# Practices for Lesson 14

# Multitenant Architecture and RAC Environment

# Practices for Lesson 14: Overview

## Overview

In this practice, you will create a new CDB named `cdb1` including one PDB named `pdb1`. The CDB is hosted in an existing server pool.

Then you will create another PDB named `pdb2` and manage the services to affinitize the PDB services to instances.

At the end of the practice, you drop the `pdb2` PDB.

## Practice 12-1: Creating a CDB

### Overview

In this practice, you will create a new CDB named `cdb1` with DBCA.

#### Pre CDB Creation Tasks

As `root`, start `host03`. Then, remove the existing RAC database. From your classroom PC desktop, execute `ssh -X oracle@ol7-122-rac1` to open a terminal session on `ol7-122-rac1` asthe `oracle` user. Then navigate to /`u01/app/oracle/product/12.1.0/dbhome_1/bin` and execute DBCA.

```
[root@ol7-122-rac1 ~]$ xhost +
Password:

[root@ol7-122-rac1 ~]# su - oracle

[oraclec@ol7-122-rac1 ~]# cd
/u01/app/oracle/product/12.1.0/dbhome_1/bin

[oracle@ol7-122-rac1 bin]$ ./dbca
```

| Step | Screen/Page Description | Choices or Values |
|---|---|---|
| a. | Database Operation | Select Delete Database. Click Next. |
| b. | Delete Database | Select orcl and click Next. |
| c. | Management Options | Click Next. |
| d. | Summary | Click Finish. |
| e. | Database Configuration Assistant dialog box | You are informed that the instances and datafiles will be deleted. Click Yes to proceed. |
| f. | Database Configuration Assistant dialog box | You are informed that database deletion is complete. Click OK. |
| g. | Progress Page | Click Close. |

Remove the `orcldb` server pool.

```
[oracle@ol7-122-rac1 bin]$ /u01/app/12.1.0/grid/bin/srvctl
removesrvpool -serverpool orcldb

[oracle@ol7-122-rac1 bin]$ /u01/app/12.1.0/grid/bin/srvctl status
srvpool
```

```
Server pool name: Free
Active servers count: 3
Server pool name: Generic
Active servers count: 0
[oracle@ol7-122-rac1 bin]$
```

## Tasks

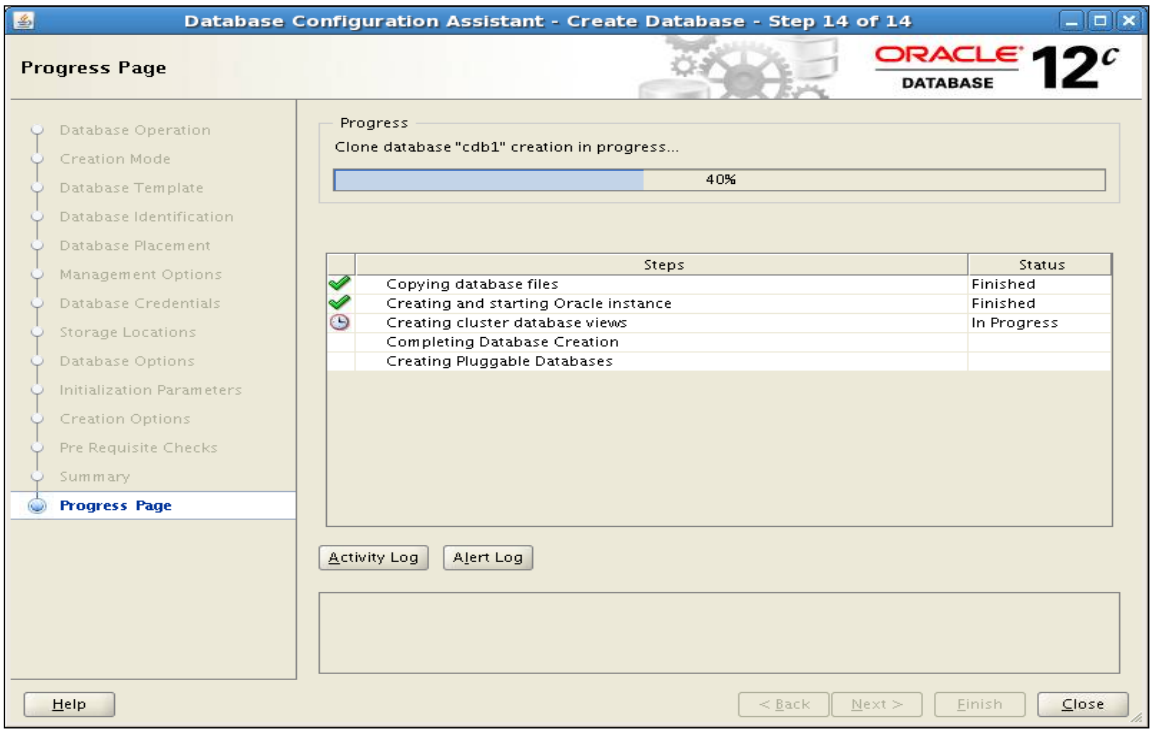
1. Start DBCA and perform the following steps.

```
[oracle@ol7-122-rac1 bin]$ ./dbca
```

