



Prácticas de Administración de Bases de Datos

Grado en Ingeniería Informática

PRÁCTICA 1

Arquitectura del SGBD Oracle

OBJETIVOS

- Saber consultar el diccionario de datos para conocer la arquitectura y el rendimiento de una BD Oracle

Ejercicios

Obteniendo información de la BD

Debéis guardar la información que se solicita en un fichero de texto para que, posteriormente, podáis verificar las respuestas.

Hay diferentes vistas que permiten visualizar los parámetros de inicialización:

- V\$SYSTEM_PARAMETER: valor actual de los valores de la instancia
- V\$SYSTEM_PARAMETER2: igual que la anterior pero con una visualización en varias líneas de los parámetros que tienen una lista de valores
- V\$PARAMETER y V\$PARAMETER2: tienen la misma estructura que las anteriores pero muestran los valores actuales de los parámetros de la sesión actual. Muestran el valor de los parámetros que han sido eventualmente modificados en la sesión mediante la sentencia ALTER SESSION
- V\$SPPARAMETER: contenido actual del fichero de parámetros de servidor activo

1. Haciendo consultas a las vistas del fichero de inicialización, responded a las siguientes preguntas:

- a) ¿Se gestiona de forma automática la memoria de la instancia?. ¿Por qué?

Oracle gestiona de forma automática la memoria de la instancia (SGA + PGA) a partir de la versión 11G. Esto es debido a que Oracle reparte esa cantidad entre la PGA y los componentes de la SGA gestionados automáticamente.

- b) ¿Cuánta memoria asigna Oracle a la SGA + PGA?

Para poder ver cuanta memoria esta asignada a la SGA + PGA, se debe utilizar el siguiente comando:

SELECT * FROM V\$SYSTEM_PARAMETER where name = 'memory_target'

- c) ¿Hay un valor máximo para el tamaño de la memoria de la instancia? ¿Ha sido asignado explícitamente?

Sí, para poder visualizarlo se utilizará el siguiente comando:

SELECT * FROM V\$SYSTEM_PARAMETER where name = 'memory_max_target'

No se asigna explícitamente, aunque se podría. Oracle lo hace por defecto de forma automática a partir de la versión 11g.

d) ¿Se asegura un valor mínimo para la SGA?

Con la gestión automática no se asegura un valor mínimo, se debería indicar explícitamente.

Para comprobar el valor actual de la SGA se utiliza el siguiente comando:

SELECT * FROM V\$SYSTEM_PARAMETER where name = 'sga_target'

e) ¿Cuántos parámetros del fichero de inicialización son explícitos?

Los parámetros explícitos son aquellos que se indican en el fichero de parámetros, es decir, en la vista V\$SPPARAMETER.

Para poder saber el nº de parámetros explícitos, tendremos que comprobar aquellos que no tengan el valor a null, y contarlos con COUNT:

select count(*) from v\$spparameter where value is not null

f) ¿Cuál es el tamaño del bloque en la BD Onuba?

Para comprobarlo se utiliza la siguiente sentencia:

SELECT * FROM V\$SYSTEM_PARAMETER where name = 'db_block_size'

2. Asignar un valor de 256M a la memoria compartida pero mantener dicho valor únicamente en memoria (no guardar en el fichero spfile)

alter system set shared_pool_size = 256M scope = MEMORY

3. Comprobar que el cambio se ha realizado únicamente en la sesión actual. Volver a asignar el mismo valor pero almacenándolo en el fichero de inicialización.

Para comprobar que el cambio es únicamente efectivo en la sesión actual, primero se comprobará la vista V\$SYSTEM_PARAMETER para ver si ha tomado efecto el cambio:

select * from v\$system_parameter where NAME = 'shared_pool_size'

Como el cambio es únicamente en memoria, si comprobamos los parámetros del fichero de inicialización (vista V\$SPPARAMETER), debería tener aún los valores originales:

select * from v\$spparameter where NAME = 'shared_pool_size'

Para asignar el valor al fichero de inicialización:

```
alter system set shared_pool_size = 256M scope = SPFILE
```

4. Reiniciar la base de datos y comprobar que arranca con el nuevo valor para la memoria compartida

Para cerrar la base de datos, vamos a sqlplus:

```
CONNECT / AS SYSDBA;
```

```
SHUTDOWN IMMEDIATE;
```

