OCI Database Migration Service End-To-End Online Migration Tutorial

Aimed for scenarios where your application must remain online, and your source database has a direct connection to OCI.

November 2022, Version 2.3
Copyright © 2022, Oracle and/or its affiliates
Public
Purpose statement

This document walks you through all the steps to get you started using Oracle Cloud Infrastructure (OCI) Database Migration (DMS). You will provision a Virtual Cloud Network (VCN), an Oracle Database 19c instance, and an Oracle Autonomous Database (ADB) instance to perform a database migration using DMS.

With DMS we make it quick and easy for you to migrate databases from on-premises, Oracle, or third-party cloud into Oracle databases on OCI.

Disclaimer

This document in any form, software, or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement, nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle. Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.
# Table of contents

Purpose statement 2  
Disclaimer 2  
Introduction to OCI Database Migration – DMS 4  
Task 0 – Understand New DMS Concepts 4  
Task 1 – Have the Administrator Set Required Permissions 6  
Task 2 – Sign In and Open DMS Console 7  
Task 3 – Create Virtual Cloud Network 7  
Task 4 – Update Security List for Virtual Cloud Network Subnet 7  
Task 5 – Create Vault 8  
Task 6 – Create Source Database 9  
Task 7 – Create Target Autonomous Database 10  
Task 8 – Create GoldenGate Marketplace Instance 11  
Task 9 – Prepare Source and Target Databases 17  
Task 10 – Create Object Store Bucket for Datapump Storage 21  
Task 11 – Create Registered Database for Source CDB 21  
Task 12 – Create Registered Database for Source PDB 23  
Task 13 – Create Registered Database for Target 24  
Task 14 – Create Migration 25  
Task 15 – Validate Migration 28  
Task 16 – Run Migration 32  
Task 17 – Review your data 34
Introduction to OCI Database Migration – DMS

OCI Database Migration (DMS) provides a high performant, self-service experience to achieve migrations, which include:

Migration of data from on-premises, Oracle, or 3rd party cloud databases into Oracle databases on OCI.

Logical Online and Offline Migration providing enterprise-level migration with minimal downtime and on-premises to cloud migration.

Based on industry-leading GoldenGate replication and Zero Downtime Migration engine

DMS Documentation:
Please review the documentation here –
https://docs.oracle.com/en/cloud/paas/database-migration

Task 0 – Understand New DMS Concepts

DMS provides a fully managed approach to migrating databases from various locations into OCI-hosted databases.

Migrations can be either one of the following modes:

- **Offline**: The Migration makes a point-in-time copy of the source to the target database. Any changes to the source database during migration are not copied, requiring any applications to stay *offline* for the duration of the migration.

- **Online**: The Migration makes a point-in-time copy and replicates all subsequent changes from the source to the target database. This allows applications to stay *online* during the migration and then be switched over from source to target database.

DMS supports both offline and online mode. In the first release, we will support Oracle databases located on-premises, in 3rd party clouds, or on Oracle OCI as source and Oracle Autonomous Database shared or dedicated as target.

The DMS service runs as a managed cloud service separate from the user’s tenancy and resources. The service operates as a multi-tenant service in a DMS Service Tenancy and communicates with the user’s resources using Private Endpoints (PEs). PEs are managed by DMS and are transparent to the user.
Compartment: A compartment is a collection of related resources (such as cloud networks, compute instances, or block volumes) that can be accessed only by those groups that have been given permission by an administrator in your organization. For example, one compartment could contain all the servers and storage volumes that make up the production version of your company's Human Resources system. Only users with permission to that compartment can manage those servers and volumes.

Data region: A geographical region that's associated with one or more data centers. When you sign up for an Oracle Cloud account, you select a default data region, where your services will be hosted.

DMS Control Plane: Used by DMS end user to manage Migration and Registered Database objects. The control plane is exposed through the DMS Console UI as well as the Rest API.

DMS Data Plane: Managed by DMS Control Plane and transparent to the user. The GGS Data Plane manages ongoing migration jobs and communicates with the user's databases and GoldenGate instance using PEs. The DMS data plane does not store any customer data, as data flows through GoldenGate and Data Pump directly within the user's tenancy.

Migration: A Migration contains metadata for migrating one database. It contains information about source, target, and migration methods and is the central object for users to run migrations. After creating a migration, a user can validate the correctness of the environment and then run the migration to perform the copy of database data and schema metadata from source to target.
**Migration Job:** A Migration Job displays the state or a given Migration execution, either for validation or migration purposes. A job consists of several sequential phases, users can opt to wait after a given phase for user input to resume with the following phase.