| Step | Window/Page Description | Choices or Values |
|---|---|---|
| a. | Step 1: Database Operation | Select “Create Database.”<br>Click Next. |
| b. | Step 2: Creation Mode | Select “Advanced Mode.”<br>Click Next. |
| c. | Step 3: Database Template | Select “Oracle Real Application Clusters (RAC) database” for Database Type.<br>Select “Policy-Managed” for Configuration Type.<br>Select “General Purpose or Transaction Processing.”<br>Click Next. |
| d. | Step 4: Database Identification | Enter<br>Global Database Name: `cdb1`<br>Select **“Create As Container Database.”**<br>Select **“Create A Container Database with one or more PDBs.**”<br>Select 1 for Number of PDBs.<br>Enter `pdb1` for PDB Name.<br>Click Next. |
| e. | Step 5: Database Placement | Select “Create New Server pool for this database”.<br>Enter “`cdb1pool`“ for Server pool Name and 3 for Cardinality.<br>Click Next. |
| f. | Step 6: Management Options | Deselect “Configure Enterprise Manager (EM) Database Express.”<br>Click Next. |

| Step | Window/Page Description | Choices or Values |
|---|---|---|
| g. | Step 7: Database Credentials | Select “Use same Administrative password…”<br>Enter:<br>Password: *database_administrative_password*<br>Confirm password: *database_administrative_password*<br>Click Next. |
| h. | Step 8: Storage Locations | Confirm Storage type is “Automatic Storage Management (ASM).”<br>Confirm “Use Common Location for All Database Files.” in `+DATA` diskgroup.<br>Deselect “Specify Fast Recovery Area”.<br>Click Next. |
| i. | Step 9: Database Options | Click Next. |
| j. | Step 10: Initialization Parameters | Set “Memory Size (SGA and PGA)” to 840 MB.<br>Select “Use Automatic Memory Management”<br>Select “Character Sets.”<br>Select “Use Unicode (**AL32UTF8**).”<br>Click Next. |
| k. | Step 11: Creation Option | Select “Create Database.”<br>Click Next. |
| l. | Step 12: Pre Requisite Checks | Click Next. |
| m. | Step 13: Summary | Click Finish. |
| n. | Step 14: Progress Page | On the Database Configuration Assistant page (for password management), click Exit.<br>Click Close. |

The screenshot below corresponds to step n.




2. Explore the CDB instances hosted on the nodes of the server pool. You will also see that the `pdb1` PDB can be accessed on any instance of the CDB just like a non-CDB can be accessed on any instance in a RAC environment.

   a. Check the `cdb1pool` server pool and its cardinality.

```
[oracle@ol7-122-rac1 bin]$ su - grid
grid@ol7-122-rac1's password:
[grid@ol7-122-rac1 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid

[grid@ol7-122-rac1 ~]$ srvctl status srvpool
Server pool name: Free
Active servers count: 0
Server pool name: Generic
Active servers count: 0
Server pool name: cdb1pool
Active servers count: 3

[grid@ol7-122-rac1 ~]$ srvctl status srvpool -serverpool cdb1pool
Server pool name: cdb1pool
Active servers count: 3
```

```
[grid@ol7-122-rac1 ~]$ exit
logout

[oracle@ol7-122-rac1 ~]$
```

