



UNIVERSIDAD SIMÓN BOLÍVAR

Departamento de Computación y Tecnología de la Información

Arquitectura y Administración de Bases de Datos

Septiembre-Diciembre 2014

TAREA 1

Integrantes

Luis Fernandes. 10-10239.

José Montenegro. 10-10469

Septiembre 2014

TABLA DE CONTENIDOS

1	Parte C: Tablespaces y Datafiles.....	3
2	Parte D: Análisis de Script.....	4
3	Parte E: Estados de Inicialización y Opciones de Apagado de la Base de Datos	6
4	Parte F: Consultas.....	8

1 PARTE C: TABLESPACES Y DATAFILES

1. Utilizando sólo comandos SQL altere los datafiles de cada uno de los tablespaces para que no les sea permitido crecer de forma ilimitada, sino que más bien sean de tamaño fijo con los siguientes tamaños:

Nombre DD	Tamaño
example01.dbf	128M
sysaux01.dbf	512M
system01.dbf	768M
temp01.dbf	1GB
undotbs01.dbf	1.5GB
users01.dbf	10M

```
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/users01.dbf'  
RESIZE 10M;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/users01.dbf'  
AUTOEXTEND OFF;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/example01.dbf'  
RESIZE 128M;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/example01.dbf'  
AUTOEXTEND OFF;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/sysaux01.dbf'  
RESIZE 512M;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/sysaux01.dbf'  
AUTOEXTEND OFF;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/system01.dbf'  
RESIZE 768M;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/system01.dbf'  
AUTOEXTEND OFF;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/undotbs01.dbf'  
RESIZE 1536M;  
ALTER DATABASE DATAFILE '/tablespace/fuentemagna/fuentemagna/undotbs01.dbf'
```

```
AUTOEXTEND OFF;
```

```
ALTER DATABASE TEMPFILE '/tablespace/fuentemagna/fuentemagna/temp01.dbf'
```

```
RESIZE 1G;
```

```
ALTER DATABASE TEMPFILE '/tablespace/fuentemagna/fuentemagna/temp01.dbf'
```

```
AUTOEXTEND OFF;
```

```
SELECT FILE_NAME, BYTES, AUTOEXTENSIBLE FROM DBA_DATA_FILES;
```

```
SELECT FILE_NAME, BYTES, AUTOEXTENSIBLE FROM DBA_TEMP_FILES;
```

2. Cree un tablespace denominado tbsautouniforme que sea *Locally managed*, cuyo datafile sea de tamaño fijo: 100MB, la administración del espacio libre en los segmentos sea automática, es decir con un mapa de bits y los extents sean de tamaño uniforme de un 1MB.

```
CREATE TABLESPACE tbsautouniforme DATAFILE
```

```
'/tablespace/fuentemagna/fuentemagna/tbsautouniforme.dbf'
```

```
SIZE 100M
```

```
EXTENT MANAGEMENT LOCAL UNIFORM // UNIFORM por defecto extiende a 1M
```

```
SEGMENT SPACE MANAGEMENT AUTO;
```

2 PARTE D: ANÁLISIS DE SCRIPT

A continuación se le muestra un *script* de SQL*Plus que está siendo desarrollado por un DBA para automatizar algunas funciones de administración. Los números a la izquierda corresponden al número de línea del *script*:

```
1: define FILE=datafiles.sql
2: SET ECHO OFF;
3: SET PAGESIZE 50;
4: SET LINESIZE 150;
5: SET HEADING OFF;
6: spool &FILE
7: select 'ALTER TABLESPACE ' || RTRIM(tablespace_name) ||
8: 'ADD DATAFILE' || chr(39) ||
9: SUBSTR(file_name,1,INSTR(file_name,'.',1,1)-1) ||
10: 'automatic.dbf' || chr(39) || 'SIZE' ||
11: RTRIM(BYTES/1024) || 'K AUTOEXTEND' ||
12: RTRIM(DECODE(autoextensible, 'YES', 'ON NEXT' ||
13: RTRIM(INCREMENT_BY) || ' MAXSIZE' ||
14: RTRIM(MAXBYTES/1024) || 'K', 'OFF')) || ' ';
15: from dba_temp_files t
16: where tablespace_name in (select tablespace_name
17: from dba_temp_files
18: group by tablespace_name
19: having (sum(maxbytes-bytes)/1024)/1024<40000)
20: and file_id=(select min(file_id)
21: from dba_temp_files t1
22: where t.tablespace_name=t1.tablespace_name);
23: spool off;
```

24: SET HEADING ON

Responda las siguientes preguntas relacionadas con el *script* anterior:

1. Descargue el script del aula virtual del curso y ejecútelo. Examine el archivo SPOOL generado. En base a esta información diga qué hace este script.