**Registered Database:** A Registered Database represents information about a source or target database, such as connection and authentication credentials. DMS uses the OCI Vault to store credentials. A registered database is reusable across multiple Migrations.

**Task 1 – Have the Administrator Set Required Permissions**

The following permissions need to be set to have access to the necessary objects, unless you have administrative privileges. The following permissions assume that the user is part of group DMS_LA and all resources are created in a compartment called DMS_LA. Have your tenancy administrator set these permissions.

**PERMISSIONS REQUIRED BY DMS TO USE DATABASES, VAULTS, AND NETWORKING**

- Allow group DMS_LA to inspect compartments in tenancy
- Allow group DMS_LA to inspect tenancies in tenancy
- Allow group DMS_LA to inspect tenancies in tenancy where request.operation='GetTenancy'
- Allow group DMS_LA to manage autonomous-database-family in compartment DMS_LA
- Allow group DMS_LA to manage database-family in compartment DMS_LA
- Allow group DMS_LA to manage instance-family in compartment DMS_LA
- Allow group DMS_LA to manage keys in compartment DMS_LA
- Allow group DMS_LA to manage object-family in compartment DMS_LA
- Allow group DMS_LA to manage odms-connection in tenancy
- Allow group DMS_LA to manage odms-job in tenancy
- Allow group DMS_LA to manage odms-migration in tenancy
- Allow group DMS_LA to manage public-ips in compartment DMS
- Allow group DMS_LA to manage secret-family in compartment DMS_LA
- Allow group DMS_LA to manage vaults in compartment DMS_LA
- Allow group DMS_LA to manage virtual-network-family in compartment DMS_LA
PERMISSIONS REQUIRED TO CREATE OGG MARKETPLACE IMAGE

- Allow group DMS_LA to use tag-namespaces in tenancy
- Allow group DMS_LA to manage app-catalog-listing in tenancy
- Allow group DMS_LA to use volume-family in compartment DMS_LA
- Allow group DMS_LA to use cloud-shell in tenancy. (Only if OCI cloud-shell will be used).

Task 2 – Sign In and Open DMS Console

To perform this learning path you need to have access to an OCI tenancy with access to a region where DMS is released, such as the US-Ashburn-1 region. Please review https://www.oracle.com/cloud/data-regions/ for available regions.

1. Open browser with URL https://console.us-ashburn-1.oraclecloud.com/ (Adjust for home region)
2. Log in using your tenancy name and username/password.
3. In the OCI console title bar change region if applicable.

Task 3 – Create Virtual Cloud Network

The following task is optional if a suitable VCN is already present.

1. In the OCI Console Menu, go to Networking > Virtual Cloud Networks
2. Pick a compartment on the left-hand side Compartment list. You need to have the necessary permissions for the compartment.
3. Press Start VCN Wizard and pick VCN with Internet Connectivity.
4. Enter a VCN Name, such as VCN_DMS_LA. Leave CIDR block defaults, unless you need non-overlapping addresses for peering later. Press Next.
5. Review Summary and press Create.

Task 4 – Update Security List for Virtual Cloud Network Subnet

This task assumes default permissions in your public subnet. If you disabled or restricted your default permissions such as port 22 SSH access or restricted egress, please add default permissions as needed.

1. In the OCI Console Menu, go to Networking > Virtual Cloud Networks and pick your VCN.
2. In the Subnets list, pick Public Subnet-VCN NAME.
3. In the Security Lists list, pick Default Security List for VCN NAME.
4. In the Ingress Rules list press Add Ingress Rules.
5. Enter the following values, otherwise leave defaults:
   - Source CIDR: 0.0.0.0/0
   - Destination Port Range: 443
   - Description: OGG HTTPS
   
   Close dialog by pressing Add Ingress Rules.
7. Enter the following values, otherwise leave defaults:
   - Source CIDR: **10.0.0.0/16**
   - Destination Port Range: **1521**
   - Description: Oracle DB access for PEs
   
   Close dialog by pressing Add Ingress Rules.

---

**Task 5 – Create Vault**

The following task is optional if a Vault is already present.

1. In the OCI Console Menu, go to **Identity & Security > Vault**.
2. Pick a compartment on the left-hand side **Compartment** list.
3. Press Create Vault.
4. In the Create Vault dialog, enter a Name such as **DMS_Vault**.
5. Close the dialog by pressing Create Vault.
6. Wait until the state of the new vault is **Active**.
7. Click on the new vault and press Create Key in the **Master Encryption Keys** list.
8. In the **Create Key** dialog, enter a Name such as **DMS_Key**.