b. Use SRVCTL to know on which nodes the instances of the CDB are running, as you traditionally do for any non-CDB.

```
[oracle@ol7-122-rac1 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
[oracle@ol7-122-rac1 ~]$ cd $ORACLE_HOME/bin

[oracle@ol7-122-rac1 bin]$ ./srvctl status database -d cdb1

[oracle@ol7-122-rac1 bin]$ pgrep -l cdb1_3
9770 ora_pmon_cdb1_3
9772 ora_psp0_cdb1_3
9774 ora_vktm_cdb1_3
9778 ora_gen0_cdb1_3
9780 ora_mman_cdb1_3
9784 ora_diag_cdb1_3
9786 ora_dbrm_cdb1_3
9790 ora_ping_cdb1_3
9792 ora_acms_cdb1_3
9794 ora_dia0_cdb1_3
9796 ora_lmon_cdb1_3
9798 ora_lmd0_cdb1_3
9800 ora_lms0_cdb1_3
9804 ora_rms0_cdb1_3
9806 ora_lmhb_cdb1_3
9808 ora_lck1_cdb1_3
9810 ora_dbw0_cdb1_3
9812 ora_lgwr_cdb1_3
9814 ora_ckpt_cdb1_3
9816 ora_smon_cdb1_3
9818 ora_reco_cdb1_3
9820 ora_lreg_cdb1_3
9822 ora_rbal_cdb1_3
9824 ora_asmb_cdb1_3
```

```
9826 ora_mmon_cdb1_3
9830 ora_mmnl_cdb1_3
9832 ora_d000_cdb1_3
9834 ora_s000_cdb1_3
9836 ora_mark_cdb1_3
9841 ora_gcr0_cdb1_3
9843 ora_lck0_cdb1_3
9857 ora_rsmn_cdb1_3
9906 ora_tmon_cdb1_3
9908 ora_tt00_cdb1_3
9960 ora_smco_cdb1_3
9962 ora_w000_cdb1_3
9968 ora_gtx0_cdb1_3
9970 ora_rcbg_cdb1_3
9972 ora_ppa7_cdb1_3
9987 ora_aqpc_cdb1_3
9989 ora_qm02_cdb1_3
9991 ora_q001_cdb1_3
9993 ora_q002_cdb1_3
9995 ora_qm05_cdb1_3
10013 ora_p000_cdb1_3
10015 ora_p001_cdb1_3
10017 ora_p002_cdb1_3
10019 ora_p003_cdb1_3
10319 ora_cjq0_cdb1_3
11166 ora_w001_cdb1_3
11995 ora_w002_cdb1_3
13641 ora_p004_cdb1_3
13643 ora_p005_cdb1_3
[oracle@ol7-122-rac1
bin]$
```

c. Use LSNRCTL to list the CDB instances on two nodes of the server pool.

1) Check the services on the first node.

```
[oracle@ol7-122-rac1 bin]$ ./lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-SEP-
2014 06:54:40

Copyright (c) 1991, 2014, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
```

```
------------------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0
- Production
Start Date                30-AUG-2014 07:22:00
Uptime                    2 days 23 hr. 32 min. 40 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
/u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File
/u01/app/grid/diag/tnslsnr/ol7-122-rac1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.247)(PORT=1521
)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.101)(PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully
[oracle@ol7-122-rac1 bin]$
```

2) Check the services on the second node.

```
[oracle@ol7-122-rac1 bin]$ ssh host02
Last login: Mon Sep 2 01:17:02 2014 from 192.0.2.1
[oracle@host02 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
```

```
The Oracle base has been set to /u01/app/oracle

[oracle@host02 ~]$ lsnrctl status
…
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.245)(PORT=1521
)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.102)(PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM3", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
The command completed successfully

[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@ol7-122-rac1 bin]$
```

d. Use SRVCTL to stop and restart the CDB as you traditionally would do for any non-CDB.

```
[oracle@ol7-122-rac1 bin]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_3
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@ol7-122-rac1 bin]$ srvctl stop database -d cdb1

[oracle@ol7-122-rac1 bin]$ srvctl status database -db cdb1
Instance cdb1_1 is not running on node host03
```

```
Instance cdb1_2 is not running on node host02
Instance cdb1_3 is not running on node ol7-
122-rac1

[oracle@ol7-122-rac1 bin]$ srvctl start database -d cdb1

[oracle@ol7-122-rac1 bin]$ srvctl status database -db cdb1
Instance cdb1_1 is running on node host03
Instance cdb1_2 is running on node host02
Instance cdb1_3 is running on node ol7-
122-rac1

[oracle@ol7-122-rac1 bin]$ cd
[oracle@ol7-122-rac1 ~]$
```

e. Use SQL*Plus to connect to the instances of the `cdb1` CDB, check the UNDO tablespaces and the groups of redo log files, and verify the existence of the `pdb1` PDB.

1) Check the UNDO tablespaces created in the CDB.

```
[oracle@ol7-122-rac1 ~]$ sqlplus / as sysdba

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 -
64bit Production
With the Partitioning, Real Application Clusters, Automatic
Storage Management, OLAP, Advanced Analytics and Real
Application Testing options

SQL> SELECT name, cdb, con_id FROM v$database;

NAME      CDB     CON_ID
--------- --- ----------
CDB1      YES          0

SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME        CON_ID
-------------------- --------------
cdb1_3                    0

SQL> show con_name

CON_NAME
------------------------------------------
CDB$ROOT
```

```
SQL> SELECT tablespace_name, con_id
     FROM   cdb_tablespaces
     WHERE  contents = 'UNDO';

TABLESPACE_NAME                          CON_ID
---------------------------------------- ---------------
UNDOTBS1                                       1
UNDOTBS2                                       1
UNDOTBS3                                       1

SQL>
```

