, *ORASDPM*, *IAU*, *MDS*, *OPSS*, *STB*, and *WLS* schemas:

```
GRANT read, write ON DIRECTORY test_dir to upg_soainfra,
upg_orasdpm, upg_iau, upg_mds, upg_opss, upg_stb,
upg_wls;
```

8. Exit SQL using the `exit` command.
9. Export all schemas at once using the `expdp` command.

   For example:

   ```
   expdp \"sys@<DB_SID> as sysdba\" dumpfile=soa_infra.dmp
   logfile=product.log directory=test_dir
   schemas = upg_orasdpm,upg_iau,upg_mds,upg_opss,upg_stb,upg_wls
   ```

   where `DB_SID` is the service ID of the on-premises database.

   > **Note:**
   >
   > If the schema size is large, you can run the `expdp` command to export each schema individually.

   For Oracle Enterprise Scheduler (ESS), to export schemas and PROCOBJ data, you must also run this `expdp` command:

   ```
   expdp \"sys/<DB_SID> as sysdba\" dumpfile=<dumpfilename>.dmp
   logfile=product.log full=y INCLUDE=PROCOBJ\:\"LIKE
   \'<Schema_Prefix>\%\'\" INCLUDE=GRANT INCLUDE=ROLE_GRANT
   directory=test_dir
   ```

```
#Set the environment variables and connect to the database as SYS user
using sqlplus,
and then select the users that are applicable to Oracle Forms product.
Refer steps 1 to 4.

ALTER USER abc identified by <Schema_Password> account unlock;
GRANT read,write on DIRECTORY test_dir to abc;
commit;
# Export the schemas using expdp
expdp system/<SYS_PWD>@<DB_SID> schemas=abc directory=test_dir
dumpfile=abc_meta.dmp logfile=abc1.log
```

## Creating a Domain Template for an On-Premises Domain

Use the Domain Template Builder to create a custom domain template.

To create a domain template for an on-premises domain using the Domain Template Builder:

1. Start the Domain Template Builder on the on-premises domain host:

```
ORACLE_HOME/oracle_common/common/bin/config_builder.sh
```

2. On the Create Domain Template screen:
   a. Select **Create Domain Template**, and then **select Use Domain as a Source**.
   b. In the **Source Location** field, specify the location of the source domain directory, or browse to select the location of an existing domain directory, from which you want to create a domain template.
   c. In the **Template Location** field, specify the JAR file name and the location, or browse to select an existing domain directory in which the template is located, to create the new template.
   d. Click **Next**.
3. On the Template Information screen, review the information, and click **Next**.
4. On the Template Summary screen, review the information, and click **Next**.
5. The Configuration Progress screen displays the progress of template creation. After the configuration process is complete, click **Next**, and then click **Finish** to end the configuration.
6. Copy this domain template you created to the Oracle Cloud Infrastructure (OCI) domain host.

# Uploading the Data Dump File and Template to Oracle Cloud Infrastructure Object Storage

Use this task to create a new storage bucket and upload the export files to it.

To upload the data dump file and template to Oracle Cloud Infrastructure (OCI) Object Storage:

1. Open a supported browser, and sign in to the Oracle Cloud Infrastructure Console.
2. Click the navigation menu  , then under **Object Storage**, click **Object Storage**.
3. In the **Compartment** drop-down list, under the root tenancy, select the compartment in which to upload the dump file and template.
4. Click **Create Bucket** and enter the bucket name.

   For example, in this case, enter `wcc_ins_db_migration`.
5. Keep the default encryption options and click **Create**.
6. Click the bucket name that you created, in this case, `wcc_ins_db_migration`.
7. Click **Upload**, and choose the dump file, <*`dumpfilename`*>`.dmp`, downloaded in the `test_dir` folder in your on-premises machine, and upload the file.
8. In the list of objects, locate the uploaded dump file, click  , and then click **View Object Details** to copy the **URL Path**.
9. Repeat step 8 and step 9 to upload the domain template you created for an on-premises domain to the bucket, `wcc_ins_db_migration`, where you uploaded the dump file. See Creating a Domain Template for an On-Premises Domain.

# Creating Users and Tablespaces

To create users and tablespaces using Oracle Instant Client:

1. Set the following environment variables in the OCI VM:

   ```
   export LD_LIBRARY_PATH=/usr/lib/oracle/18.5/client64/
   lib:$LD_LIBRARY_PATH
   export PATH=/usr/lib/oracle/18.5/client64/bin:$PATH
   export TNS_ADMIN=/home/opc/idm
   export ORACLE_SID=FMWATPDedic2 (Optional)
   ```

   where, `TNS_ADMIN` is the location where you downloaded the database wallet on your OCI host and and `ORACLE_SID` is the Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database name.

