

Misceláneo de Nuevas Funcionalidades de Oracle Database 12c (parte 3)

Por Francisco Riccio

Introducción

A continuación se detallará una lista de nuevas funcionalidades disponibles en Oracle Database 12c, que están orientadas a optimizar nuestras actividades de administración.

Implementación

1. Multithreaded Database

Oracle Database se ejecuta en un modelo multi-proceso en los sistemas operativos Unix/Linux, es decir cada background process y server process se ejecuta mediante un proceso del sistema operativo independiente. Esta arquitectura entrega escalabilidad pero en caso tengamos demasiados procesos creados también puede generar una indisponibilidad en el sistema operativo al quedarse sin recursos.

El context switch consiste en la conmutación del CPU entre un proceso y otro y es overhead puro, por lo tanto debe ser lo más rápido posible. Algunos valores típicos oscilan entre 1 y 100 mseg que se conoce como dispatch latency. En un context switch un proceso pasa de estado running a waiting y otro proceso pasa a estado running. Cuando existen demasiados procesos creados en el sistema operativo, el context switch puede generar demasiadas esperas.

Frente a este problema, Oracle Database 12c nos permite configurar nuestra base de datos a un esquema multi-hilos, donde solo existirá un proceso creado por Oracle Database y múltiples hilos, tal cual como ocurre en un Oracle Database ejecutándose en un ambiente Windows.

Esta configuración nueva no es el default y si la cambiamos no nos permitirá eliminar sesiones mediante comandos de sistema operativo, ya que cada sesión será un hilo del único proceso Oracle creado al iniciar la instancia.

Esta configuración la debemos implementar en caso tengamos eventos de espera en el sistema operativo donde evidencie que los context switch están afectando la performance del servidor y adicionalmente los recursos del sistema operativo son escasos.

Para cambiar a esta nueva configuración debemos configurar el parámetro: `threaded_execution` al valor de `true`.

Ejemplo:

```
SQL> show parameter thread
```

NAME	TYPE	VALUE
parallel_threads_per_cpu	integer	2
thread	integer	0
threaded_execution	boolean	FALSE

```
SQL> alter system set threaded_execution=true scope=spfile;
```

System altered.

Una vez aplicado el cambio podemos realizar el siguiente query:

```
SQL> SELECT PNAME, PID, SOSID FROM V$PROCESS
```

Donde el campo SOSID nos indicará si el proceso de base de datos corre mediante un proceso o un hilo.

2. PGA_AGGREGATE_LIMIT

Oracle Database 11g nos entregó un feature llamado Automatic Memory Management (AMM) el cual puede configurar libremente el tamaño del PGA y SGA basado en la cantidad de memoria configurada en el parámetro MEMORY_TARGET. Si no utilizamos este feature Oracle Database tratará de limitar el tamaño del PGA a la cantidad de memoria configurada en el parámetro PGA_AGGREGATE_TARGET.

Oracle Database 12c nos da la posibilidad de imponer una restricción fuerte del tamaño del PGA siendo controlado por el parámetro PGA_AGGREGATE_LIMIT, ejemplo:

```
SQL> show parameter pga
```

NAME	TYPE	VALUE
pga_aggregate_limit	big integer	1G
pga_aggregate_target	big integer	0

```
SQL> alter system set pga_aggregate_limit=500M;
```

System altered.

En esta configuración estamos indicando que el tamaño máximo del PGA no podrá llegar a más de 500 MB.

Nota: Este parámetro puede ser utilizado si estamos usando AMM.

3. Nuevas opciones sobre el Password File

Oracle Database 12c ha implementado nuevos parámetros disponibles en la creación del archivo Password File, los cuales se detallan:

```
ORAPWD FILE=<Ruta_Archivo> [ENTRIES=#_Usuarios] [FORCE={y|n}] [ASM={y|n}]  
[DBUNIQUENAME=<DBNAME>] [FORMAT={12|legacy}] [SYSBACKUP={y|n}] [SYSDG={y|n}]  
[DELETE={y|n}] [INPUT_FILE=<Ruta_Archivo_Previo_12c>
```

Donde los nuevos parámetros son:

a) FILE, ubicación donde se creará el Password File. En esta nueva versión está disponible la creación del archivo Password File en un ASM Diskgroup.

b) ASM=Y, significa que estamos creando un Password File para una instancia ASM.

c) DBUNIQUENAME=<DBNAME>, esto le permite al Oracle Database identificar el Password File que reside en el ASM Diskgroup. Esta opción solo debe ser utilizada cuando nuestro Password File reside en un ASM Diskgroup.

d) FORMAT=12, es el default. Si este parámetro tiene el valor de legacy entonces no podremos almacenar los nuevos privilegios de Oracle Database 12c (SYSBACKUP, SYSDG) en el Password File.

Solo estarán disponibles los privilegios SYSDBA y SYSOPER si le colocamos el valor de legacy.

e) INPUT_FILE, es usado cuando queremos migrar un Password File que proviene de una versión anterior al nuevo formato disponible en Oracle Database 12c. En ese parámetro indicamos la ubicación original del Password File a migrar al nuevo formato.

