

PDB Snapshot Carousel - Oracle Database 18c

Por Francisco Riccio

Introducción

PDB Snapshot Carousel es un nuevo feature de Oracle Database 18c que nos permite realizar copias instantáneas “snapshot” de nuestros Pluggable Database (PDB) y mantenerlos en una bitácora externa con la finalidad de usarlo en los siguientes escenarios:

- Generar ambientes no productivos.
- Recuperación de un PDB Productivo ante un problema.

Esta versión nos permite mantener hasta un máximo de 8 snapshots por PDB, una vez alcanzado este número, estos van sobre escribiéndose conforme vayan requiriéndose nuevas copias instantáneas. Los snapshots pueden ser configurados de manera automática (se ejecuta cada cierto tiempo) como manual.

Los snapshots incluyen la copia de los data files del PDB original, excluyendo los redo logs. Esta copia instantánea se almacena en disco y por defecto se encuentra en el mismo directorio del PDB.

Estos snapshots son distintos a los snapshots copy PDB, debido a que este último no realiza una copia completa de los data files y siempre dependerá del CDB donde se aloje.

Implementación

a. Obtener la configuración de Snapshot de un PDB

Para conocer la cantidad de snapshots soportados por PDB consultaremos las vistas: CDB_PROPERTIES y CDB_PDBS.

```
col property_value format a15
select p.pdb_name, property_name, property_value
from cdb_properties r join cdb_pdbbs p
on r.con_id = p.con_id
where property_name = 'MAX_PDB_SNAPSHOTS';
```

```
SQL> select p.pdb_name, property_name, property_value
       from cdb_properties r join cdb_pdbbs p
       on r.con_id = p.con_id
       where property_name = 'MAX_PDB_SNAPSHOTS';
```

PDB_NAME	PROPERTY_NAME	PROPERTY_VALUE
PDB1	MAX_PDB_SNAPSHOTS	8

Podemos apreciar que el PDB (PDB1), está configurado para soportar hasta 8 snapshots.

b. Snapshot Manual

A continuación se ejecutará un snapshot de manera manual:

```
alter pluggable database snapshot;  
SQL> alter session set container=PDB1;  
  
Session altered.  
  
SQL> alter pluggable database snapshot;  
  
Pluggable database altered.
```

Se procederá a validar los snapshots disponibles:

```
set linesize 150  
col CON_NAME FORMAT a10  
col SNAPSHOT_NAME FORMAT a25  
col SNAP_SCN FORMAT 9999999  
col FULL_SNAPSHOT_PATH FORMAT a45  
SELECT CON_NAME, SNAPSHOT_NAME, FULL_SNAPSHOT_PATH  
FROM DBA_PDB_SNAPSHOTS;  
  
SQL> set linesize 150  
SQL> col CON_NAME FORMAT a10  
SQL> col SNAPSHOT_NAME FORMAT a25  
SQL> col SNAP_SCN FORMAT 9999999  
SQL> col FULL_SNAPSHOT_PATH FORMAT a45  
SQL> SELECT CON_ID, CON_NAME, SNAPSHOT_NAME,  
SNAPSHOT_SCN AS snap_scn, FULL_SNAPSHOT_PATH  
FROM DBA_PDB_SNAPSHOTS  
ORDER BY SNAP_SCN;  
  
CON_ID CON_NAME SNAPSHOT_NAME SNAP_SCN FULL_SNAPSHOT_PATH  
-----  
4 PDB1 SNAP_953306977_973136888 3076479 /u02/app/oracle/oradata/snap_953306977_307647  
9.pdb
```

Los snapshots tienen el siguiente formato como nombre de archivo: snap_<ID>_<SCN>.pdb.