2. Go to the directory where Oracle Instant Client is installed:

   ```
   cd /usr/lib/oracle/18.5/client64/bin
   ```

3. Connect to sqlplus:

   ```
   connect admin/
   <admin_password>@<database_service_name_found_in_tnsnames.ora>
   ```

   For example:

   ```
   connect ADMIN/<admin_password>@fmwatpdedic2_tp
   ```

4. Create the tablespaces on your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database

   You can use the results from step 4 in Exporting an On-Premises Database to a Data Dump File to determine the tablespaces to be created.

   An example command to create tablespaces:

   ```
   CREATE TABLESPACE "MIG2_IAS_WEBCENTER"
   CREATE TABLESPACE "MIG2_IAS_PORTLET"
   ```

5. Create the users on your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

   > **Note:**
   >
   > The users should be created based on the product.
   > You can use the results from step 4 in Exporting an On-Premises Database to a Data Dump File to determine the users to be created.

An example command to create users:

```
Create user MIG2_WEBCENTER identified by <user_password> DEFAULT
TABLESPACE MIG2_IAS_WEBCENTER TEMPORARY TABLESPACE MIG2_IAS_TEMP ;
Create user MIG2_PORTLET identified by <user_password>  DEFAULT
TABLESPACE MIG2_IAS_PORTLET TEMPORARY TABLESPACE MIG2_IAS_TEMP;
```

6. Grant unlimited amount of disk space in the tablespaces to all users:

   You can use the results from step 4 to determine the users to grant unlimited disk space in the tablesapces.

   For example, to grant unlimited tablespace to `MIG2_WEBCENTER` user:

   ```
   (Optional) ALTER USER MIG2_WEBCENTER IDENTIFIED BY <password>;
   GRANT UNLIMITED TABLESPACE to MIG2_WEBCENTER;
   ```

7. Grant privileges that allows all users to create objects in the schema:

   ```
   GRANT CONNECT, create view, create table, create procedure, create
   trigger, create synonym, create sequence, create type to <user_name>;
   ```

   For example, to grant privileges to `WCS_WLS_RUNTIME` user:

   ```
   GRANT CONNECT, create view, create table, create procedure, create
   trigger, create synonym, create sequence, create type to WCS_WLS_RUNTIME;
   ```

   For Oracle Enterprise Scheduler (ESS) users, you must also grant the following privileges:

   ```
   grant execute on DBMS_LOCK to <Schema_Prefix_User>;
   grant execute on UTL_FILE to <Schema_Prefix_User>;
   grant execute on UTL_RAW to <Schema_Prefix_User>;
   grant execute on DBMS_LOB to <Schema_Prefix_User>;
   grant execute on DBMS_SCHEDULER to <Schema_Prefix_User>;
   grant execute on DBMS_XMLDOM to <Schema_Prefix_User>;
   grant execute on DBMS_APPLICATION_INFO to <Schema_Prefix_User>;
   grant execute on DBMS_UTILITY to <Schema_Prefix_User>;
   grant execute on DBMS_SESSION to <Schema_Prefix_User>;
   grant execute on DBMS_OUTPUT to <Schema_Prefix_User>;
   grant execute on SYS.DBMS_ASSERT to <Schema_Prefix_User>;

   grant select on sys.v_$instance to <Schema_Prefix_User>;
   grant select on sys.gv_$instance to <Schema_Prefix_User>;
   grant select on sys.v_$session to <Schema_Prefix_User>;
   grant select on sys.gv_$session to <Schema_Prefix_User>;
   grant select on sys.v_$parameter to <Schema_Prefix_User>;

   grant create any job to <Schema_Prefix_User>;
   grant create job to <Schema_Prefix_User>;
   grant manage scheduler to <Schema_Prefix_User>;

   grant select on dba_scheduler_jobs to <Schema_Prefix_User>;
   grant select on dba_scheduler_job_run_details to <Schema_Prefix_User>;
   ```

```
grant select on dba_scheduler_running_jobs to <Schema_Prefix_User>;
grant select on dba_scheduler_job_classes to <Schema_Prefix_User>;
```

> **Note:**
>
> Ensure that you create the same ESS user that was created in the on-premises database and execute all the required grants to the user, postmigration to the ATP-D database.

## Importing the Data Dump File to an Oracle Autonomous Transaction Processing-Dedicated Database

Use this task to import the data for the schemas.

To import the data dump file to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database database using data dump:

1. Set the following environment variables on the Oracle Cloud Infrastructure (OCI) host:

   ```
   export LD_LIBRARY_PATH=/usr/lib/oracle/18.5/client64/
   lib:$LD_LIBRARY_PATH
   export PATH=/usr/lib/oracle/18.5/client64/bin:$PATH
   export TNS_ADMIN=<path_of_the_wallet_files>
   export ORACLE_SID=<Database_name> (Optional)
   ```

   where, `TNS_ADMIN` is the location where you downloaded the database wallet on your OCI host and `ORACLE_SID` is the database name.

2. Go to the directory where Oracle Instant Client is installed:

   ```
   cd /usr/lib/oracle/18.5/client64/bin
   ```

3. (Optional) Connect to sqlplus:

   > **Note:**
   >
   > If `TNS_ADMIN` is already setup and you are able to connect to the database, it is not mandatory to run this command.