2) Check the groups of redo log files created for the three CDB instances.

```
SQL> SELECT group#, con_id FROM   v$logfile;

    GROUP#     CON_ID
---------- ----------
         2          0
         1          0
         5          0
         6          0
         3          0
         4          0

6 rows selected.

SQL>
```

3) Check the PDB created in the CDB and its open mode. If the PDB is not opened, open it.

```
SQL> COL pdb_name format a10
SQL> SELECT pdb_id, pdb_name, guid, status FROM cdb_pdbs;

    PDB_ID PDB_NAME   GUID                             STATUS
---------- ---------- -------------------------------- ----------
         3 PDB1       E13E44A728D5266BE043650200C0187D NORMAL
         2 PDB$SEED   E13D83F6E4966F2AE043650200C0058C NORMAL

SQL> SELECT name, open_mode FROM v$pdbs;
```

```
NAME                                     OPEN_MODE
---------------------------------------- ----------
PDB$SEED                                 READ ONLY
PDB1                                     MOUNTED

SQL> ALTER SESSION SET CONTAINER=pdb1;

Session altered.

SQL> show con_name

CON_NAME
------------------------------------------
PDB1

SQL> CONNECT / AS SYSDBA
Connected.

SQL> SELECT name FROM cdb_services;

NAME
--------------------------------------------------------------------------------------------
SYS$BACKGROUND
SYS$USERS
cdb1XDB
cdb1

SQL> ALTER PLUGGABLE DATABASE pdb1 OPEN;

Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                                     OPEN_MODE
---------------------------------------- ----------
PDB$SEED                                 READ ONLY
PDB1                                     READ WRITE

SQL>
```

4) Check the services.

```
SQL> SELECT name FROM v$services;
```

```
NAME
--------------------------------------------------------------------------------
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

SQL> EXIT
```

f. Switch to the second node to verify the open mode of the PDB in the second instance of the CDB.

```
[oracle@ol7-122-rac1 ~]$ ssh
host02 [oracle@host02 ~]$ .
oraenv ORACLE_SID = [oracle] ?
cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host02 ~]$ sqlplus / as sysdba

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 -
64bit Production
With the Partitioning, Real Application Clusters, Automatic
Storage Management, OLAP, Advanced Analytics and Real
Application Testing options

SQL> SELECT name, cdb, con_id FROM v$database;

NAME      CDB     CON_ID
--------- --- ----------
CDB1      YES          0

SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME        CON_ID
---------------------- --------------
cdb1_2                      0

SQL> show con_name

CON_NAME
```

```
------------------------------------------
CDB$ROOT

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                           OPEN_MODE
------------------------------ ----------
PDB$SEED                       READ ONLY
PDB1                           MOUNTED

SQL> ALTER SESSION SET CONTAINER=pdb1;

Session altered.

SQL> SELECT name FROM v$services;

NAME
----------------------------------------------------------------------------------------------------
pdb1

SQL> exit
[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@ol7-122-rac1 ~]$
```

g. Verify that the `pdb1` service is accessible from instance `cdb1_3` on the first node but also from the `cdb1_2` instance on the second node and from the `cdb1_1` instance on the third node. First restart the listener.

```
[oracle@ol7-122-rac1 ~]$ su - grid
Password:
Last login: Mon Sep 2 05:16:31 2014 from ol7-122-rac1.example.com
[grid@ol7-122-rac1 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid [grid@ol7-

122-rac1 ~]$ srvctl stop listener -listener LISTENER

[grid@ol7-122-rac1 ~]$ srvctl start listener -listener

LISTENER

[grid@ol7-122-rac1 ~]$ exit
logout
```

```
[oracle@ol7-122-rac1 ~]$ sqlplus /nolog
SQL*Plus: Release 12.1.0.2.0 Production on Wed Jul 17 00:38:28
2014

Copyright (c) 1982, 2014, Oracle. All rights reserved.

SQL> CONNECT system@"ol7-122-rac1:1521/pdb1"
Enter password:
Connected.
SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME        CON_ID
---------------------- --------------
cdb1_3                      0

SQL> show con_name

CON_NAME
------------------------------------------
PDB1

SQL> CONNECT system@"host02:1521/pdb1"
Enter password:
ERROR:
ORA-01033:ORACLE initialization or shutdown in progress
Process ID: 0
Session ID: 0 Serial Number: 0

Warning: You are no longer connected to ORACLE.
SQL>
```

Notice that the connection does not complete because `pdb1` was opened for instance `cdb1_3` on `ol7-122-rac1` only. Remember that the clause `INSTANCES` was not used in the `ALTER PLUGGABLE DATABASE OPEN` statement in task 2.e.3).

```
SQL> CONNECT / AS SYSDBA
Connected.
SQL> ALTER PLUGGABLE DATABASE pdb1 OPEN INSTANCES=('cdb1_2');

Pluggable database altered.
```

```
SQL> CONNECT system@"host02:1521/pdb1"
```

```
Enter password:
Connected.

SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME        CON_ID
-------------------- --------------
cdb1_2                    0

SQL> show con_name

CON_NAME
------------------------------------------
PDB1

SQL> CONNECT system@"host03:1521/pdb1"
Enter password:
ERROR:
ORA-01033:ORACLE initialization or shutdown in progress
Process ID: 0
Session ID: 0 Serial Number: 0

Warning: You are no longer connected to ORACLE.

SQL> EXIT
```

The connection does not complete on host03 because `pdb1` was opened for instance `cdb1_3` on `ol7-122-rac1` and `cdb1_2` on `host02` only.