9. Close the dialog by pressing **Create Key**.

**Task 6 – Create Source Database**

The following task is optional if a source database is already present. In this example the source database is a DBCS VM with Oracle Database 19c.

1. You will need an SSH key pair for logging into your database and GoldenGate environments. If you don’t already have one, please **create one**. **Important:** The key needs to be in RSA key in PEM format, other formats like OpenSSL are currently not supported. You can use a command like:

   ```
   ssh-keygen -t rsa -N '' -b 2048 -C "<key_name>" -f <path/root_name>
   ```

2. In the OCI Console Menu, go to Oracle Database > Oracle Base Database (VM, BM).

3. Press Create DB System.

4. Enter the following values, otherwise leave defaults. You can adjust shapes and storage to your use case requirements and available quota.
   - **Name:** SourceDB
   - Add public SSH keys: Upload the public key file you use.
   - Choose a license type: BYOL
   - Virtual cloud network: VCN_DMS_LA *(Or your VCN name)*
   - Client subnet: Public Subnet-VCN_DMS_LA *(Or your subnet name)*
   - Hostname prefix: sourcedb

5. Press **Next**

6. Enter the following values, otherwise leave defaults.
   - Database name: sourcedb
   - PDB name: pdb
   - Create administrator credentials – Password: **password of your choice**

7. Press Create DB System

8. The provisioning of the database can take 30 or more minutes. Wait for the Lifecycle State of the database to change to Active.

9. Open the database system SourceDB in the DB Systems table

10. Open the database sourcedb in the Databases table

11. Press DB Connection

12. Press Show next to the Easy Connect Connection String. A string similar to
sourcedb.sub12062328210.vcndmsla.oraclevcn.com:1521/sourcedb_iad158.sub12062328210.vcndmsla.oraclevcn.com

should be shown. Copy the string after the /, in this case
sourcedb_iad158.sub12062328210.vcndmsla.oraclevcn.com

This is the service name of your CDB, you will need this string later for accessing your database and creating migrations. Close the dialog.

13. Click on Pluggable Databases link on the left side under Resources section and click on pdb.

14. Repeat steps 11 and 12. This is the service name of your PDB a string similar to
pdb.sub12062328210.vcndmsla.oraclevcn.com

15. Go back to the DB Systems Details page of your database and select Nodes on the left-hand side Resources list.

16. The Nodes list shows the sourcedb node. Note the Public IP Address and Private IP Address of the node, in this case 129.213.162.34 and 10.0.0.3. You will need these IP addresses later.

<table>
<thead>
<tr>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
<tr>
<td>sourcedb</td>
</tr>
</tbody>
</table>

Task 7 – Create Target Autonomous Database

The following task is optional if a target autonomous database is already present. In the first phase of DMS LA an autonomous database with private IP address is required. In this example the target database is an ATP-shared instance.

1. You first need to create a Network Security Group for use in a Private IP ADB instance. In the OCI Console Menu, go to Networking > Virtual Cloud Networks and pick your VCN.

2. In the left-hand Resources list, pick Network Security Groups.


4. Enter Name such as DMS_NSG and press Next.

5. In the Rule box please enter the following entries, otherwise leave defaults:
• Source Type: CIDR
  • Source CIDR: 0.0.0.0/0

6. Press **Create**.
7. Now you can create the ADB instance. In the OCI Console Menu, go to **Oracle Database > Autonomous Transaction Processing**.
8. Pick a compartment on the left-hand side **Compartment** list.
9. Press **Create Autonomous Database**.

10. Enter the following values, otherwise leave defaults. You can adjust shapes and storage to your use case.
    • Display Name: TargetATP
    • Database name: TargetATP
    • Create administrator credentials – Password: password of your choice
    • Access Type: Private endpoint access only
    • Virtual cloud network: VCN_DMS_LA (Or your VCN name)
    • Client subnet: Public Subnet-VCN_DMS_LA (Or your subnet name)
    • Network security group: DMS_NGS (Or your NSG name)
    • Choose a license type: Bring Your Own License (BYOL)

11. Close the dialog by pressing **Create Autonomous Database**.

**Task 8 – Create GoldenGate Marketplace Instance**

1. The following task is optional if a suitable GoldenGate instance is already present. This instructions apply for OGG 21c.
2. In the OCI Console Menu, go to **Marketplace > All Applications**.
3. In the “Search for listings…” field enter GoldenGate.
4. In the All Applications search results press on the box with Oracle GoldenGate for Oracle – Database Migrations.
5. In the upper right-hand side box, make sure the Compartment is correct, check the “I have reviewed and accept the Oracle Standard Terms and Restrictions” box and press **Launch Stack**.
6. Creating the stack is divided into three stages, Stack Information, Configurable Variables, and Final Review. For the first stage, Stack Information, scroll down to the bottom and click on Next.