   ```
   connect admin/
   <admin_password>@<database_service_name_found_in_tnsnames.ora>
   ```

   For example:

   ```
   connect ADMIN/<admin_password>@fmwatpdedic2_tp
   ```

4. (Optional) Exit SQL using the `exit` command.

> **Note:**
>
> If `TNS_ADMIN` is already setup and you are able to connect to the database, it is not mandatory to run this command.

5. Run the following command to import the data dump file to an ATP-D database for your schemas:

```
impdp admin/<password_of_admin_user_for_ATP-
D_host>@<database_service_name_found_in_tnsnames.ora>
credential=<credential_name>
dumpfile=<schema_export_dump_file_cloud_object_storage_location
exclude=user TABLE_EXISTS_ACTION=REPLACE
```

Example:

```
impdp admin/<password_of_admin_user_for_ATP-D_host>@fmwatpdedic2_tp
credential=def_cred_name dumpfile=https://objectstorage.us-
ashburn-1.oraclecloud.com/n/atpdpreview2/b/wcc_install_mig1/o/ocs.dmp
exclude=user TABLE_EXISTS_ACTION=REPLACE
```

For Oracle Enterprise Scheduler (ESS), when you import the data dump file to an ATP-D database, you must also run this `impdp` command:

```
impdp admin/<password_of_admin_user_for_ATP-
D_host>@<database_service_name_found_in_tnsnames.ora>
credential=<credential_name>
dumpfile=<PROCOBJ_schema_export_dump_file_cloud_object_storage_location>
TABLE_EXISTS_ACTION=REPLACE
```

6. Run the following procedure to move the `.log` and `.sql` files to the Object Storage:

```
BEGIN
  DBMS_CLOUD.PUT_OBJECT(
  credential_name => '<userXX_cred>',
  object_uri =>  '<the_storage_bucket_URL>/import.log',
  directory_name =>  '<data_dump_dir>',
  file_name =>  'import.log');
END;
/
```

7. Verify if there are any errors in the `import.log` file.

```
#Set the environment variables and navigate to the directory where Oracle
Instant Client is installed, and then log into the database as an admin user
using sqlplus. Refer steps 1
to 3.

CREATE TABLESPACE "USERS";
Create user abc identified by <Schema_Password> DEFAULT TABLESPACE USERS
TEMPORARY TABLESPACE TEMP;
GRANT UNLIMITED TABLESPACE to abc;
```

```
GRANT CONNECT, create view, create table, create procedure, create
trigger, create synonym, create sequence, create type to abc;
commit;
#connect as abc user
connect abc/<password>@fmwatpdedic2_tp;
CREATE TABLE DEPT("DEPTNO" NUMBER(2,0),"DNAME" CHAR(14 BYTE),"LOC"
CHAR(13 BYTE));
CREATE TABLE EMP("EMPNO" NUMBER(4,0),"ENAME" CHAR(10 BYTE), "JOB"
CHAR(9 BYTE), "MGR" NUMBER(4,0),"HIREDATE" DATE,"SAL"
NUMBER(7,2),"COMM" NUMBER(7,2), "DEPTNO" NUMBER(2,0));
commit;
# Drop the current table in the database and recreate the new table as
in the dump file using impdp
impdp admin/<admin_password>@fmwatpdedic2_tp credential=DEF_CRED_NAME /
dumpfile=https://objectstorage.us-ashburn-1.oraclecloud.com/n/
atpdpreview2/b/FormsInstallDBMigration/o/abc_meta.dmp /
TABLE_EXISTS_ACTION=REPLACE
```

## Restoring the Schema Version Registry

To migrate schema version registry from an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, you must restore the schema version registry on your dedicated autonomous database.

Before you restore the schema version registry:

- You must download or copy the registry from an on-premises host to the OCI host.
- You must have applied OPatch for `ua restoreRegistry`. See 32089134 to apply the OPatch.

> **Note:**
>
> If you do not apply OPatch, you might run into issues while creating the domain.

You can restore the schema version registry only for the schemas that are imported to the ATP-D database.

To restore the schema version registry to your dedicated autonomous database (ATP-D):

1. Install the 12c (12.2.1.4.0) Oracle Fusion Middleware product distribution on the Oracle Cloud Infrastructure (OCI) VM.
2. Copy the `registry.xml` you created in Creating a Backup of the Schema Version Registry to `oracle_common/upgrade/bin` directory.
3. Navigate to the `oracle_common/upgrade/bin` directory.
4. Run the following command:

```
./ua -restoreRegistry

Oracle Fusion Middleware Upgrade Assistant 12.2.1.4.0
```

```
Enter location of Schema Version Registry backup file:
/home/opc/wcc/12215/oracle_common/upgrade/bin/registry.xml
Restoring from /home/opc/wcc/12215/oracle_common/upgrade/bin/registry.xml
Enter prefix or * for list:
<Schema_Prefix>
Enter the Database Connect String:
(host:port/service or host:port:SID or TNS connect string)
jdbc::oracle:thin:@<TNS_alias>?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
# Example of Database Connect String: jdbc:oracle:thin:@fmwatpdedic2_tp?
TNS_ADMIN=/home/opc
Enter the DBA User Name:
<user_name>
Enter the DBA Password:
<password>
Schema Version Registry restored from /home/opc/wcc/12215/oracle_common/
upgrade/bin/registry.xml
Rows removed: 0. Rows inserted: 12
```

> **Note:**
>
> You must enter the complete location of the `registry.xml` file and pass the complete database connect string.