## Practice 12-2: Cloning a PDB in the RAC CDB

### Overview

In this practice, you will clone the `pdb1` PDB into a new PDB named `pdb2` in the `cdb1` CDB. This operation requires to close and open PDBs on multiple instances of the CDB.

### Tasks

1. Connect to the root of the multitenant container database `cdb1` on any of the three instances.

```
[oracle@ol7-122-rac1 ~]$ . oraenv
ORACLE_SID = [cdb1] ? cdb1_3
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base has been set to /u01/app/oracle
[oracle@ol7-122-rac1 ~]$ sqlplus / as sysdba


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 -
64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real
Application Testing options


SQL>
```

2. Use Oracle Managed Files to locate the data files of the new `pdb2`.

```
SQL> SHOW PARAMETER db_create_file_dest
NAME                                 TYPE        VALUE
------------------------------------ ----------- ----------------------
db_create_file_dest                  string      +DATA


SQL>
```

3. Create `pdb2` from `pdb1`.
   a. Use the `CREATE PLUGGABLE DATABASE` command to create `pdb2`.

```
SQL> CREATE PLUGGABLE DATABASE pdb2 FROM pdb1;


Pluggable database created.


SQL> SELECT name, open_mode FROM v$pdbs;


NAME                           OPEN_MODE
```

```
---------------------------------------- ----------
PDB$SEED                                 READ ONLY
PDB1                                     READ WRITE
PDB2                                     MOUNTED

SQL>
```

b. Now, open both PDBs in READ WRITE mode on all the CDB instances.

```
SQL> ALTER PLUGGABLE DATABASE pdb1 CLOSE IMMEDIATE
INSTANCES=ALL;

Pluggable database altered.

SQL> ALTER PLUGGABLE DATABASE ALL OPEN READ WRITE INSTANCES=ALL;

Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                                     OPEN_MODE
---------------------------------------- ----------
PDB$SEED                                 READ ONLY
PDB1                                     READ WRITE
PDB2                                     READ WRITE

SQL> SELECT pdb_id, pdb_name, guid, status FROM cdb_pdbs;

    PDB_ID PDB_NAME   GUID                             STATUS
---------- ---------- -------------------------------- ----------
         3 PDB1       E13E44A728D5266BE043650200C0187D NORMAL
         2 PDB$SEED   E13D83F6E4966F2AE043650200C0058C NORMAL
         4 PDB2       E2B1483E90856557E043650200C01D40 NORMAL

SQL> ALTER SESSION SET CONTAINER=pdb2;

Session altered.

SQL> SELECT name FROM dba_services;

NAME
----------------------------------------------------------------------------------------------------
pdb2
```

```
SQL> EXIT
```

c. Use LSNRCTL to verify that the new `pdb2` service associated to the new PDB in the CDB instance is automatically started after the PDB is opened. Because the PDB is opened in all the CDB instances, the `pdb2` PDB service is started in all the CDB instances.

```
[oracle@ol7-122-rac1 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 30-JUL-
2014 05:44:45

Copyright (c) 1991, 2014, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0
- Production
Start Date                16-JUL-2014 05:38:51
Uptime                    14 days 0 hr. 5 min. 55 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listening Endpoints Summary...
```

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.101)(PORT=1521
)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.239)(PORT=1521
)))
Services Summary...

Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 2 handler(s) for this
service...

Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
```

```
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb2" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully

[oracle@ol7-122-rac1 ~]$
```

4. Use the net service name to connect to `pdb2` as the `system` user on any of the three instances of the CDB.

```
[oracle@ol7-122-rac1 ~]$  sqlplus
/nolog SQL> CONNECT system@"ol7-122-
rac1:1521/pdb2"Enter password:
Connected.

SQL> show con_name
CON_NAME
------------------------------------------
PDB2

SQL> CONNECT system@"host02:1521/pdb2"
Enter password:
Connected.

SQL> CONNECT system@"host03:1521/pdb2"
Enter password:
Connected.

