Basic Administration

Basics about CDB and PDB

Database log files:

In oracle 12c there is no changes has been done for database log file.
Example

select * from v$diag_info;

INST_ID NAME VALUE CON_ID
---------- ---------- ---------- ------
1 Diag Enabled TRUE 0
1 ADR Base /u02 0
1 ADR Home /u02/diag/rdbms/asrblg1 0
1 Diag Trace /u02/diag/rdbms/asrblg/asrblg1/trace 0
1 Diag Alert /u02/diag/rdbms/asrblg/asrblg1/alert 0
1 Diag Incident /u02/diag/rdbms/asrblg/asrblg1/incident 0
1 Diag Cdump /u02/diag/rdbms/asrblg/asrblg1/cdump 0
1 Health Monitor /u02/diag/rdbms/asrblg/asrblg1/hm 0
1 Default Trace File /u02/diag/rdbms/asrblg/asrblg1/trace/asrblg1_ora_80103.trc 0
1 Active Problem Count 0 0
1 Active Incident Count 0 0

11 rows selected.

Grid log files and location:

From 12.1 onwards all the Grid log files has centralized and move to ADR in $ORACLE_BASE/diag/crs/

oracle@localhost$: /u01/oracle/product/diag/crs/ctrsnydb11/crs
asrblg1: ls -lthr
total 472K
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 sweep
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 metadata_pv
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 metadata_dgif
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 incpkg
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 incident
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 stage
drwxrwxr-x 4 oracle oinstall 4.0K Feb 20 2019 log
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 cdump
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 metadata
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 lck
drwxrwxr-x 2 oracle oinstall 4.0K Feb 20 2019 alert
drwxrwxr-x 2 oracle oinstall 424K Aug 26 00:02 trace

00:15:18 asrblg1: pwd
/u01/oracle/product/diag/crs/asrblg/crs/trace
oracle@localhost$: /u01/oracle/product/diag/crs/ asrblg /crs/trace

```
00:15:27 asrblg1: ls -lthr alert*
-rw-rw---- 1 oracle oinstall 156K Aug 25 21:14 alert.log ### Cluster alert log file
```

User management :

user management has been classified into 5 categories. They are:

- Common & Local Users.
- Privilege management
- Lock down profile
- Oracle-maintained users password change.
- Common user password change.

From 12c onwards we have 2 types of database users are available.
They are:

- Common user
- Local user

https://asrblogger.com/common-user-vs-local-user-in-12c/

Privilege Management:

Privilege management has been classified into 2 sub divisions. They are:

- Grants
- Roles

Grants:

oracle12c have two variant of grants.
They are:

- Common grants
- Local grants

https://asrblogger.com/common-grant-vs-local-grant/

Roles:

Collection of privileges called role.

In 12c we have 2 different types of role. They are:

- Common role
- Local role

https://asrblogger.com/common-roles-vs-local-roles/

Oracle-maintained users password change

https://asrblogger.com/common-user-password-change/

Lockdown Profile:

From 12c onwards we have new feature called lockdown profile to control the PDB level activities. It can be created in CDB container.

https://asrblogger.com/lockdown-profile/

Database storage and structure

Managing Tablespace in CDB and PDB

https://asrblogger.com/managing-tablespace-in-cdb-and-pdb/

Renaming and Relocating Datafiles Online

https://asrblogger.com/renaming-and-relocating-datafiles-online/

Undo Tablespace

https://asrblogger.com/undo-tablespace/

https://asrblogger.com/doooooooonttt-publlishhundo-rentention-time-with-autoextend-on-and-autoextend-off/

https://asrblogger.com/undo-tablespace-ora-01555-snapshot-too-old-error/

https://asrblogger.com/undo-block/

Data dictionary and performance views

https://asrblogger.com/dynamic-performance-views-in-oracle/

```
SQL> select name, cdb, con_id from v$database;

NAME      CDB     CON_ID
--------- --- ----------
DEV       YES          0
```

```
set lines 300 pages 300
col NAME format a30
select dbid,name,open_mode,TOTAL_SIZE/1024/1024 from v$pdbs;

      DBID NAME                           OPEN_MODE  TOTAL_SIZE/1024/1024
---------- ------------------------------ ---------- --------------------
1238907645 PDB$SEED                       READ ONLY                   498
1982736540 DEV                            READ WRITE                91872



SQL> show pdbs

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



SQL> show con_name

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



SQL> select sys_context('USERENV','CON_NAME') FROM DUAL;


SQL> SELECT CDB FROM V$DATABASE;

CDB
---
YES
```

Viewing Information About Containers: -

```
col PDB_NAME format a30

select pdb_id, pdb_name, status from cdb_pdbs;


    PDB_ID PDB_NAME                       STATUS
---------- ------------------------------ ----------
         2 PDB$SEED                       NORMAL
         4 DEV                            NORMAL
```

Finding the Open Mode of a PDB:

```
col name format a20

select name, open_mode, restricted, open_time from v$pdbs;


NAME                 OPEN_MODE  RES OPEN_TIME
-------------------- ---------- --- ----------------------------------------------------------
PDB$SEED             READ ONLY  NO  29-JUL-19 11.03.59.387 AM -05:00
DEV                  READ WRITE NO  29-JUL-19 11.04.01.994 AM -05:00
```

Viewing the Open Mode of Each PDB:

```
COLUMN NAME FORMAT A15

COLUMN RESTRICTED FORMAT A10

COLUMN OPEN_TIME FORMAT A40

SQL> SELECT NAME, OPEN_MODE, RESTRICTED, OPEN_TIME FROM V$PDBS;


NAME            OPEN_MODE  RESTRICTED OPEN_TIME
--------------- ---------- ---------- --------------------------------
PDB$SEED        READ ONLY  NO         29-JUL-19 11.03.59.387 AM -05:00
DEV18           READ WRITE NO         29-JUL-19 11.04.01.994 AM -05:00
```

Password File management in 12C

https://asrblogger.com/password-file-management-in-12c-19c/

PDB$SEED

What is PDB seed?

→ The seed PDB is a system-supplied template that the CDB can use to create new
PDB$SEED.

What is the use ?

Below are the methods for creating a new pluggable database. PDB$SEED is useful while creating pluggable database by copying files from the seed database.

Remaining methods of pluggable creation, PDB$SEED are not useful.

Creating pluggable database from seed, when pdb$seed in in mount state and open state.

https://asrblogger.com/creating-pluggable-database-from-seed-when-pdbseed-in-in-mount-state-and-open-state/

Creating pluggable database from another pdb, when pdb$seed in in mount state

https://asrblogger.com/creating-pluggable-database-from-another-pdb-when-pdbseed-in-in-mount-state/