## Creating a New Domain Using the Configuration Wizard

Use the Configuration Wizard to create a new domain in your Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database Oracle Cloud Infrastructure (OCI) VM with the Domain Template you created for an on-premises domain.

> **Note:**
>
> During migration, the source and target `process group` name of the domain must be the same, else, you cannot view the imported source details for the OCI VM, if you need to look for this information from the OCI VM.

To create a domain template, see Creating a Domain Template for an On-Premises Domain.

You must download or copy the domain template file from an on-premises host to the OCI host before you create a domain.

To create a domain using the Configuration Wizard:

1. Start the Configuration Wizard on the OCI domain host:

   ```
   ORACLE_HOME/oracle_common/common/bin/config.sh
   ```

2. On the Create Domain screen, select **Create a new domain**, and in the **Domain Location** field, specify the location of the domain, or browse to select an existing directory in which your domain is located, and then click **Next**.

3. On the Templates screen, select **Create Domains Using Custom Templates**, and in the **Template Location** field, browse to select the directory in which you copied or downloaded the domain template, and then click **Next**.
4. On the High Availability screen, click **Next**.
5. On the Application Location screen, in the **Application Location** field, specify the location of the directory or browse to select an existing directory, in which you want to store the applications that are associated with the domain, and then click **Next**.
6. On the Administrator Account page, specify the **Name** and **Password**, reconfirm the password, and then click **Next**.
7. On the Domain Mode and JDK screen, select a **Domain Mode** and select the **JDK** for the domain, and then click **Next**.
8. On the JDBC Data Sources screen, select the MDS schema, specify **Name** and **Password**, and the connect string using the **Connection URL String** option, and then click **Next**.

   Use the following format for the connect string:

   `jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`

   In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

   > **Note:**
   >
   > This step is applicable only for Oracle WebCenter Content.

9. On the JDBC Data Sources Test screen, select the data source, `mds-WCCUIMDSREPO` and click **Test Selected Connections**. Ensure that the test is successful, and then click **Next**.

   > **Note:**
   >
   > This step is applicable only for Oracle WebCenter Content.

10. On the Database Configuration Type screen, specify **Schema Owner**, **Schema Password** and the connect string using the **Connection URL String** option.

    Use the following format for the connect string:

    `jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`

    In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

11. On the JDBC Component Schema screen, specify **Schema Owner**, **Schema Password** and the connect string for each of the data sources, and click **Next**.

    Use the following format for the connect string:

`jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

12. On the JDBC Component Schema Test screen, select all the JDBC component schemas, click **Test Selected Connections**. Ensure that the test is successful, and then click **Next**.
13. On the Advanced Configuration screen, select **Administration Server**, **Node Manager**, **Topology**, and then click **Next**.
14. On the Administration Server screen, select a value for the **Listen Address**, and click **Next**.
15. On the Node Manager screen, review the information, and click **Next**.
16. On the Managed Servers screen, select a value for the **Listen Address** for each managed server and click **Next**.
17. Continue to click **Next** until you reach the Machines screen.
18. On the Machines screen, select a value for the **Node Manager Listen Address** and click **Next**.
19. Continue to click **Next** until you reach the Configuration Summary screen.

    In case of Oracle Forms, the Oracle Forms application should be deployed on Forms Managed Servers only.
20. On the Configuration Summary page, click **Create**.
21. The Configuration Progress screen displays the progress of domain creation. After the configuration process is complete, click **Next**, and then click **Finish** to end the configuration.

## Updating the Configuration Files in the Oracle Cloud Infrastructure Domain Host

Follow the steps in this section to update the `config.xml` configuration file, and the Oracle Platform Security Services (OPSS) configuration files, `jps-config.xml` and `jps-config-jse.xml` files in the Oracle Cloud Infrastructure (OCI) domain host.

1. To update the `config.xml` file:
   a. Navigate to the directory, `DOMAINHOME/config` on the OCI domain host.
   b. If the RDBMS security store is enabled in the on-premises domain, update the `config.xml` file in one of the following ways:
      - In the WebLogic Server Administration Console, go to **Security Realms**, select the *RealmName*, click **RDBMS Security Store**, and update the RDBMS connection configuration.
      - Use the WebLogic Scripting Tool (WLST).