Y para volver a iniciarla:

```
CONNECT / AS SYSDBA;
```

```
STARTUP MOUNT;
```

```
ALTER DATABASE OPEN;
```

```
select display_value from v$system_parameter where NAME =  
'shared_pool_size'
```

5. ¿Cómo obtenemos los parámetros que son modificables a nivel de sesión con alter sesión?

```
select * from v$system_parameter where ISSES_MODIFIABLE = 'TRUE'
```

6. Editar el archivo de inicialización de la base de datos (estará ubicado en c:\app\usuario\admin\\${ORACLE_SID}\pfile y se llamará **init.ora** seguido de una serie de números). Observar algunos de los parámetros de inicialización e intentar entender su significado

7. Añadir una entrada en el fichero init asignando un tamaño máximo (algo mayor que el actual) a la memoria de la instancia. Comprobar que el cambio se ha llevado a cabo correctamente (Nota, es posible que alguna operación deba hacerla bajo Sql Plus con el usuario SYS (o SYSDBA))

Una vez dentro del archivo, en MISCELLANEOUS, se añade el parámetro memory_max_target:

```
#####  
# Miscellaneous  
#####  
compatible=11.2.0.0  
diagnostic_dest=C:\app\AdminBBDD  
memory_target=1717567488  
memory_max_target = 2G
```

Una vez puesto el parámetro, será necesario crear un nuevo fichero spfile para que tome efecto este aumento de la memoria máxima. Para ello deberemos iniciar sesión como SYSDBA en SQLPLUS:

```
create spfile =
'C:\app\AdminBBDD\product\11.2.0\dbhome_2\database\SPFILEONUBA.ORA.
NEW' from pfile =
'C:\app\AdminBBDD\admin\Onuba\pfile\init.ora.1132023124134'
```

Una vez creado el fichero, se cambiará de nombre el SPFILE original a SPFILEONUBA.ORA.ORIGINAL y el que hemos creado a SPFILEONUBA.ORA.

Posteriormente se procederá a comprobar que se ha hecho efectivo el cambio con:

```
select * from v$system_parameter where NAME = 'memory_max_target'
```

Hay varias vistas dinámicas de la BD que nos dan información sobre el tamaño y la estructura de la SGA:

- V\$SGAINFO
- V\$SGA_DYNAMIC_COMPONENTS
- V\$SGA_TARGET_ADVICE
- V\$SGA
- V\$SGASTAT
- V\$MEMORY_DYNAMIC_COMPONENTS
- V\$SGA_DYNAMIC_FREE_MEMORY

8. ¿Qué tamaño tiene la SGA fija? ¿Qué tamaño tiene el buffer del registro de deshacer?

Para ver el tamaño de la SGA fija:

```
select BYTES from v$sgainfo where NAME = 'Fixed SGA Size'
```

Para ver el buffer del registro de deshacer:

```
select BYTES from v$sgainfo where NAME = 'Redo Buffers'
```

9. ¿Qué tamaño se le ha asignado a la SGA y a la PGA de forma automática? ¿Por qué el tamaño de la shared_pool es distinto al que se especificó en el ejercicio 3 (PISTA: debéis mostrar las columnas current_size y user_specified_size)?

Para ver el tamaño de la SGA:

```
select component, current_size/(1024 * 1024) as CurrentSizeMB,
user_specified_size/(1024 * 1024) as UserSizeMB from
```

v\$memory_dynamic_components where COMPONENT = 'SGA Target'

Para ver el tamaño de la PGA:

**select component, current_size/(1024 * 1024) as CurrentSizeMB,
user_specified_size/(1024 * 1024) as UserSizeMB from
v\$memory_dynamic_components where COMPONENT = 'PGA Target'**

El tamaño de la shared pool se puede observar mediante:

**select component, current_size/(1024 * 1024) as CurrentSizeMB,
user_specified_size/(1024 * 1024) as UserSizeMB from
v\$memory_dynamic_components where COMPONENT = 'shared pool'**

Es distinto al establecido en el ejercicio 3 ya que reconstruimos el archivo SPFILEONUBA.ORA mediante un archivo de inicialización que no contemplaba el parámetro shared pool.

10. Consultar información sobre la base de datos y la instancia. Indique las instrucciones necesarias para obtener la siguiente información ¿En qué fecha fue creada la base de datos ONUBA?. ¿Cómo se llama la máquina en la que corre la base de datos?