7. Configure the following variables for section **Name for New Resources**:
   - Display Name (As is)
   - Host DNS Name, enter oggdms

8. On **Network Settings** enter the following values, otherwise leave defaults:
   - VCN Network Compartment: **DMS_LA** (Or your Compartment)
   - VCN: **VCN_DMS_LA** (Or your VCN name)
   - Subnet Network Compartment: **DMS_LA** (Or your Compartment)
   - Subnet: **Public Subnet-VCN_DMS_LA** (Or your subnet name)
9. Configure the following variables for the section **Instance Settings**, bear in mind that the Availability Domain must be the same as the rest of your resources. The Compute Shape of choice is VM. Standard2.1:

- **Availability Domain**: **US-ASHBURN-AD-1** (Adjust for the correct region)
- **Compute Shape**: Pick according to your available quota and use case, **VM. Standard2.1** is sufficient for simple use cases
- **Assign Public IP** (Check)
- **Check Custom Volume Sizes** to save on block storage quota. **50GB** for each volume is sufficient for simple use cases

10. Configure the following variables for the section **Create OGG Deployment**:

- **Deployment Name**, enter **Marketplace** (you will require this when creating the migration)
- **Deployment 2 - Autonomous Database** (Check)
- Deployment 2 - Autonomous Database Compartment: **DMS_LA** (Or your Compartment)
- Deployment 2 - Autonomous Database Instance: **TargetATP** (Or your ADB instance)

### Create OGG Deployment

<table>
<thead>
<tr>
<th>Deployment Name</th>
<th>Marketplace</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name for OGG deployment</td>
<td></td>
</tr>
<tr>
<td><strong>Deployment - Autonomous Database</strong></td>
<td></td>
</tr>
<tr>
<td>Select if deployment connects to an Autonomous Database</td>
<td></td>
</tr>
<tr>
<td><strong>Deployment - Autonomous Database Compartment</strong></td>
<td></td>
</tr>
<tr>
<td>Choose...</td>
<td></td>
</tr>
<tr>
<td>Compartment that existing Autonomous Database resides in</td>
<td></td>
</tr>
<tr>
<td><strong>Deployment - Autonomous Database Instance</strong></td>
<td></td>
</tr>
<tr>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

Choose an Autonomous Database instance

*This variable is required.*

11. Configure the following variable for the section **Shell Access**:

**SSH Public Key**: Open your *.pub key in an editor and cut and paste the contents into this field.

### Shell Access

<table>
<thead>
<tr>
<th>SSH Public Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public Key for allowing SSH access as the 'opc' user</td>
</tr>
</tbody>
</table>

12. Review your entries & Click **Next**

13. Proceed to do a final review & Click **Create** when ready. Provisioning will take a few minutes. Please follow the logs for additional information. Wait for the Job State to change to Succeeded.

### Create Stack

<table>
<thead>
<tr>
<th>Stack Information</th>
<th>Continuous Variables</th>
<th>Review</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Run Apply on the created stack?**

Immediately provision the resources defined in the Tomtom configuration by running the Apply action on the new stack.

- Run Apply

14. Upon creation, scroll down to the bottom of the logs and copy the **ogg_instance_id**, and the **ogg_public_ip**. Save them for later use, you will require them during the migration.
15. You will now need to open an SSH terminal to the GoldenGate instance. The instructions are for Unix-style ssh command:

```
ssh -i <private_key_file> opc@<ogg_public_ip>
```

16. (For example: `ssh -i ./id_rsa opc@193.122.169.5`)

17. You can use OCI Cloud Shell for this step, the icon is at the top right of the screen:

![OCI Cloud Shell Icon](image)

19. In the shell, enter the following command:

```
cat ./ogg-credentials.json
```

20. The output is like this:

```
{"username": "oggadmin", "credential": "Xvp7vUL4DXLKjqv"}
```

21. Copy the credential password (Here: Xv%p7vUL4DXLKjqv) You will need it in the next steps.

22. In a browser, open the OGG Service Manager home page:

```
HTTPS://<ogg_public_ip>
```

(For example https://193.122.169.5)

23. The browser will show warnings that the page is insecure because it uses a self-signed certificate. Ignore those warnings and proceed). On the GoldenGate Service Manager login screen, enter username **oggadmin** and the **password** copied from the credentials file earlier.
24. On the top-left “hamburger” menu, choose Administrator. In the Users table, press the edit icon for user **oggadmin**
   Enter the following values, otherwise leave defaults:
   
   - Info: admin
   - Password/Verify Password: <password of your choice>
   - Press Submit

25. You will be logged out, log back in with **oggadmin** and your password.

26. In the Services table, click on the port of the Marketplace Administration Server (typically 9011), this will open a new Sign In page for the **Oracle GoldenGate Administrator Server**.