Es posible crear un snapshot con un nombre específico como a continuación se presenta:

```
alter pluggable database snapshot <nombre_snapshot>;  
SQL> alter pluggable database snapshot FOTO1_PDB1;  
  
Pluggable database altered.  
  
SQL> alter pluggable database snapshot FOTO2_PDB1;  
  
Pluggable database altered.
```

Se validará los snapshots generados hasta este momento:

```
SQL> SELECT CON_NAME, SNAPSHOT_NAME, FULL_SNAPSHOT_PATH
FROM DBA_PDB_SNAPSHOTS;
```

CON_NAME	SNAPSHOT_NAME	FULL_SNAPSHOT_PATH
PDB1	SNAP_953306977_973136888	/u02/app/oracle/oradata/snap_953306977_3076479.pdb
PDB1	FOTO1_PDB1	/u02/app/oracle/oradata/snap_953306977_3077712.pdb
PDB1	FOTO2_PDB1	/u02/app/oracle/oradata/snap_953306977_3078260.pdb

A partir de este punto, se realizaron 6 snapshots más, con la finalidad de validar que el noveno reemplazó al primero:

```
SQL> alter pluggable database snapshot;
```

Pluggable database altered.

```
SQL> SELECT CON_NAME, SNAPSHOT_NAME, FULL_SNAPSHOT_PATH
2 FROM DBA_PDB_SNAPSHOTS;
```

CON_NAME	SNAPSHOT_NAME	FULL_SNAPSHOT_PATH
PDB1	FOTO1_PDB1	/u02/app/oracle/oradata/snap_953306977_3077712.pdb
PDB1	FOTO2_PDB1	/u02/app/oracle/oradata/snap_953306977_3078260.pdb
PDB1	SNAP_953306977_973137446	/u02/app/oracle/oradata/snap_953306977_3078730.pdb
PDB1	SNAP_953306977_973137508	/u02/app/oracle/oradata/snap_953306977_3079126.pdb
PDB1	SNAP_953306977_973137580	/u02/app/oracle/oradata/snap_953306977_3079525.pdb
PDB1	SNAP_953306977_973137718	/u02/app/oracle/oradata/snap_953306977_3080309.pdb
PDB1	SNAP_953306977_973137798	/u02/app/oracle/oradata/snap_953306977_3080704.pdb
PDB1	SNAP_953306977_973137883	/u02/app/oracle/oradata/snap_953306977_3081107.pdb

El primer snapshot que teníamos al iniciar el ejercicio fue eliminado para crear el nuevo, esto debido a que ya se contaba con 8 copias el cual es el valor máximo como se especificó al inicio.

c. Snapshot Automático

Es posible realizar la programación de snapshots de manera automática como se muestra a continuación:

```
alter pluggable database snapshot mode every <valor> [MINUTES|HOURS];
```

Donde el valor debe ser menor a:

- 3,000 en caso de ser expresado en minutos.
- 2,000 en caso ser expresado en horas.

```
SQL> col PDB_NAME format a20
```

```
SQL> select pdb_name, snapshot_mode from dba_pdbs;
```

PDB_NAME	SNAPSH
PDB1	MANUAL

```
SQL> alter pluggable database snapshot mode every 24 hours;
```

Pluggable database altered.

```
SQL> select pdb_name, snapshot_mode from dba_pdbs;
```

PDB_NAME	SNAPSH
PDB1	AUTO

El ejemplo muestra la ejecución de snapshots de manera automática cada 24 horas y además a través de la vista DBA_PDBS se puede visualizar que la configuración cambió de MANUAL a AUTO.

d. Deshabilitar y Eliminación de Snapshots

Es posible deshabilitar la toma de snapshots con el siguiente comando:

```
alter pluggable database snapshot mode NONE;
```

```
SQL> alter pluggable database snapshot mode NONE;
```

Pluggable database altered.