SQL> EXIT
```

## Practice 12-3: Affinitizing PDB Services to CDB Instances

### Overview

In this practice, you will “affinitize” connections to a PDB to one or particular CDB instances. Because server pools determine which services run together or separately, you can configure and maintain required affinity or isolation.

### Tasks

1. Create a dynamic PDB service, `mypdb1serv`, for the `pdb1` PDB in the CDB which will “affinitize” connections to `pdb1` to all the CDB instances.

   a. Check the configuration of the server pools.

```
[oracle@ol7-122-rac1 ~]$ srvctl status srvpool
Server pool name: Free
Active servers count: 0
Server pool name: Generic
Active servers count: 0
Server pool name: cdb1pool
Active servers count: 3
[oracle@ol7-122-rac1 ~]$

[oracle@ol7-122-rac1 ~]$ srvctl config srvpool -serverpool
cdb1pool
Server pool name: cdb1pool
Importance: 0, Min: 0, Max: 3
Category: hub
Candidate server names:
[oracle@ol7-122-rac1 ~]$
```

   b. Check the services. You notice that the default services created at PDB creation are not managed by the clusterware.

```
[oracle@ol7-122-rac1 ~]$ srvctl status service -db cdb1
[oracle@ol7-122-rac1 ~]$
[oracle@ol7-122-rac1 ~]$ srvctl config service -db cdb1
[oracle@ol7-122-rac1 ~]$
```

   c. Create a dynamic PDB service for the `pdb1` PDB in the CDB which will “affinitize” connections to `pdb1` to all the CDB instances uniformly.

      1) Create the service from the connection on the first node of the server pool.

```
[oracle@ol7-122-rac1 ~]$ srvctl add service -db cdb1 -pdb
pdb1 - service mypdb1serv -policy automatic -serverpool
cdb1pool -cardinality uniform
```

```
[oracle@ol7-122-rac1 ~]$ srvctl config service -db cdb1
Service name: mypdb1serv
Service is enabled
Server pool: cdb1pool
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:
[oracle@ol7-122-rac1 ~]$
```

2) Check that the PDB service is also created on the two other nodes of the server pool.

```
[oracle@ol7-122-rac1 ~]$ ssh
host02 [oracle@host02 ~]$
[oracle@host02 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base has been set to /u01/app/oracle
```

```
[oracle@host02 ~]$ srvctl config service -db cdb1
```

```
Service name: mypdb1serv
Service is enabled
Server pool: cdb1pool
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:
[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@ol7-122-rac1 ~]$
```

You can reiterate the same verification on the third node of the server pool.

```
[oracle@ol7-122-rac1 ~]$ ssh
host03 [oracle@host03 ~]$ .
oraenv ORACLE_SID = [oracle] ?
cdb1_1
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host03 ~]$ srvctl config service -db cdb1
Service name: mypdb1serv
```

```
Service is enabled
Server pool: cdb1pool
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:
[oracle@host03 ~]$ exit
Logout
Connection to host03 closed.

[oracle@ol7-122-rac1 ~]$
```

d. Close the PDB. You will verify that restarting the CDB automatically starts the dynamic PDB service and opens the associated PDB.

```
[oracle@ol7-122-rac1 ~]$ sqlplus / as sysdba


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 -
64bit Production
With the Partitioning, Real Application Clusters, Automatic
Storage Management, OLAP, Advanced Analytics and Real
Application Testing options
```

```
SQL> ALTER PLUGGABLE DATABASE pdb1 CLOSE IMMEDIATE
INSTANCES=ALL;
Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                                     OPEN_MODE
---------------------------------------- ----------
PDB$SEED                                 READ ONLY
PDB1                                     MOUNTED
PDB2                                     READ WRITE

SQL> EXIT
```

e. Stop and restart the CDB.

```
[oracle@ol7-122-rac1 ~]$ srvctl stop database -d cdb1
[oracle@ol7-122-rac1 ~]$
[oracle@ol7-122-rac1 ~]$ srvctl start database -db cdb1 -eval
Database cdb1 will be started on nodes host03,host02,ol7-122-
rac1 Service mypdb1serv will be started on nodes
host03,host02,ol7-122-rac1[oracle@ol7-122-rac1 ~]$
[oracle@ol7-122-rac1 ~]$ srvctl start database -db cdb1

[oracle@ol7-122-rac1 ~]$ srvctl status database -db cdb1
Instance cdb1_1 is running on node host03
Instance cdb1_2 is running on node host02
Instance cdb1_3 is running on node ol7-
122-rac1
[oracle@ol7-122-rac1 ~]$
```

f. Verify that the new dynamic PDB service is started and the PDB opened automatically. It may take a few moments for the service to be started on all hosts. (If you do not wish to wait, use SRVCTL to start the service: `srvctl start service -d cdb1 -service mypdb1serv`).