27. Enter the username and password (labeled credential) from the out copied from the ogg-credentials.json file from step above. Do not enter the newly updated password.

28. Click on the top left navigation menu and repeat the steps:

29. On the top-left “hamburger” menu, choose Administrator. In the Users table, press the edit icon for user **oggadmin**

30. Enter the following values, otherwise leave defaults:
   
   - Info: admin
   - Password/Verify Password: <password of your choice>
• Press Submit

31. You will be logged out. You may now close any Oracle GoldenGate Service Manager leftover tab.

Task 9 – Prepare Source and Target Databases

This task prepares required user accounts and settings for Migration in the Source DB. It assumes default settings in the database. If you changed default settings, further settings might be necessary.

1. Open an SSH terminal to the source database instance. The instructions are for Unix-style ssh command:

   ```
   ssh -i <private_key_file> opc@<dbnode_public_ip>
   ```

2. You can use OCI Cloud Shell for this step, the icon is at the top right of the screen:

3. Create a new directory in the user volume:

   ```
   sudo su - oracle
   mkdir /u01/app/oracle/dumpdir
   ```

4. Open an SSH terminal to the GoldenGate instance. See the GoldenGate task for details.

5. Create the script file `create_ggadmin_cdbroot.sql` with the following content (Replace `<password>` with actual password, use double quotes to delimit it i.e "********"):

   ```sql
   create user c##ggadmin identified by <password> default
   tablespace users temporary tablespace temp;
   grant connect, resource to c##ggadmin container=all;
   grant select on sys.ccol$ to c##ggadmin container=all;
   grant select on sys.cdef$ to c##ggadmin container=all;
   grant select on sys.col$ to c##ggadmin container=all;
   grant select on sys.con$ to c##ggadmin container=all;
   grant select on sys.deferred_stg$ to c##ggadmin container=all;
   grant select on sys.icol$ to c##ggadmin container=all;
   ```
grant select on sys.ind$ to c##ggadmin container=all;
grant select on sys.lob$ to c##ggadmin container=all;
grant select on sys.lobfrag$ to c##ggadmin container=all;
grant select on sys.obj$ to c##ggadmin container=all;
grant select on sys.seg$ to c##ggadmin container=all;
grant select on sys.tab$ to c##ggadmin container=all;
grant select on sys.tabcompart$ to c##ggadmin container=all;
grant select on sys.tabpart$ to c##ggadmin container=all;
grant select on sys.tabsubpart$ to c##ggadmin container=all;
grant create view to c##ggadmin container=all;
grant execute on dbms_lock to c##ggadmin container=all;
alter user c##ggadmin quota 100M on USERS container=all;
grant unlimited tablespace to c##ggadmin container=all;
exec dbms_goldengate_auth.GRANT_ADMIN_PRIVILEGE('c##ggadmin',container=>'all');
alter system set streams_pool_size=2G scope=both SID='sourcedb';
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
alter system switch logfile;
ALTER SYSTEM SET ENABLE_GOLDENGATE_REPLICATION=TRUE SCOPE=BOTH;

6. Create the script file create_ggadmin_pdbroot.sql with the following content (Replace <password> with actual password, use double quotes to delimit it i.e "*********")

```
create user ggadmin identified by <password> default tablespace users;
grant connect, resource to ggadmin;
grant select on sys.ccol$ to ggadmin;
grant select on sys.cdef$ to ggadmin;
grant select on sys.col$ to ggadmin;
grant select on sys.con$ to ggadmin;
grant select on sys.deferred_stg$ to ggadmin;
grant select on sys.icol$ to ggadmin;
grant select on sys.ind$ to ggadmin;
grant select on sys.lob$ to ggadmin;
grant select on sys.lobfrag$ to ggadmin;
grant select on sys.obj$ to ggadmin;
grant select on sys.seg$ to ggadmin;
grant select on sys.tab$ to ggadmin;
grant select on sys.tabcompart$ to ggadmin;
grant select on sys.tabpart$ to ggadmin;
grant select on sys.tabsubpart$ to ggadmin;
grant create view to ggadmin;
grant execute on dbms_lock to ggadmin;
alter user ggadmin quota 100M on USERS;
grant unlimited tablespace to ggadmin;
exec dbms_goldengate_auth.GRANT_ADMIN_PRIVILEGE('ggadmin');
```
7. Enter the following commands:

    export ORACLE_HOME="/u01/app/ogg/lib/instantclient"

    $ORACLE_HOME/sqlplus sys/<db password>@<db private ip>/<db cdb service> as sysdba

8. In SQL Plus enter the following commands:

    SQL> @create_ggadmin_cdbroot.sql
    User created.
    [...] System altered.