> **Note:**
>
> For `sec:connection-url`, update `jdbc:oracle:thin:@@//dbserver:listener_port/DB_ServiceName` with the new database location, `jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`.

Example command to update the `config.xml` file:

```
store = realm.getRDBMSSecurityStore()
store.setUsername('<Db_SchemaUser>')
store.setPassword('<Db_SchemaPassword>')
store.setConnectionURL('jdbc:oracle:thin:@TNS_alias?
TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and
tnsnames.ora>')
store.setDriverName('<driverName>')
```

Example of `config.xml` after update:

```
<sec:rdbms-security-store>
    <sec:username><Db_SchemaUser></sec:username>
    <sec:password-encrypted><Db_SchemaPassword></
sec:password-encrypted>
    <sec:connection-url>jdbc:oracle:thin:@TNS_alias?
TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and
tnsnames.ora>
    <sec:driver-name><driverName></sec:driver-name>
</sec:rdbms-security-store>
```

2. To update the `jps-config.xml` and `jps-config-jse.xml` files:
   a. Navigate to the directory, `DOMAINHOME/config/fmwconfig` on the OCI domain host.
   b. In the `jps-config.xml` file, for the `jdbc.url` property, update the connect string with the new database location, `jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`.

      For example, update:

      ```
      <property name="jdbc.url" value="jdbc:oracle:thin:@//
      dbserver:listener_port/DB_ServiceName"/>
      ```

      To:

      ```
      <property name="jdbc.url" value="jdbc:oracle:thin:@TNS_alias?
      TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and
      tnsnames.ora>"/>
      ```

   c. In the `jps-config-jse.xml`, for the `jdbc.url` and the `audit.loader.jdbc.string` property, update the connect string with the new

database location, `jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and tnsnames.ora>`.

For example, update:

```
<property name="jdbc.url" value="jdbc:oracle:thin:@//
dbserver:listener_port/DB_ServiceName"/>
<property name="audit.loader.jdbc.string" value="jdbc:oracle:thin:@//
dbserver:listener_port/DB_ServiceName"/>
```

To:

```
<property name="jdbc.url" value="jdbc:oracle:thin:@TNS_alias?
TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and
tnsnames.ora>"/>
<property name="audit.loader.jdbc.string"
value="jdbc:oracle:thin:@TNS_alias?
TNS_ADMIN=<path_of_the_wallet_files, ojdbc.properties, and
tnsnames.ora>"/>
```

**Note:**

During migration, if you have changed the OPSS password in your dedicated autonomous database (ATP-D), execute the following WebLogic Scripting Tool (WLST) commands in offline mode:

```
cd <Domain_Home>/oracle_common/common/bin
./wlst.sh

Initializing WebLogic Scripting Tool (WLST) ...Jython scans all
the jar files it can find at first startup.
Depending on the system, this process may take a few minutes to
complete, and WLST may not return a prompt right away.
Welcome to WebLogic Server Administration Scripting Shell Type
help() for help on available commands

modifyBootStrapCredential(jpsConfigFile="<Domain_Home>/config/
fmwconfig/jps-config-jse.xml",
username="<Prefix>_OPSS", password="<New_Password>")
```

# Postmigration Tasks

After migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, perform the tasks described in this section. Some of these tasks apply to specific schemas.

If you run into any issues during post-migration, see Errors Postmigration to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database.

- Postmigration Tasks for Oracle WebCenter Sites
  Perform the following steps after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Sites:
- Postmigration Tasks for Oracle Enterprise Content Management Suite
  Perform the following steps after migrating on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle Enterprise Content Management Suite (ECM):
- Postmigration Tasks for Oracle WebCenter Content
  After migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Content, perform the steps described in Migrating Oracle WebCenter Content.
- Postmigration Tasks for Oracle WebCenter Portal
  Perform the following steps after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Portal:
- Postmigration Tasks for Oracle Forms and Oracle Reports
  Perform the following steps after migrating on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle Forms and Oracle Reports.
- Importing Users Using the WebLogic Console
  Migrate the WebLogic users manually from an on-premises VM to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) VM.
- Starting the Servers
  After the migration of your on-premises to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, restart all processes and servers, including the Administration Server and any Managed Servers.
- Performing Sanity Check
  After the migration of your on-premises to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, verify the application URLs, and ensure that data is accessible from the application.