```
[oracle@ol7-122-rac1 ~]$ srvctl status service -db cdb1
Service mypdb1serv is running on nodes: host03,host02,ol7-

122-rac1.[oracle@ol7-122-rac1 ~]$ sqlplus / as sysdba

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 -
64bit Production
```

```
With the Partitioning, Real Application Clusters, Automatic
Storage Management, OLAP,
Advanced Analytics and Real Application Testing options

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                           OPEN_MODE
------------------------------ --------------
PDB$SEED                       READ ONLY
PDB1                           READ WRITE
PDB2                           MOUNTED

SQL> SELECT name FROM v$services;

NAME
----------------------------------------------------------------------------------------------------
mypdb1serv
pdb2
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

7 rows selected.

SQL> EXIT
[oracle@ol7-122-rac1 ~]$
```

Notice that PDBs are automatically opened by clusterware in all the instances in which the service is started. There is therefore no need to create a trigger `AFTER STARTUP ON DATABASE` to open PDBs as it is the case in non-RAC CDBs.

g. You can also stop and restart the service manually.

```
[oracle@ol7-122-rac1 ~]$ srvctl stop service -d cdb1 -
servicemypdb1serv

[oracle@ol7-122-rac1 ~]$ srvctl status service -d cdb1 -
servicemypdb1serv
Service mypdb1serv is not running.

[oracle@ol7-122-rac1 ~]$ srvctl start service -d cdb1 -
servicemypdb1serv
```

```
[oracle@ol7-122-rac1 ~]$ srvctl status service -d cdb1 -
servicemypdb1serv
Service mypdb1serv is running on nodes: host03,host02,ol7-

122-rac1[oracle@ol7-122-rac1 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 17-JUL-
2014 06:07:37

Copyright (c) 1991, 2014, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0
- Production
Start Date                16-JUL-2014 05:39:02
Uptime                    1 days 0 hr. 28 min. 37 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
/u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File
/u01/app/grid/diag/tnslsnr/ol7-122-
rac1/listener/alert/log.xmlListening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.101)(PORT=1521
)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.239)(PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 2 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "mypdb1serv" has 1 instance(s).
```

```
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb2" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully
[oracle@ol7-122-rac1 ~]$
```

h. Use the service to connect to the PDB on any of the CDB instances.

```
[oracle@ol7-122-rac1 ~]$ sqlplus /nolog

SQL> CONNECT system@"ol7-122-
rac1/mypdb1serv"Enter password:
Connected.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                           OPEN_MODE
------------------------------ ----------
PDB1                           READ WRITE

SQL> CONNECT system@"host02/mypdb1serv"
Enter password:
Connected.

SQL> SELECT name FROM v$services;

NAME
----------------------------------------------------------------------------------------------------
mypdb1serv
pdb1

SQL> CONNECT system@"host03/mypdb1serv"
Enter password:
Connected.

SQL> SELECT name FROM v$services;

NAME
```

```
--------------------------------------------------------------------------------------------
mypdb1serv
pdb1

SQL> EXIT
```

2. You can also “affinitize” connections to `pdb2` to a single node by defining the `singpdb2serv` service cardinality to `SINGLETON`.

   a. Create and start the service for `pdb2`.

```
[oracle@ol7-122-rac1 ~]$ srvctl add service -db cdb1 -pdb
pdb2 - service singpdb2serv -policy automatic -serverpool
cdb1pool -cardinality singleton

[oracle@ol7-122-rac1 ~]$ srvctl start service -d cdb1 -
servicesingpdb2serv -eval
Service singpdb2serv will be started on node ol7-122-rac1

[oracle@ol7-122-rac1 ~]$ srvctl start service -d cdb1 -
servicesingpdb2serv

[oracle@ol7-122-rac1 ~]$ srvctl status service -d cdb1 -
servicesingpdb2serv
Service singpdb2serv is running on nodes: ol7-
122-rac1[oracle@ol7-122-rac1 ~]$
```

   b. Check that you can use the service to connect to `pdb2` only on `ol7-122-rac1` and that thePDB is opened in the CDB instance on `ol7-122-rac1` only.

```
[oracle@ol7-122-rac1 ~]$ sqlplus /nolog

SQL> CONNECT system@"ol7-122-
rac1/singpdb2serv"Enter password:
Connected.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                           OPEN_MODE
------------------------------ ----------
PDB2                           READ WRITE

SQL> SELECT name FROM v$services;

NAME
```

```
-------------------------------------------------------------------------------------------
singpdb2serv
pdb2

SQL>
```

c. Check that you cannot use the service to connect to `pdb2` on `host02` nor `host03` and that the PDB is closed in the CDB instances on `host02` and `host03`.

```
SQL> CONNECT system@"host02/singpdb2serv"
Enter password:
ERROR:
ORA-12514: TNS:listener does not currently know of service
requested in connect descriptor

Warning: You are no longer connected to ORACLE.

