

## Lab 13: Moving PDBs

### Overview

In this lab, you will create a new CDB named `CDBDEV` and several PDBs, then practice moving a PDB from `CDBLAB` to `CDBTEST` using the near-zero downtime PDB relocation feature.

### Part A: Creating the New CDB and PDBs

#### 1. Create the New CDB ( `CDBDEV` )

- a. Open a terminal window and set the environment variables for `CDBLAB` :

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBLAB
The Oracle base remains unchanged with value /u01/app/oracle
```

- b. Start SQL\*Plus and log in as a user with SYSDBA privileges:

```
sqlplus / as sysdba
```

- c. Create the new CDB `CDBDEV` :

```
CREATE PLUGGABLE DATABASE cdbdev
ADMIN USER pdb_admin IDENTIFIED BY password
FILE_NAME_CONVERT = ('/u01/app/oracle/oradata/CDBLAB/pdbseed',
'/u01/app/oracle/oradata/CDBDEV');
```

#### Expected Output:

```
Pluggable database created.
```

#### 2. Create PDBs in `CDBDEV`

- a. Open a terminal window and set the environment variables for `CDBDEV` :

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBDEV
The Oracle base remains unchanged with value /u01/app/oracle
```

- b. Start SQL\*Plus and log in as a user with SYSDBA privileges:

```
sqlplus / as sysdba
```

- c. Create PDBs `PDB1` , `PDB2` , and `PDB3` :

```
CREATE PLUGGABLE DATABASE pdb1 FROM pdb$seed
ADMIN USER pdb1_admin IDENTIFIED BY password
FILE_NAME_CONVERT = ('/u01/app/oracle/oradata/CDBDEV/pdbseed',
'/u01/app/oracle/oradata/CDBDEV/pdb1');

CREATE PLUGGABLE DATABASE pdb2 FROM pdb$seed
ADMIN USER pdb2_admin IDENTIFIED BY password
FILE_NAME_CONVERT = ('/u01/app/oracle/oradata/CDBDEV/pdbseed',
'/u01/app/oracle/oradata/CDBDEV/pdb2');
```

```
CREATE PLUGGABLE DATABASE pdb3 FROM pdb$seed
ADMIN USER pdb3_admin IDENTIFIED BY password
FILE_NAME_CONVERT = ('/u01/app/oracle/oradata/CDBDEV/pdbseed',
'/u01/app/oracle/oradata/CDBDEV/pdb3');
```

#### Expected Output:

```
Pluggable database created.
```

#### d. Open the PDBs:

```
ALTER PLUGGABLE DATABASE pdb1 OPEN;
ALTER PLUGGABLE DATABASE pdb2 OPEN;
ALTER PLUGGABLE DATABASE pdb3 OPEN;
```

## Part B: Moving a PDB

### 1. Prepare for PDB Relocation

#### a. Open a terminal window and set the environment variables for CDBLAB :

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBLAB
The Oracle base remains unchanged with value /u01/app/oracle
```

#### b. Start SQL\*Plus and log in as a user with SYSDBA privileges:

```
sqlplus / as sysdba
```

#### c. Verify that the source ( CDBLAB ) is configured to use local undo:

```
SET sqlprompt "CDBLAB1> "
SELECT property_name, property_value
FROM database_properties
WHERE property_name = 'LOCAL_UNDO_ENABLED';
```

#### Expected Output:

PROPERTY_NAME	PROPERTY_VALUE
LOCAL_UNDO_ENABLED	TRUE

### 2. Verify the Test User and Table in PDB3

#### a. Connect to the PDB3 service:

```
connect test@pdb3
Enter password: password
```

#### b. Verify the test data:

```
SELECT label, COUNT(*) FROM test.bigtab GROUP BY label;
```

### Expected Output:

LABEL	COUNT (*)
DATA FROM test.bigtab	10000

### 3. Prepare to Relocate PDB3 from CDBLAB to CDBTEST

a. In CDBLAB , create the database link to access CDBTEST :

```
connect / as sysdba
DROP PUBLIC DATABASE LINK link_cdbtest;
CREATE PUBLIC DATABASE LINK link_cdbtest
CONNECT TO system IDENTIFIED BY password
USING 'CDBTEST';
```

### Expected Output:

Database link created.

b. List the PDBs to verify they are open:

```
SHOW PDBS;
```

### Expected Output:

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PDB1	READ WRITE	NO
4	PDB2	READ WRITE	NO
5	PDB3	READ WRITE	NO

c. If any of the PDBs are not open, open them:

```
ALTER PLUGGABLE DATABASE ALL OPEN;
```

d. Save the state of the PDBs:

```
ALTER PLUGGABLE DATABASE ALL SAVE STATE;
```

### 4. Create the Directory for PDB3 in CDBTEST

a. Open a new terminal window and set the environment variables for CDBTEST :

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBTEST
The Oracle base remains unchanged with value /u01/app/oracle
```

b. Create the directory for PDB3 :

```
mkdir -p /u01/app/oracle/oradata/CDBTEST/pdb3
```

### 5. Relocate PDB3 from CDBLAB to CDBTEST

a. In `CDBTEST` , create the database link to access `PDB3` in `CDBLAB` :

```
connect / as sysdba
DROP PUBLIC DATABASE LINK link_orclcdb;
CREATE PUBLIC DATABASE LINK link_orclcdb
CONNECT TO system IDENTIFIED BY password
USING 'CDBLAB';
```

**Expected Output:**

```
Database link created.
```

b. Relocate `PDB3` from `CDBLAB` to `CDBTEST` :

```
CREATE PLUGGABLE DATABASE pdb3
FROM pdb3@link_orclcdb relocate
FILE_NAME_CONVERT=('/u01/app/oracle/oradata/CDBLAB/pdb3',
'/u01/app/oracle/oradata/CDBTEST/pdb3');
```

**Expected Output:**

```
Pluggable database created.
```

## 6. Open the Relocated `PDB3` in Read-Only Mode

a. Open the relocated `PDB3` in read-only mode:

```
ALTER PLUGGABLE DATABASE pdb3 OPEN READ ONLY;
```

**Expected Output:**

```
Pluggable database altered.
```

b. Set the session to the relocated `PDB3` and verify the data:

```
ALTER SESSION SET CONTAINER=pdb3;
SELECT label, COUNT(*) FROM test.bigtab GROUP BY label;
```

**Expected Output:**

LABEL	COUNT (*)
DATA FROM test.bigtab	10000

## 7. Verify and Finalize the Relocation

a. In `CDBLAB` , verify that `PDB3` does not exist anymore:

```
SELECT pdb_name, status FROM cdb_pdbs;
```

**Expected Output:**

PDB_NAME	STATUS
-----	-----

PDB\$SEED	NORMAL
PDB1	NORMAL
PDB2	NORMAL

b. In `CDBTEST` , open `PDB3` in read-write mode:

```
ALTER PLUGGABLE DATABASE pdb3 OPEN READ WRITE;
```

**Expected Output:**

```
Pluggable database altered.
```

## 8. Clean Up

a. Drop the relocated `PDB3` in `CDBTEST` :

```
ALTER SESSION SET CONTAINER=CDB$ROOT;  
ALTER PLUGGABLE DATABASE pdb3 CLOSE;  
DROP PLUGGABLE DATABASE pdb3 INCLUDING DATAFILES;
```

**Expected Output:**

```
Pluggable database dropped.
```

b. In `CDBLAB` , revoke the `SYSOPER` privilege from