# Postmigration Tasks for Oracle WebCenter Sites

Perform the following steps after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Sites:

1. Copy the contents of `{DOMAIN_HOME}/wcsites` directory from on-premises to the same location in the Oracle Cloud Infrastructure (OCI) VM.
2. Locate the following files to be updated in the `wcsites` folder. Open these files in a text editor, and update the on-premises hostname and the domain home entries with the OCI VM name and the domain home in the ATP-D VM respectively:

> **Note:**
>
> You can use the command, `grep -rnw . -e 'mystring'` to search files containing the string: *mystring*.

```
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/bootstrap/
search/SearchEngineMetaDataConfig.html${MW_HOME}/user_projects/domains/$
{DOMAIN_NAME}/wcsites/bin/grant-opss-permission.py
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/satelliteserver/
config/wcs_satelliteserver_properties_bootstrap.ini
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/satelliteserver/
config/wcs_properties.json
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/satelliteserver/
config/SSOConfig.xml
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/visitorservices/
config/visitors.properties
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/sitecapture/config/
wcs_properties.json
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/sitecapture/config/
spring/root-context.xml
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/sitecapture/config/
wcs_sitecapture_properties_bootstrap.ini
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/Shared/
bootstrap/fsii/webroot/WebRoot.html
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
host.properties
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
cas.properties
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
jbossTicketCacheReplicationConfig.xml
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
wcs_properties.json
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
deployerConfigContext.xml
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
wcs_properties_bootstrap.ini
{MW_HOME}/user_projects/domains/${DOMAIN_NAME}/wcsites/wcsites/config/
customBeans.xml
```

3. Set the number of open files limit using the following commands:
   a. Open the file `/etc/security/limits.conf` in vi editor.
   b. Add the following lines at the end of the file:

```
*        hard     nofile     500000
*        soft     nofile     500000
root     hard     nofile     500000
root     soft     nofile     500000
```

4. In the `wcs_properties.json` file, set the value for the properties `cc.security` and `cs.session` to *true*.

5. Update all files with on-premises datasource connection strings to ATP-D connection string. See Connection Credentials for ATP-D Database.
6. Run the following SQL command to change the current schema of the session:

   ```
   ALTER SESSION SET CURRENT_SCHEMA = <Schema_Prefix>_WCSITES;
   ```

7. Run the following SQL commands to update the tables:

   ```
   UPDATE SYSTEMSATELLITE SET HOST=REPLACE(host,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE WEBROOT SET ROOTURL=REPLACE(rooturl,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_VIEW SET SRCURL=REPLACE(srcurl,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_APPLICATION SET ICONURL=REPLACE(iconurl,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_APPLICATION SET LAYOUTURL=REPLACE(layouturl,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_APPLICATION SET ICONURLCLICK=REPLACE(iconurlclick,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_APPLICATION SET ICONURLHOVER=REPLACE(iconurlhover,'<on-
   premises_hostname>','<oci_hostname>');
   UPDATE FW_APPLICATION SET ICONURLACTIVE=REPLACE(iconurlactive,'<on-
   premises_hostname>','<oci_hostname>');
   ```

8. In `SearchEngineMetaDataConfig` table, update the `WORKINGFOLDER` column.

   For example:

   ```
   UPDATE <Schema_Prefix>_WCSITES.SearchEngineMetaDataConfig set
   workingfolder='<location_of_lucene_in_oci>';
   ```

9. In `SystemInfo` table, update the location of `shared` folder.

   For example:

   ```
   UPDATE <Schema_Prefix>_WCSITES.SystemInfo set defdir =
   replace(defdir, '<location_of_shared_directory_in_on-premises>',
   '<location_of_shared_directory_in_oci>');
   ```

10. In `SystemInfo` table, update the domain location.

    For example:

    ```
    UPDATE <Schema_Prefix>_WCSITES.SystemInfo set defdir =
    replace(defdir, '<location_of_domain_in_on-premises>',
    '<location_of_domain_in_oci>');
    ```

# Postmigration Tasks for Oracle Enterprise Content Management Suite

Perform the following steps after migrating on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle Enterprise Content Management Suite (ECM):

> **Note:**
>
> These steps are applicable only to Oracle Universal Content Management (UCM), Oracle Inbound Refinery (IBR), Oracle Universal Records Management (URM), and Oracle Imaging and Process Management (IPM).

1. Follow steps in Migrate Non-Clustered WebCenter Content 12c to a Dissimilar Infrastructure from, *Copy WebCenter Content Directory to the New Host* to *Verify that Everything Works* in *Migrating Oracle WebCenter Content*.
2. If the IPM server does not start after you log in to the OCI VM, run the following command to create an Imaging connection:

   ```
   UPDATE <Schema_Prefix>_IPM.connection_details set
   detailvalue='<oci_vmhost>:4444' where detailkey='repository.secondaries';
   ```

3. Import users from the WebLogic console to the Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database OCI VM. See Importing Users Using the WebLogic Console.

# Postmigration Tasks for Oracle WebCenter Content

After migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Content, perform the steps described in Migrating Oracle WebCenter Content.

For more information, see Migrate Non-Clustered WebCenter Content 12c to a Dissimilar Infrastructure in *Migrating Oracle WebCenter Content*.

To configure full-text search, you must rebuild indexes using the Repository Manager. For more information, see Enable Full-Text Search in *Administering Oracle WebCenter Content*.