SQL> CONNECT system@"host03/singpdb2serv"
Enter password:
ERROR:
ORA-12514: TNS:listener does not currently know of service
requested in connect descriptor

Warning: You are no longer connected to ORACLE.

SQL> CONNECT system@"host02/cdb1"
Enter password:
Connected.

SQL> select name, open_mode from v$pdbs;

NAME                           OPEN_MODE
------------------------------ --------------
PDB$SEED                       READ ONLY
PDB1                           READ WRITE
PDB2                           MOUNTED

SQL> CONNECT system@"host03/cdb1"
Enter password:
Connected.

SQL> select name, open_mode from v$pdbs;
```

## Practice 12-4: Dropping a PDB

### Overview

In this practice, you will drop a PDB in the CDB and verify that the services and data files are deleted.

### Tasks

1. To drop the pdb2, first stop and remove the service from the resources configuration.

```
[oracle@ol7-122-rac1 ~]$ srvctl stop service -d cdb1 -
servicesingpdb2serv

[oracle@ol7-122-rac1 ~]$ srvctl remove service -d cdb1 -
servicesingpdb2serv
[oracle@ol7-122-rac1 ~]$
```

2. Drop the pdb2 PDB.

```
[oracle@ol7-122-rac1 ~]$ ~]$ sqlplus /nolog

SQL> CONNECT system@"ol7-122-
rac1/pdb2"Enter password:
Connected.

SQL> SELECT name FROM v$datafile;

NAME
--------------------------------------------------------------------------------
----------------------
+DATA/CDB1/DATAFILE/undotbs2.294.825668383
+DATA/CDB1/C45A345T5F09726D9C25F01AZ04366B8/DATAFILE/system.268.
335670735
+DATA/CDB1/C45A345T5F09726D9C25F01AZ04366B8/DATAFILE/sysaux.273.
335670729
+DATA/CDB1/C45A345T5F09726D9C25F01AZ04366B8/DATAFILE/users.282.3
35671601

SQL> CONNECT / AS SYSDBA
Connected.

SQL> SELECT name FROM v$services;

NAME
----------------------------------------------------------------------------------
mypdb1serv
pdb2
```

```
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

7 rows selected.

SQL> DROP PLUGGABLE DATABASE pdb2 INCLUDING DATAFILES;
*
ERROR at line 1:
ORA-65025: Pluggable database PDB2 is not closed on all
instances.

SQL> ALTER PLUGGABLE DATABASE pdb2 CLOSE INSTANCES=ALL;

Pluggable database altered.

SQL> DROP PLUGGABLE DATABASE pdb2 INCLUDING DATAFILES;

Pluggable database dropped.

SQL>
```

3. Verify that the data files are deleted.

```
SQL> SELECT name FROM v$datafile;

NAME
--------------------------------------------------------------------------------
-------------------------
+DATA/CDB1/DATAFILE/system.285.825666373
+DATA/CDB1/DATAFILE/sysaux.287.825666251
+DATA/CDB1/DATAFILE/undotbs1.283.825666541
+DATA/CDB1/DD7C48AA5A4404A2E04325AAE80A403C/DATAFILE/system.286.
825666707
+DATA/CDB1/DATAFILE/users.271.825666537
+DATA/CDB1/DD7C48AA5A4404A2E04325AAE80A403C/DATAFILE/sysaux.284.
825666703
+DATA/CDB1/DATAFILE/undotbs2.294.825668383
+DATA/CDB1/DATAFILE/undotbs3.279.825668393
+DATA/CDB1/E5F09726D9C25FC4E043660200C075A9/DATAFILE/system.268.
825670735
```

```
+DATA/CDB1/E5F09726D9C25FC4E043660200C075A9/DATAFILE/sysaux.273.
825670729
+DATA/CDB1/E5F09726D9C25FC4E043660200C075A9/DATAFILE/users.282.8
25671601

11 rows selected.

SQL>
```

Note that all files related to `pdb2` are removed. The UNDO datafile is associated with the instance, and not with any PDB.

4. Verify that the services are deleted. Check in `V$SERVICES` view and with LSNRCTL.

```
SQL> SELECT name FROM v$services;

NAME
----------------------------------------------------------------------------------------------------
mypdb1serv
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

6 rows selected.

SQL> EXIT
[oracle@ol7-122-rac1 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 17-JUL-
2014 06:07:37

Copyright (c) 1991, 2014, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0
- Production
Start Date                16-JUL-2014 05:39:02
Uptime                    1 days 0 hr. 28 min. 37 sec
Trace Level               off
```

```
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
/u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File
/u01/app/grid/diag/tnslsnr/ol7-122-
rac1/listener/alert/log.xmlListening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.101)(PORT=1521
)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.239)(PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
Instance "+ASM1", status READY, has 2 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "mypdb1serv" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully
[oracle@ol7-122-rac1 ~]$
```

5. Close all terminal windows opened for this practice.