Practice

Performing Basic CDB Administration Tasks

Practice Target

In this practice, you will perform some basic CDB administration tasks.

Specifically, in this practice you will perform the following:

- Change the states of the CDB and the PDBs.
- Modify the PDB restart states.
- Modify a parameter in a PDB
- Change the Global Database Name of a PDB

Changing the States of the CDB and the PDBs

In the following steps, you will shut down and start up PDBs once after logging on to the root and then after switching current container to the PDB. You will also shut down and start up the CDB.

A. Changing the state of a specific PDB when the current container is the root

1. Create a Putty session to srv1 and login as oracle user. Then login to the CDB as sysdba in SQL*Plus.

```
sqlplus / as sysdba
```

2. Verify that you are connected to the root.

   The root container always has its CON_ID equals to 1 and its name is always CDB$ROOT

   Observe that we can pass two variables to the SHOW command in the same line.

```
show CON_ID CON_NAME
```

3. Check the state of PDB1. If the PDB is closed, start it up.

```
SELECT OPEN_MODE FROM V$PDBS WHERE NAME='PDB1';
```

4. Shutdown PDB1

   Observe that IMMEDIATE option can be passed to the command. It has the same effect as the IMMEDIATE option in the SHUTDOWN command.

   Shutting down a PDB is much faster than shutting down the CDB.

```
ALTER PLUGGABLE DATABASE pdb1 CLOSE IMMEDIATE;
```

5. Check the current state of PDB1.

   The view displays its state as MOUNTED. In the context of PDBs, MOUNTED state means closed.

```
SELECT OPEN_MODE FROM V$PDBS WHERE NAME='PDB1';
```

6. Startup PDB1. Any of the following commands gives the same

   result. Starting up a PDB is way faster than starting up a CDB.

   ALTER PLUGGABLE DATABASE is a SQL statement. STARTUP is a SQL*Plus command.

```
-- execute one of the following commands:
ALTER PLUGGABLE DATABASE pdb1 OPEN;
STARTUP PLUGGABLE DATABASE pdb1
```

B. Changing the State of all the PDBs

In the following steps you will learn how to close all the PDBs in one single statement. You will then start them all up.

7. Close all the PDBs

```
ALTER PLUGGABLE DATABASE ALL CLOSE;
```

8. Check the state of all the PDBs.

Observe that the seed PDB has not been affected by the close command.

```
col name format a10
SELECT NAME, OPEN_MODE FROM V$PDBS ORDER BY 1;
```

9. Start up all the PDBs.

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

C. Changing the State of the CDB

In the following steps you will shutdown and then start up the entire CDB.

10. Shutdown the entire CDB.

The entire CDB is shutdown because the current session is connected to the root container.

```
SHUTDOWN IMMEDIATE
```

11. Start up the CDB.

OPEN is the default option, therefore, it is optional.

```
STARTUP
```

12. Check the states of the PDBs.

Observe that all the user-created PDBs are closed.

```
SELECT NAME, OPEN_MODE FROM V$PDBS ORDER BY 1;
```

D. Changing the PDB Default Restart States

In this section you will change the PDB default restart states for all the user PDBs.

13. Startup all the PDBs

The statement can be applied on specific PDB.

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

14. Save the PDB states.

This statement accepts specific PDB as well.

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

15. Verify the PDB saved states

```
col con_name format a10
SELECT CON_NAME, STATE FROM CDB_PDB_SAVED_STATES;
```

16. Test the changes by restarting the CDB and checking the state of the PDBs.

```
SHUTDOWN IMMEDIATE
STARTUP OPEN
SELECT NAME, OPEN_MODE FROM V$PDBS ORDER BY 1;
```

E. Changing the state of a Specific PDB when the Current Container is that PDB

17. Switch the current container to PDB1

```
ALTER SESSION SET CONTAINER=PDB1;
```

18. Verify that the current container is PDB1

```
SHOW CON_ID CON_NAME
```

19. Shutdown the container then start it up.

I would rather use the ALTER PLUGGABLE DATABASE command to startup and shutdown a PDB because if the current container is the root and I am not aware about that, the SHUTDOWN command will shut down the entire CDB. Using the ALTER PLUGGABLE DATABASE command to shut down or startup the root is not possible.

```
SHUTDOWN IMMEDIATE
STARTUP OPEN
```

20. Set the current container back to the root.

```
ALTER SESSION SET CONTAINER=CDB$ROOT;
```

Modifying a Parameter in a PDB

In the following section, you will examine changing the parameters at the CDB level and at the PDB level.

F. Modifying a non-PDB-modifiable Parameter

21. Verify that the parameter DB_RECOVERY_FILE_DEST_SIZE cannot be changed in the PDB

level. The ISPDB_MODIFIABLE is FALSE which means it cannot be modified at the PDB level.