    SQL> show pdbs

    CON_ID  CON_NAME       OPEN MODE  RESTRICTED
    -------  -----------------  ----------  ---------
          2 PDB$SEED        READ ONLY  NO
          3 PDB           READ WRITE NO

    SQL> alter session set container=PDB;
    Session altered.

    SQL> @create_ggadmin_pdbroot.sql
    User created.
    [...] Grant succeeded.
    PL/SQL procedure successfully completed.

    SQL> quit;

9. The next steps add a user HR01 with a sample table and data. If your database already contains data for migration, you can skip these steps.

    Create the script file create_hr01.sql with the following content:

    ```sql
    DROP USER HR01 CASCADE;
    CREATE USER HR01 IDENTIFIED BY HR##hr01123;
    GRANT CONNECT,RESOURCE,CREATE TABLE,CREATE SEQUENCE to HR01;
    GRANT CREATE ANY PROCEDURE to HR01;
    ALTER USER HR01 quota unlimited on users;
    CREATE TABLE HR01.EMPL (col1 number, col2 varchar2(9), col3 varchar2(100), col4 timestamp);
    ALTER TABLE HR01.EMPL ADD CONSTRAINT EMPL_i1 PRIMARY KEY (col1, col2);
    ```

10. Create the script file data_hr01.sql with the following content:

    ```sql
    SET ECHO OFF;
    SET HEADING OFF;
    SET FEEDBACK OFF;
    SET SERVEROUTPUT ON;
    DECLARE
        SCN     HR01.EMPL.COL1%TYPE;
        RND1    HR01.EMPL.COL2%TYPE;
    ```
11. Enter the following commands:

```
$ORACLE_HOME/sqlplus sys/<db password>@<db private ip>/<db pdb service>
```

as sysdba

12. In SQL Plus enter the following commands:

```
SQL> @create_hr01.sql
DROP USER HR01 CASCADE  (You can ignore this error)
```

```
ERROR at line 1:
ORA-01918: user 'HR01' does not exist
```

```
SQL> @data_hr01.sql
Number of Rows = 0
[...]
Number of rows = 1000
SQL> quit
```

Your source DB now has a user HR01 with a table EMPL that has 1000 rows.

13. The next steps will connect to the target ADB instance and enable the standard ggadmin user.

14. You can skip these steps if the user is already enabled.
15. Make sure the ADB regional wallet has been placed in /u02/deployments/Marketplace/etc/adb. If not, you can download the zip file from OCI Console and unzip it there.

16. Modify sqlnet.ora so it correctly has the wallet location (needed if connecting with sqlplus):

```bash
cat sqlnet.ora
WALLET_LOCATION = (SOURCE = (METHOD = file) (METHOD_DATA = (DIRECTORY="/u02/deployments/Marketplace/etc/adb")))
SSL_SERVER_DN_MATCH=yes
```

17. Unlock ggadmin in ADB, connecting from the OGG instance using sqlplus:

18. You need to set the following Export variables:

```bash
export ORACLE_HOME="/u01/app/ogg/lib/instantclient"
export LD_LIBRARY_PATH="$ORACLE_HOME"
export PATH="$ORACLE_HOME:$PATH"
export TNS_ADMIN="/u02/deployments/Marketplace/etc/adb"
$ORACLE_HOME/sqlplus admin/ <ATP password>@ targetatp_high
```

19. In SQL Plus enter the following commands:

```sql
SQL> alter user ggadmin identified by <new password> account unlock;
User altered.
SQL> quit
```

**Task 10 – Create Object Store Bucket for Datapump Storage**

Object Store is used as temporary storage between source and target databases with Datapump. This task is creating an empty bucket for use in the migration.