Ejemplo:

Migraremos un Password File de una versión Oracle Database 11g al nuevo formato disponible en Oracle Database 12c y lo alojaremos en un ASM Diskgroup llamado +DG_PRD.

```
orapwd FILE='+DATA/DG_PRD/orapwPRD' DBUNIQUENAME='PRD' FORMAT=12 INPUT_FILE='  
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/orapwPRD' FORCE=y
```

El contenido del archivo puede ser consultado mediante la vista V\$PWFILERS.

```
SQL> select * from v$pwfile_users;
```

USERNAME	SYSDB	SYSOP	SYSAS	SYSBA	SYSDG	SYSKM	CON_ID
SYS	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	0

4. Mejoras sobre el utilitario DATAPUMP.

Oracle Database 12c ha traído nuevas mejoras sobre el utilitario DATAPUMP, dos de ellas se presentan a continuación.

a) Exportación e Importación de Vistas como tablas.

Es posible exportar o importar una vista como una tabla, por lo cual se creará un segmento de tipo TABLE que tendrá físicamente la información de la vista.

El parámetro que permite esta labor es: VIEW_AS_TABLES=<NOMBRE_VISTA>, ejemplo:

```
C:\>expdp userid=friccio/oracle@PDB_PRD directory=DIR_ORACLE dumpfile=datos.dmp views_as_tables=V_DEMO
Export: Release 12.1.0.0.2 - Beta on Mar May 21 14:54:53 2013
Copyright (c) 1982, 2012, Oracle and/or its affiliates. All rights reserved.
Conectado a: Oracle Database 12c Enterprise Edition Release 12.1.0.0.2 - 64bit Beta
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Iniciando "FRICCIO"."SYS_EXPORT_TABLE_01":  userid=friccio/*****@PDB_PRD directory=DIR_ORACLE dumpf
s_as_tables=V_DEMO
Estimaci3n en curso mediante el m3todo BLOCKS...
Procesando el tipo de objeto TABLE_EXPORT/VIEWS_AS_TABLES/TABLE_DATA
Estimaci3n total mediante el m3todo BLOCKS: 16 KB
Procesando el tipo de objeto TABLE_EXPORT/VIEWS_AS_TABLES/TABLE
. . "FRICCIO"."V_DEMO" 5.046 KB 1 filas exportadas
La tabla maestra "FRICCIO"."SYS_EXPORT_TABLE_01" se ha cargado/descargado correctamente
*****
El juego de archivos de volcado para FRICCIO.SYS_EXPORT_TABLE_01 es:
C:\DATOS.DMP
El trabajo "FRICCIO"."SYS_EXPORT_TABLE_01" ha terminado correctamente en Mar May 21 14:55:04 2013 elap
```

Aquí podemos apreciar que la vista V_DEMO fue exportada como una tabla.

b) Deshabilitaci3n del modo DIRECT PATH.

Por default Oracle Database 12c realiza las operaciones de INSERT con DIRECT PATH logrando generar mínima informaci3n de redo y mejorando el tiempo de la importaci3n.

Si queremos deshabilitar esta opci3n en el utilitario IMPDP debemos colocar el siguiente parámetro: DATA_OPTIONS=DISABLE_APPEND_HINT.

Colocando est3 opci3n importaríamos de la manera convencional.

5. Novedades en el Oracle Data Guard.

En configuraciones de Data Guard donde tenemos una base de datos primaria que envía la información de sus redo a múltiples standby database, se podría conseguir una carga y uso de recursos considerable, haciendo que la sincronización pueda verse afectada.

Frente a este problema, Oracle Database 12c ha introducido un nuevo mecanismo en el cual la base de datos primaria envía a un standby database central conocido como "Far Sync Standby" la información de los redo generados y este hará el envío a los demás standby database que configuremos como destinatarios. Básicamente actúa como un coordinador.

El Far Sync Standby no cuenta con datafiles y no realiza la operación de recover, solo recibe la información de la base de datos primaria y la retransmite a los demás standby database, liberando la carga al productivo. Asimismo tiene la capacidad de hacer failover si es requerido por el usuario.

Este nuevo tipo de standby database soporta máxima performance y disponibilidad.

El control file que mantiene este nuevo tipo de standby database tendrá el siguiente rol: FARSYNC SERVER, esto lo podemos validar en la vista V\$DATABASE en el campo: DATABASE_ROLE.

La creación de este tipo especial de control file lo generamos a partir del siguiente comando en la base de datos primaria:

```
SQL> ALTER DATABASE CREATE FAR SYNC STANDBY CONTROLFILE AS '<ruta_control_file>';
```

En el siguiente URL se adjunta una implementación completa de este nuevo feature paso a paso:

<http://friccio-oracle.blogspot.com/2013/05/far-sync-standby-database.html>

Conclusiones

Durante el transcurso de este material se ha mostrado algunas de las nuevas opciones que Oracle Database 12c nos entrega en pro de tener más alternativas de solución a los problemas que se presentan en el día a día de una administración de base de datos.

Publicado por Ing. Francisco Riccio. Es un IT Oracle Specialist e instructor de cursos oficiales de certificación Oracle. Está reconocido por Oracle como un Oracle ACE y certificado en productos de Oracle Application / Oracle Database / Oracle Developer.

e-mail: francisco@friccio.com

web: www.friccio.com