# Postmigration Tasks for Oracle WebCenter Portal

Perform the following steps after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle WebCenter Portal:

1. Run the following WLST command to update the host and port for the WebCenter Content Server connection:

   ```
   setContentServerConnection (appName='webcenter',
   name='<existing_connection_name>',
   serverHost='<new_host_name>', serverPort='<new_port_number>',
   isPrimary='true')
   ```

2. Run the following WLST command to update the host name for the discussion server connection.

```
setDiscussionForumConnection(appName='webcenter',
name='<existing_connection_name>',
url='<new_host_url>', default=1)
```

3. Restart the managed servers, `WC_Spaces` and `WC_Collaboration`.

## Postmigration Tasks for Oracle Forms and Oracle Reports

Perform the following steps after migrating on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database for Oracle Forms and Oracle Reports.

- For Oracle Forms, take a backup of your `Domain_Home`, and create a configuration file for postmigration of Oracle Forms.

> **Note:**
>
> Ensure that you have execute permissions for the `Postmigrationformsconfig.sh` file.

```
./Postmigrationformsconfig.sh
#!/bin/sh
#
OLD_DOMAIN_HOME=<ON_PREMISE_DOMAIN_HOME>
NEW_DOMAIN_HOME=<OCI_DOMAIN_HOME>

OLD_ORACLE_HOME=<ON_PREMISE_ORACLE_HOME>
NEW_ORACLE_HOME=<OCI_ORACLE_HOME>

DEFAULT_ENV=$NEW_DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/
applications/formsapp_12.2.1/config/default.env
BACKUP_DEFAULT_ENV=$NEW_DOMAIN_HOME/config/fmwconfig/servers/
WLS_FORMS/applications/formsapp_12.2.1/config/default.env.pre_script

FORMS_WEB_CFG=$NEW_DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/
applications/formsapp_12.2.1/config/formsweb.cfg
BACKUP_FORMS_WEB_CFG=$NEW_DOMAIN_HOME/config/fmwconfig/servers/
WLS_FORMS/applications/formsapp_12.2.1/config/
formsweb.cfg.pre_script

cp -rpf $DEFAULT_ENV $BACKUP_DEFAULT_ENV
cp -rpf $FORMS_WEB_CFG $BACKUP_FORMS_WEB_CFG

sed -i 's#'"$OLD_DOMAIN_HOME"'#'"$NEW_DOMAIN_HOME"'#g' $DEFAULT_ENV
sed -i 's#'"$OLD_ORACLE_HOME"'#'"$NEW_ORACLE_HOME"'#g' $DEFAULT_ENV

sed -i
's#'"$OLD_DOMAIN_HOME"'#'"$NEW_DOMAIN_HOME"'#g' $FORMS_WEB_CFG
```

```
sed -i 's#'"$OLD_ORACLE_HOME"'#'"$NEW_ORACLE_HOME"'#g' $FORMS_WEB_CFG
```

- For Oracle Reports, update the `tnsnames.ora` file located in `DOMAIN_HOME/config/fmwconfig` with the ATP-D database details.

  For example:

  ```
  -connectString description=(CONNECT_TIMEOUT=120)(RETRY_COUNT=20)
  (RETRY_DELAY=3) \
  (TRANSPORT_CONNECT_TIMEOUT=3)(ADDRESS_LIST=(LOAD_BALANCE=on)
  (ADDRESS=(PROTOCOL=protocol_name) \
  (HOST=host_name)(PORT=port_number)))
  (CONNECT_DATA=(SERVICE_NAME=service_name)))
  ```

## Importing Users Using the WebLogic Console

Migrate the WebLogic users manually from an on-premises VM to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) VM.

To import users from an on-premises to ATP-D VM using the WebLogic Console:

1. Open the Oracle WebLogic Server Administration Console in your on-premises environment, and login with your credentials.
2. Navigate to **Security Realms** > **myrealm**
3. On the **Migration** tab, select **Export**.
4. Enter the export directory location in which to export the users.

   The directory must exist on the server in your on-premises environment. For example, let's enter the directory location as `/scratch/users_in_wls`.
5. Create a tar file of `/scratch/users_in_wls` in your on-premises environment, and upload the file to Oracle Cloud Infrastructure Object Storage.

   See Uploading Files to Object Storage in *Using Oracle Autonomous Database on Dedicated Exadata Infrastructure*.
6. Copy the tar file you created in 5 to a directory in ATP-D VM.

   For example, lets copy the tar file to the `home/opc/users directory`.
7. Open the Oracle WebLogic Server Administration Console in your ATP-D VM.
8. Navigate to **Security Realms** > **myrealm**
9. On the **Migration** tab, select **Import**.
10. Enter the import directory location in which to import the users.

    For example, let's enter the directory location as `home/opc/users`.
11. Save the changes.

## Starting the Servers

After the migration of your on-premises to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, restart all processes and servers, including the Administration Server and any Managed Servers.

See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware*.

## Performing Sanity Check

After the migration of your on-premises to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database, verify the application URLs, and ensure that data is accessible from the application.

3

# Troubleshooting an On-Premises Database to an Autonomous Transaction Processing-Dedicated Database Migration

Learn to troubleshoot any issues you might encounter as part of the migration process.