1. In the OCI Console Menu, go to Storage > Object Storage & Archive...
2. Press Create Bucket.
3. On the page Create Bucket, fill in the following entries, otherwise leave defaults:
   - Bucket Name: DMSStorage
4. Press Create Bucket

**Task 11 – Create Registered Database for Source CDB**

For this task you need the following info from previous steps:

- Source DB Private IP
- Source DB CDB Service Name

1. In the OCI Console Menu, go to Migration > Database Migration > Registered Databases.
2. Press Register Database.
3. On the page Database Details, fill in the following entries, otherwise leave defaults:
   - Name: SourceCDB
   - Vault: DMS_Vault
   - Encryption Key: **DMS_Key**
   - Database Type: Database (Bare Metal, VM, Exadata)
   - Database System: **SourceDB**
   - Database: **sourcedb**
   - Connect String: Change existing string by replacing the qualified hostname with the private IP of the database node. This is important as DMS does not accept FQDNs or hostnames in the connect string.
   - Subnet: Pick the Subnet that the DB is located in.

4. Press **Next**.

5. On the page Connection Details, fill in the following entries, otherwise leave defaults:
   - Database Administrator Username: **system**
   - Database Administrator Password: <Admin password>
• SSH Database Server Hostname: < DB Node Private IP Address>
• SSH Private Key: Select private key file
• SSH Username: opc
• SSH Sudo Location: /usr/bin/sudo

Press Register

Task 12 – Create Registered Database for Source PDB

For this task you need the following info from previous steps:
• Source DB Private IP
• Source DB PDB Service Name

1. In the OCI Console Menu, go to Migration > Database Migration > Registered Databases.

2. Press Register Database.

3. On the page Database Details, fill in the following entries, otherwise leave defaults:
   • Name: SourcePDB
   • Vault: DMS_Vault
   • Encryption Key: DMS_Key
   • Database Type: Database (Bare Metal, VM, Exadata)
   • Database System: SourceDB
   • Database: sourcedb
   • Connect String: Change existing string by replacing the qualified hostname with the private IP of the database node. This is important as DMS does not accept FQDNs or hostnames in the connect string. Then replace service name with PDB service name, for example 10.0.0.3:1521/pdb.subXXXXXXXX.vcndmsla.oraclevcn.com
   • Subnet: Pick the Subnet that the DB is located in.

4. Press Next.
5. On the page Connection Details, fill in the following entries, otherwise leave defaults:
   - Database Administrator Username: **system**
   - Database Administrator Password: *<Admin password>*
   - SSH Database Server Hostname: *<DB Node Private IP Address>*
   - SSH Private Key: Select private key file
   - SSH Username: **opc**
   - SSH Sudo Location: */usr/bin/sudo*

Press Register

**Task 13 – Create Registered Database for Target**

1. In the OCI Console Menu, go to Migration > Database Migration > Registered Databases.
2. Press Register Database.
3. On the page Database Details, fill in the following entries, otherwise leave defaults:
   - Name: TargetATP
   - Vault: DMS_Vault
   - Encryption Key: **DMS_Key**
   - Database Type: Autonomous Database
   - Database: **TargetATP**

Press **Next.**
4. On the page Connection Details, fill in the following entries, otherwise leave defaults:
   - Database Administrator Username: admin
   - Database Administrator Password: <Admin password>
   Press Register

**Task 14 – Create Migration**

1. In the OCI Console Menu, go to Migration > Database Migration > Migrations.
2. Press Create Migration.
3. On the page Add Details, fill in the following entries, otherwise leave defaults:
   - Name: TestMigration
   - Vault: DMS_Vault
   - Encryption Key: DMS_Key
   Press Next
4. On the page **Select Databases**, fill in the following entries, otherwise leave defaults:

- **Source Database**: **SourcePDB**
- **Check Database is pluggable database (PDB)**
- **Registered Container Database**: **SourceCDB**
- **Target Database**: **TargetATP**

Press **Next**

5. On the page **Migration Options**, fill in the following entries, otherwise leave defaults:

- In Initial Load select Datapump via Object Storage
- **Object Storage Bucket**: **DMSSStorage**
- **Export Directory Object**:
  - Name: **dumpdir**
  - Path: `/u01/app/oracle/dumpdir`
• Check Use Online Replication
• GoldenGate Hub URL: **https://<goldengate public IP>**
  For example:
  https://193.122.169.5
• GoldenGate Administrator Username: **oggadmin**
• GoldenGate Administrator Password: **<As previously selected>**

• Source Database
  • GoldenGate Deployment Name: If you are not using the default **Marketplace** GoldenGate deployment name, enter the value you provided.
  • Database Username: **ggadmin**
  • Database Password: **<As previously selected>**
  • Container Database Username: **c##ggadmin**
  • Container Database Password: **<As previously selected>**
• **Target Database**
  - **GoldenGate Deployment Name:** This will be the same as the Source Database GoldenGate Deployment Name.
  - **Database Username:** `ggadmin`
  - **Database Password:** `<As previously selected>`

• Press Show Advanced Options
• Press Replication tab
• **Golden Gate Instance OCID:** `<OCID as copied from GoldenGate compute instance>`

6. Click **Create**

**Task 15 – Validate Migration**

In this step you will validate a migration prior to running it. It will check that all associated database and GoldenGate environments are correctly set up.