El script agrega un nuevo datafile al tablespace temp llamado temp01automatic con tamaño fijo de 1048576K bytes. El datafile parece ser una especie de extensión a un tempfile, pues la consulta generadora solo busca de los tempfiles.

2. ¿Qué función realiza la consulta expresada entre las líneas 7 y 22?

Primero se tiene el SELECT, el cual retorna un texto con la instrucción necesaria. Desglosando cada punto de concatenación se tiene:

+ALTER TABLESPACE: Alterar un tablespace.

+RTRIM(tablespace_name): El nombre de la tablespace objetivo, eliminando espacios en blanco a la derecha.

+ADD DATAFILE: Agregar un datafile.

+chr(39): Caracter ' , para comenzar un string dentro del comando.

+SUBSTR(file_name,1,INSTR(file_name,'.',1,1)-1): nombre del datafile devuelto por la query desde el inicio hasta justo antes de la primera ocurrencia de '.'

+'automatic.dbf': agregado al nombre sacado en el paso anterior.

+chr(39): Símbolo ' de finalización de string dentro del comando.

+'SIZE ': especifica el tamaño del nuevo datafile.

+RTRIM(BYTES/1024) || 'K AUTOEXTEND ': El tamaño es el mismo que el del tempfile original, expresado en Kilobytes. También se empieza a definir si el archivo es de tamaño fijo o no.

+(RTRIM(DECODE(autoextensible, 'YES', 'ON NEXT ' || RTRIM(INCREMENT_BY)||' MAXSIZE '|| RTRIM(MAXBYTES/1024)||'K','OFF'))||' ' ;': Se usa decode para cambiar la opción de AUTOEXTEND dependiendo de si el archivo original era autoextensible o no. Si no lo era, la opción queda igual. Si sí lo era, también lo será el archivo nuevo. Además, su incremento automático y tamaño máximo será igual al del original.

Luego se tiene el FROM. La búsqueda se hace dentro de los tempfiles.

Finalmente se tiene el WHERE. Se quiere restringir la búsqueda a aquellos tablespaces cuyos archivos posean espacio libre menor a 40MB en total, y luego elegir el archivo más viejo entre los que se encuentran en dicho tablespace.

3. ¿Qué efecto tienen las líneas 7-14 desde el punto de vista de lo que debe generar la consulta?

Estas líneas determinan el comando SQL a ejecutar, teniendo los datos en modo placeholder mientras se termina la consulta. Básicamente genera el texto del comando en base a la consulta realizada.

3 PARTE E: ESTADOS DE INICIALIZACIÓN Y OPCIONES DE APAGADO DE LA BASE DE DATOS

1. Los Estados de inicialización de la base de datos son: *no mount*, *mount*, *open*. Describa las operaciones realizadas en cada uno de ellos.

NOMOUNT: inicia la instancia sin montar una base de datos, esto no permite el acceso a la base de datos y usualmente podría ser hecho para la creación o re-creación de archivos de control.

MOUNT: inicia la instancia y monta la base de datos, pero la mantiene cerrada. Este estado permite algunas actividades de la DBA, pero no permite el acceso general a la base de datos.

OPEN: inicia la instancia y monta y abre la base de datos. Esto puede ser hecho en modo no-restringido, permitiendo el acceso a todos los usuarios, o en el caso del modo restringido, permitiendo el acceso para los administradores de la base de datos únicamente.

2. Si la base de datos está apagada, ¿con qué instrucción se puede iniciar en estado *mount*?

Desde SQL*Plus:

STARTUP MOUNT

O para SRVCTL (cuando Oracle Restart está en uso):

```
srvctl start database -db db_unique_name -startoption mount
```

3. Si la base de datos está en estado *nomount*, ¿cuáles serían las instrucciones para llevarla paulatinamente a estado *mount* y luego *open*?

ALTER DATABASE MOUNT;

ALTER DATABASE OPEN;

4. Las opciones de apagado de la base de datos son: *normal*, *transactional*, *immediate*, *abort*.

NORMAL: antes de que la base de datos sea apagada, espera a todos los usuarios conectados en el momento se desconecten de la base de datos. Nuevas conexiones no son permitidas luego de que la orden ha sido efectuada.

TRANSACTIONAL: este modo es usado para hacer un shutdown planificado de una instancia, permitiendo que transacciones activas sean completadas primero. Al hacer un transactional shutdown:

- Nuevas conexiones no son permitidas, tampoco nuevas transacciones.
- Luego de que todas las transacciones han sido completadas, cualquier cliente aún conectado a la instancia es desconectado.
- Luego, el proceso de apagado es como si un SHUTDOWN IMMEDIATE fuera ejecutado.

El próximo startup de la base de datos no requerirá ningún procedimiento de recuperación.

IMMEDIATE: usar un immediate shutdown solo en las siguientes situaciones:

- Para iniciar un backup automático e inatendido.
- Cuando un corte de energía va a ocurrir pronto.
- Cuando la base de datos o alguna de sus aplicaciones está funcionando irregularmente y no se puede contactar a los usuarios para que se desconecten o están incapacitados para desconectarse.