- Metadata Error When Exporting Schemas
  You might receive a metadata error when you are exporting schemas from an on-premises database using the `expdp` command..
- Errors Postmigration to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database
  Troubleshoot the errors you receive after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

## Metadata Error When Exporting Schemas

You might receive a metadata error when you are exporting schemas from an on-premises database using the `expdp` command..

To fix the issue, perform the following steps:

1. Set execute permissions on `xsl`:

   ```
   chmod 755 <DB $OH>/rdbms/xml/xsl
   ```

2. Reload the stylesheets using the `dbms_metadata_util.load_stylesheets` procedure in SQL.

   ```
   SQL > execute dbms_metadata_util.load_stylesheets
   ```

3. Restart the database and execute the `expdp` command again.

## Errors Postmigration to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) Database

Troubleshoot the errors you receive after migrating an on-premises database to an Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database.

**Error Migrating Schemas**

After you migrate an on-premises database to an Autonomous Database for the Oracle Fusion Middleware products, during the migration of JRF schemas, you might receive the following error:`ORA-00001: unique constraint (<Schema Prefix>_WLS.SYS_C00522571) violated`.

This error is displayed for the following tables:

- CHECKPOINTDATA;
- EXECUTIONINSTANCEDATA;
- JOBINSTANCEDATA;
- JOBSTATUS;
- STEPEXECUTIONINSTANCEDATA;
- STEPSTATUS;
- WEBLOGIC_TIMERS;
- WL_SERVLET_SESSIONS;
- WLS_EVENTS;
- WLS_HVST;

So, to migrate the schemas, use the following commands in SQL*Plus to truncate data from the WebLogic Server database tables:

```
delete from ACTIVE;
delete from CHECKPOINTDATA;
delete from EXECUTIONINSTANCEDATA;
delete from JOBINSTANCEDATA;
delete from JOBSTATUS;
delete from STEPEXECUTIONINSTANCEDATA;
delete from STEPSTATUS;
delete from WEBLOGIC_TIMERS;
delete from WL_SERVLET_SESSIONS;
delete from WLS_EVENTS;
delete from WLS_HVST;
```

### Error While Running the Reports Builder

After you migrate an on-premises database to an Autonomous Database, for Oracle Reports, you receive the following error: `Error: REP-0004 Text: Warning - "Unable to open user preference file". while running reports`.

This warning is displayed if the Oracle Reports executable file is not found in the specified location; the process will continue even if this warning is displayed.

Copy the `prefs.ora` file from your Reports Builder directory, `ORACLE_HOME/tools/admin/` directory to the Applications directory, `HOME`.

### Bind to Reports Server Failed

After you migrate an on-premises database to an Autonomous Database, for Oracle Reports, you receive the following error: `REP-51002: Bind to Reports Server rep_server failed`.

If you receive this error, you must enable naming service discovery mechanism for all Reports servers in Oracle Reports 12c.

> **Note:**
>
> Ensure that the WLS_REPORTS managed server and Reports server are down.

1. Create a backup of the `rwnetwork.conf` files, and then edit the `rwnetwork.conf` files.
   - `$DOMAIN_HOME/config/fmwconfig/components/ReportsToolsComponent/<ReportsToolsInstance>/rwnetwork.conf` (Tools)
   - `$DOMAIN_HOME/config/fmwconfig/components/ReportsServerComponent/<standalone repserver>/rwnetwork.conf` (Standalone Reports Server)
   - `$DOMAIN_HOME/config/fmwconfig/servers/WLS_REPORTS/applications/reports_12.2.1/configuration/rwnetwork.conf` (In-process Reports Server)
2. Enable naming service (COS) discovery mechanism for all the updated `rwnetwork.conf` files.

   For example:

   ```
   #From
   <multicast channel="228.5.6.7" port="14021" timeout="1000"/>
   <!--namingService name="Cos" host="%NAMING_HOST%" port="%NAMING_PORT%"/-->

   #To
   <!--multicast channel="228.5.6.7" port="14021" timeout="1000"/-->
   <namingService name="Cos" host="<your host name>" port="<Port Number>"/>
   # Ensure that you specify the port number within the range reserved for
   the Reports server (14021 to 14030).
   ```

3. Start the naming service process.

   ```
   $DOMAIN_HOME/reports/bin/namingservice.sh <Port_Number>
   ```

4. Start `WLS_REPORTS` component from the WebLogic Console and the standalone Reports Server:

   ```
   $DOMAIN_HOME/bin/startComponent.sh <standalone_repserver>
   ```

5. Run the following command to verify if Namingservice (COS) discovery mechanism is working fine:

   ```
   $DOMAIN_HOME/reports/bin/rwdiag.sh -findAll
   ```

> **Note:**
>
> `namingService` is not supported for Reports server discovery in High Availability setup.
>
> When you start the Reports server, you receive the following error: `XML Parse exception :Element 'namingService' not expected`. This error is displayed if the `rwnetwork.conf` files are not configured correctly with `namingService`.