Observe that the values of the NAME column in V$SYSTEM_PARAMETER are in lower-case.

```
col value format a15

SELECT VALUE, VALUE/1024/1024/1024 GB, ISPDB_MODIFIABLE
FROM   V$SYSTEM_PARAMETER
WHERE NAME='db_recovery_file_dest_size';
```

22. Change the current container to PDB1, query the value of this parameter, and try to change its value.

```
ALTER SESSION SET CONTAINER=PDB1;
SELECT VALUE FROM V$SYSTEM_PARAMETER WHERE NAME='db_recovery_file_dest_size';
ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE = 10903094248;
```
You should receive the following error:

```
ORA-65040: operation not allowed from within a pluggable database
```

23. Change the current container to the root, and try to change the parameter value.

The parameter value could successfully be changed at the CDB level.

```
ALTER SESSION SET CONTAINER=CDB$ROOT;
ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE = 10903094248 SCOPE=BOTH;
```

24. Change the current container to PDB1 and query the value of the

parameter. The change made at the CDB level is seen by the PDB.

```
ALTER SESSION SET CONTAINER=PDB1;
SELECT VALUE, VALUE/1024/1024/1024 GB FROM V$SYSTEM_PARAMETER WHERE
NAME='db_recovery_file_dest_size';
```

25. Set the current container back to the root.

```
ALTER SESSION SET CONTAINER=CDB$ROOT;
```

G. Modifying a PDB-modifiable Parameter

26. Verify that the parameter DDL_LOCK_TIMEOUT can be changed in the PDB level.

The ISPDB_MODIFIABLE is TRUE, which means the parameter can be modified at the PDB level.

```
col VALUE for a10
SELECT VALUE, ISPDB_MODIFIABLE
FROM V$SYSTEM_PARAMETER
WHERE NAME='ddl_lock_timeout';
```

27. Change the current container to PDB1 and try to change the parameter

value. Changing this parameter at the PDB level succeeds

```
ALTER SESSION SET CONTAINER=PDB1;
ALTER SYSTEM SET DDL_LOCK_TIMEOUT = 12;
```

28. Set the current container back to the root and query information about the parameter.

The parameter has different value for each PDB.

```
ALTER SESSION SET CONTAINER=CDB$ROOT;
col name format a20 SELECT
CON_ID, NAME, VALUE
  FROM V$SYSTEM_PARAMETER
 WHERE NAME='ddl_lock_timeout';
```
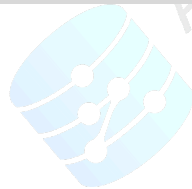
Changing the Global Database Name of a PDB

In the following section, you will examine change the global database name of a PDB. You will create a new PDB, rename it, and finally drop it from the CDB.

H. Creating a new PDB from the seed

29. Create a new pluggable database from the seed.

```
CREATE PLUGGABLE DATABASE pdb_test
 ADMIN USER pdbtestadmin IDENTIFIED BY ABcd##1234;
ALTER PLUGGABLE DATABASE pdb_test OPEN;
```

I. Changing the Global Database Name of a PDB

30. Close the PDB and open it in the restricted mode.

```
ALTER PLUGGABLE DATABASE pdb_test CLOSE IMMEDIATE;
ALTER PLUGGABLE DATABASE pdb_test OPEN RESTRICTED;
SELECT CON_ID, OPEN_MODE, RESTRICTED FROM V$PDBS WHERE NAME='PDB_TEST';
```

31. Change the global database name for pdb_test to PDB2

```
ALTER SESSION SET CONTAINER=PDB_TEST;
ALTER PLUGGABLE DATABASE pdb_test RENAME GLOBAL_NAME to pdb2;
SELECT CON_ID, OPEN_MODE, RESTRICTED FROM V$PDBS WHERE
NAME='PDB2'; ALTER PLUGGABLE DATABASE pdb2 CLOSE IMMEDIATE;
ALTER PLUGGABLE DATABASE pdb2 OPEN;
SELECT CON_ID, OPEN_MODE, RESTRICTED FROM V$PDBS WHERE NAME='PDB2';
```

32. Test connecting to the renamed PDB

```
conn system/ABcd##1234@//srv1:1521/pdb2.localdomain
```

Clean up

33. Drop the new PDB, including its datafiles.

```
conn / as sysdba
ALTER PLUGGABLE DATABASE pdb2 CLOSE IMMEDIATE;
DROP PLUGGABLE DATABASE pdb2 INCLUDING DATAFILES;

col name format a10
SELECT NAME, CON_ID, OPEN_MODE, RESTRICTED FROM V$PDBS ORDER BY 1;
```

Summary

- When you connect to a root, you can change the state of any PDB. When the current container is a PDB, you can change the state of only the current PDB.

- PDB state can be saved. The next time you start the CDB, the PDB will be in the saved state.

- Some parameters can be modified in the PDB level as well as the CDB level and some parameter can be modified only in the CDB level.

- To change the Global Database Name of a PDB, the PDB must be opened in restricted mode.