Adicional si deseamos eliminar todos los snapshots procedemos a ejecutar el comando:

```
SQL> alter pluggable database set MAX_PDB_SNAPSHOTS=0;
```

A continuación se presenta los archivos que contienen los snapshots generados durante esta implementación:

```
[oracle@PRD oradata]$ ls -lrth
total 2.0G
-rw-r--r-- 1 oracle oinstall 244M Apr 11 03:53 snap_953306977_3077712.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 03:54 snap_953306977_3078260.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 03:58 snap_953306977_3078730.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 03:59 snap_953306977_3079126.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 04:00 snap_953306977_3079525.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 04:02 snap_953306977_3080309.pdb
-rw-r--r-- 1 oracle oinstall 244M Apr 11 04:04 snap_953306977_3080704.pdb
drwxr-x--- 23 oracle oinstall 4.0K Apr 11 04:04 ORCL
-rw-r--r-- 1 oracle oinstall 244M Apr 11 04:05 snap_953306977_3081107.pdb
```

Se procederá a eliminar todos los archivos de snapshots de manera automática cuando se configure a 0 la propiedad MAX_PDB_SNAPSHOTS.

```
SQL> alter session set container=PDB1;

Session altered.

SQL> ALTER PLUGGABLE DATABASE SET MAX_PDB_SNAPSHOTS=0;

Pluggable database altered.

SQL> !
[oracle@PRD ORCL]$ ls -lrth
total 4.0K
drwxr-x--- 25 oracle oinstall 4.0K Apr 11 04:16 ORCL
```

También es posible eliminar un snapshot específico ejecutando el siguiente comando:

```
alter pluggable database drop snapshot <nombre_snapshot>;
SQL> alter pluggable database drop snapshot FOTO1_PDB1;

Pluggable database altered.
```

e. Creación de un PDB a partir de un Snapshot

Recordemos que los PDB snapshots generan copias completas de los data files del PDB original permitiendo crear nuevos PDBs de manera independiente al original.

A continuación se creará un PDB a partir de un snapshot generado:

```
create pluggable database <nombre_pdb_nuevo> from <nombre_pdb_original> using
snapshot <nombre_snapshot>;
SQL> create pluggable database PDB2 from PDB1 using snapshot FOTO1_PDB1;
Pluggable database created.
SQL> show pdbs;

```

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

```
SQL> alter pluggable database PDB2 open;
Pluggable database altered.
SQL> alter session set container=PDB2;
Session altered.
SQL> select file_name from dba_data_files;
FILE_NAME
-----
/u02/app/oracle/oradata/ORCL/698C210EBD83364AE053961041640ECD/datafile/o1_mf_sys
tem_fdv59yvs_.dbf
/u02/app/oracle/oradata/ORCL/698C210EBD83364AE053961041640ECD/datafile/o1_mf_sys
aux_fdv59yvz_.dbf
/u02/app/oracle/oradata/ORCL/698C210EBD83364AE053961041640ECD/datafile/o1_mf_und
otbs1_fdv59yvz_.dbf
```

En caso se decida crear un PDB con la opción “snapshot copy” será indiferente, debido a que los snapshots no son de tipo COPY. Se presenta el siguiente ejemplo comprobando lo explicado:

```
SQL> create pluggable database PDB3 from PDB1 using snapshot FOTO2_PDB1 snapshot copy;
Pluggable database created.
```

Conclusión

Oracle Database 18c trae una excelente funcionalidad la cual puede ser claramente utilizada para crear nuevos ambientes no productivos basados en un punto en el tiempo de nuestro PDB de producción, como también contar con un punto de restauración inmediato frente a una carga masiva, un pase a producción importante u otra actividad que podría afectar a nuestro sistema productivo.

Esta funcionalidad ofrece una alta flexibilidad para configurar estas copias de manera manual o automática, la cual dependerá de nuestras necesidades, como también es posible realizar un monitoreo exacto a partir de las vistas del sistema disponible.

Publicado por:

Francisco Riccio, actualmente se desempeña como Arquitecto de Soluciones en Oracle Perú y es instructor de cursos oficiales de certificación Oracle. Es un Oracle Certified Professional en productos de Oracle Application, Base de Datos, Cloud & Virtualización.