Al apagar la base de datos en immediate:

- Nuevas conexiones no serán permitidas, tampoco nuevas transacciones.
- Las transacciones no confirmadas se deshacen (si alguna transacción extensa existe, este método de apagado podría no completarse rápidamente a pesar de su nombre).
- La base de datos de Oracle no espera por los usuarios para ser desconectada al momento. La base de datos deshace todas las transacciones activas y desconecta a todos los usuarios conectados.

El próximo startup de la base de datos no requerirá procedimientos de recuperación de la instancia.

ABORT: con este modo se puede apagar la base de datos instantáneamente, abortando la base de datos. Este modo debe ser ejecutado únicamente en las siguientes situaciones:

- La base de datos o alguna de sus aplicaciones está funcionando irregularmente y ninguno de los otros modos de apagado funciona.
- Se ve en la obligación de apagar la base de datos instantáneamente (por ejemplo, si sabe que algún corte eléctrico ocurrirá en un minuto).
- Presenta problemas al iniciar una instancia de base de datos.

Al realizar un shutdown en este modo:

- Nuevas conexiones no son permitidas, tampoco nuevas transacciones.
- Cualquier orden SQL procesada por la base de datos de Oracle es interrumpida en el acto.

- A las transacciones no confirmadas no se les hace roll back.
- La base de datos de Oracle no espera a que los usuarios conectados se desconecten. La base de datos desconecta a todos los usuarios implícitamente.

El próximo startup de la base de datos requerirá procedimientos de recuperación de la instancia automáticamente.

5. ¿Cuál es la opción de apagado cuya utilización es poco recomendada? ¿Por qué?

El ABORT es el mecanismo de apagado menos recomendado, ya que no procede a realizar ningún mecanismo de seguridad previo al apagado. Lo que conlleva a realizar procedimientos de recuperación en la próxima inicialización de la instancia, lo cual los otros métodos no necesitan hacer.

4 PARTE F: CONSULTAS

Desarrolle consultas que permitan realizar las siguientes funciones:

1. Determinar el tamaño actual del SGA, buffer cache, share pool, large pool, redo log buffer y componentes estáticas en Kilobytes. Para verificar su consulta puede utilizar el comando *show sga*. Vale destacar que el SGA se compone de: los redo log buffers, los componentes dinámicos del SGA (variable SGA) que incluyen: Shared Pool, Large Pool, Buffer Cache. Los componentes estáticos (fixed SGA) que contiene información acerca del estado de la BD y la instancia.

```
SELECT NAME AS NOMBRE, BYTES/1024 AS TAMANHO
FROM V$SGAINFO
WHERE NAME='Maximum SGA Size' OR NAME='Buffer Cache Size'
      OR NAME='Shared Pool Size' OR NAME='Large Pool Size'
      OR NAME='Redo Buffers' OR NAME='Fixed SGA Size';
```

2. Generar un script para eliminar todas las conexiones ociosas de la BD con más de 24 horas de conexión.

```
-- Desconecta todas las sesiones que tengan más de un día inactivas.
SET HEADING OFF;
SELECT 'ALTER SYSTEM DISCONNECT SESSION '||SID||','||SERIAL#||' IMMEDIATE;'
FROM V$SESSION
WHERE STATUS='INACTIVE' AND LAST_CALL_ET > 86400;
SET HEADING ON;
```


3. Dado el nombre de una tabla y su dueño usuario indicar las columnas que forman la tabla y si admiten o no el valor NULL.

```
SELECT DISTINCT COL.COLUMN_NAME, DECODE(CON.CONSTRAINT_TYPE, 'P',
'NO','YES')
FROM ALL_IND_COLUMNS COL, ALL_CONSTRAINTS CON
WHERE COL.TABLE_NAME=CON.TABLE_NAME AND COL.TABLE_NAME=table_name
AND COL.OWNER=owner;
```

4. Dado el nombre de una tabla y su dueño usuario, indicar las restricciones de integridad de la misma, incluyendo para cada restricción su tipo.

```
SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE
FROM ALL_CONSTRAINTS
WHERE OWNER=owner AND TABLE_NAME=table_name;
```

5. Determinar los nombres de los usuarios de la base de datos cuya cuenta se encuentra bloqueada.

```
SELECT USERNAME
FROM DBA_USERS
WHERE LOCK_DATE IS NOT NULL;
```

6. Determinar el número de sesiones abiertas por usted.

```
SELECT COUNT(USERNAME) AS NRO_DE_SESIONES
FROM V$SESSION
WHERE USERNAME=
(SELECT DISTINCT USERNAME
FROM V$SESSION S, V$MYSTAT M
WHERE S.SID=M.SID);
```