Para ver la fecha de creación de ONUBA:

select CREATED from v\$database

Para ver la máquina en la que corre la instancia:

select HOST_NAME from v\$instance

11. Localizar el proceso “servidor” asociado a la sesión que estáis utilizando (pista: estas accediendo con el programa Sql Developer). ¿Es un servidor dedicado o compartido? ¿Cómo se llama el usuario del sistema operativo y el ordenador desde el que se está realizando la conexión?

Para ver si el servidor es dedicado o compartido:

select server from v\$session where SCHEMANAME = 'SYSTEM'

Para ver el usuario del SO y el ordenador:

select OSUSER, MACHINE from v\$session where SCHEMANAME = 'SYSTEM'

12. Ver las sentencias SQL que guarda la *SharedPool* (vista **v\$sqlarea**)

En la vista V\$SQLAREA podemos ver el contenido del “área SQL” de la SharedPool. Además podemos obtener información útil para el ajuste de cada una de las sentencias SQL (Shared Pool consumida, nº de veces que se ha ejecutado, nº de veces que se ha salido de la caché, lecturas físicas, tiempo de CPU, tiempo total incluyendo compilación, etc.).

Si consultamos desde el usuario SYS o SYSTEM esta vista obtendremos un número elevado de consultas internas que se hacen en el sistema. Para poder hacer bien el ejercicio, haremos lo siguiente:

- Desbloquear la cuenta del usuario scott (clave: tiger)
Ejecute las siguientes instrucciones
alter user scott account unlock;
alter profile default limit password_life_time unlimited;
alter profile DEFAULT limit FAILED_LOGIN_ATTEMPTS UNLIMITED;
alter user scott identified by tiger;
- Desde el usuario scott, crear una tabla cualquiera e insertar algunas filas
- También desde el usuario scott, realizar 2 o 3 consultas SELECT
- Desde el usuario SYS/SYSTEM, consultar el área SQL de las consultas realizadas por el usuario scott
- El campo que contiene el nombre del usuario que ha realizado la consulta es **parsing_schema_name**
- Visualizar las consultas del usuario SCOTT que se encuentran en la SharedPool.

Con el usuario SCOTT creamos la siguiente tabla:

```
CREATE TABLE DRAGON_BALL_CHARACTER(  
  NAME VARCHAR(255),  
  HEIGHT_IN_CM int,  
  RACE VARCHAR(255)  
);
```

Insertamos las siguientes filas:

```
insert into dragon_ball_character values ('Broly', 230, 'Saiyan');  
insert into dragon_ball_character values ('Gotenks', 150, 'Saiyan');  
insert into dragon_ball_character values ('Krillin', 160, 'Human');
```

Hacemos las siguientes consultas SELECT:

```
select * from dragon_ball_character  
select * from dragon_ball_character where RACE = 'Saiyan'
```

Para consultar el área SQL de SCOTT:

```
select * from v$sqlarea where parsing_schema_name = 'SCOTT'
```

Para visualizar las consultas SQL almacenadas en la SharedPool que ha hecho SCOTT:

```
select SQL_TEXT from v$sqlarea where parsing_schema_name = 'SCOTT'
```

13. Comprobar el funcionamiento de la caché de redolog, como protectora del contenido de la caché de datos. Para ello, se creará una transacción y se provocará una caída de la BD. Comprobar que, al arrancarla de nuevo, se mantiene la integridad de la misma.

- Crear una tabla cualquiera en el espacio de tablas del usuario SCOTT
- Insertar una fila sin hacer commit y forzar la caída de la BD
- Arrancar de nuevo la BD y comprobar que la fila insertada no está
- Repetir la inserción de la fila y hacer commit. Forzar de nuevo la caída de la BD
- Arrancar la BD y comprobar que ahora la fila si está insertada

14. Con este ejercicio vamos a comprobar el funcionamiento de la caché de datos.

- Activar la medición de tiempos mediante SET TIMING ON
- Lanzar la consulta `SELECT COUNT(*) FROM DBA_SOURCE`
- Volver a lanzar la misma consulta
- Comprobar que la segunda ejecución tarda mucho menos, ya que los datos ya se cargaron en la caché de datos al lanzarla la primera vez y, por tanto, se accede directamente a memoria y no a disco