1. In the OCI Console Menu, go to Migration > Database Migration > Migrations.
2. Select TestMigration.
3. If Migration is still being created, wait until Lifecycle State is Active.
4. Press Validate button
5. Click on Jobs in left-hand Resources list
6. Click on most recent Evaluation Job
7. Click on Phases in left-hand Resources list
8. Phases will be shown, and status will be updated as phases are completed. It can take 2 minutes before the first phase is shown.

9. If a phase has failed, it will show with status Failed. In this case press Download Log to learn more about the reason of failure. Press Abort on a failed job to allow further jobs or deleting of the migration.

10. Click Validate Pre-migration Advisor phase name to open the Validation pre-migration advisor detail page (You should not find issues in for this exercise but below lines would walk you thru an event when the phase fails). From this page you can download the CPAT report, view the report statistics, and drill down in the Checks list as shown:
11. You can still download the report as a text file, but now you can also navigate through the different checks. The summary view is displayed as follows:

12. You can click a check name in the list to display details about a specific check from the CPAT report. You can mark a check as **Reviewed** or **Unreviewed**, this state is only for your convenience to track each check. For certain checks, CPAT generates a remedial script on the file system of the source database server. You can run the script on the source database to resolve the issue identified by the check. The checks page will also let you filter by this state (left side of screen):

13. The **View check details** panel is displayed as follows:
14. Once you have cleared all “Action Required” checks then the validation Job can be run again. Repeat the process until **Validate premigration advisor** phase completes with no error as shown:

15. **Excluded Objects**: View the report of all excluded objects based on static exclusion rules as well as dynamic exclusion settings made by the user. The excluded objects list displays objects that are excluded from migration as shown:
Task 16 – Run Migration

After successful validation, a Migration can be run to perform the data transfer.

1. In the OCI Console Menu, go to Migration > Database Migration > Migrations.
2. Select TestMigration.
3. Press Start to begin the migration.
4. The Start Migration dialog is shown. Select the phase Monitor GoldenGate Lag in the Require User Input After list. This will cause the replication to run continuously until the Migration is resumed. Press Start to begin the Migration.

5. Click on Jobs in left-hand Resources list
6. Click on most recent Evaluation Job
7. Click on Phases in left-hand Resources list
8. Job phases are updated as the migration progresses
9. When the migration has reached the state to wait for user input, the migration job changes to **Waiting** state. This is the point where a migration user would stop the source application so that no more transactions are applied to the source DB. You can now press **Resume** on the job to complete replication.

10. In the Resume Job dialog, chose the **Switchover App** phase and press **Resume**. The Switchover App phase will gracefully stop replication and signal the target application to initiate transactions to the target DB.

11. After Job resumes and waits after Switchover App phase, press Resume. Select the last phase **Cleanup** and press **Resume**. (This is a temporary workaround tracked by ZDMCS-440, when fixed no phase has to be selected to run the migration to the end)
12. The migration runs the final cleanup phases and shows as Succeeded when finished.

**Task 17 – Review your data**

As a final step, you can review that your information is correctly migrated, for this exercise we need to validate that target database has a user HR01 with a table EMPL that contains 1000 rows.

We can start by checking our Initial Export, open your Migration and review the log of our “Export Initial Load” phase, click on the 3 dots at the top right and press “download log”. Open the file on a text editor, you should see a record like this:

```
16-FEB-22 17:16:58.228: W-2 . exported "HR01"."EMPL" 29.80 KB 1000 rows in 1 seconds using direct_path
```

To validate the data was loaded into our target database, we need to repeat the previous steps but selecting “Import Initial Load” phase, you should see a record like this on the log:

```
10-FEB-22 22:06:54.352:W-1 . imported "HR01"."EMPL" 29.80 KB 1000 rows in 3 seconds using external_table
```

We can notice how the same 1000 records, which were exported, were also imported into our target table.

Additionally you can connect to your target Database, open a SQL terminal and run the following query:

```
select count(*) from HR01.EMPL;
```

```
SQL> select count(*) from HR01.EMPL;
   COUNT(*)
----------
     1000
SQL> 
```

The count should display the 1000 records on the HR01.EMPL table.
You are now finished with this lab and ready to migrate your databases!