

PRACTICAS TEMA 2.

DISEÑO Y CREACION DE UNA BASE DE DATOS.

2.1. Diseñar, sobre el papel, una BD teniendo en cuenta las contiguas de E/S y la organización de los tablespaces básicos. Suponer 4 discos: /u01 (SW) y /u02, /u03, /u04 (BD).

2.2. Crear un init.ora, diseñando requerimientos de memoria, suponiendo 4Gb de RAM y 21 instancias con 10 usuarios cada una. Se puede usar un máximo del 64% de la RAM para las SGAs de todas las bases de datos. Los checkpoints sucederán cuando se llene el fichero redolog y, como máximo, cada 30 minutos. La gestión de rollback será automática. Tendremos dos copias del fichero de control en /u02 y /u03.

2.3. Crear la BD que hemos diseñado anteriormente, y para la que también hemos creado un init.ora. Hacerlo manualmente con CREATE DATABASE. El nombre de la BD será ALUxy. Antes de lanzar el comando de creación, ponte de acuerdo con un compañero para que no lo hagais a la vez.

2.4. Crear la misma BD anterior, esta vez usando OMF, y cambiando el nombre por OMFxy, creando los ficheros bajo /u02/oradata/OMFxy y usando dos destinos para redolog (en /u03 y /u04).

2.5. Consultar el “Simulador de DBCA”.

Recursos.

- Máquina: cursos.atika.um.es
- Usuario unix: curso51 a curso70
- Conexión máquina unix: ssh con Secure Shell
- Bases de datos: CURSO51 a CURSO70
- Conexión a bd: sqlplus desde sesión unix, usuario “/ as sysdba”

2.1. Diseñar, sobre el papel, una BD teniendo en cuenta las contiendas de E/S y la organización de los tablespaces básicos. Suponer 4 discos: /u01 (SW) y /u02, /u03, /u04 (BD).

- Comprobar discos reales del servidor y ocupación (df). También actividad (iostat). El SW de Oracle ya está instalado en el disco hdc (concretamente /u01) y la BD en el disco hda (en /u02, /u03 y /u04, siendo los dos primeros enlaces a /antiguaFC3/home/u02 y /antiguaFC3/home/u03, respectivamente, y /u04, q sólo tiene los ficheros redolog, realmente está en el el disco hda en /antiguaFC3/raiz/u04). En este apartado, vamos a suponer que no hay nada instalado, y en función de los datos que nos da "df" e "iostat", vamos a hacer un supuesto reparto del SW y BD, de forma que supondremos que el SW necesitaría 3Gb y la BD 4Gb.
- Después vamos a diseñar una base de datos ALUxy, para la que disponemos de tres directorios (imaginemos que son discos): /u02, /u03 y /u04). Además de los tablespaces básicos (system, sysaux, temp y undo_rbs) vamos a añadir los tablespaces: users, datos, índices, estatico_dat y estatico_ind.
- Tablespaces system (/u03), temp (/u03), undo_rbs (/u03), sysaux (/u02), users (/u02), datos (/u02), índices (/u03), estáticos (/u02), etc. Y redo logs en un disco aparte (/u04).

Solución:

- Comprobar espacio en disco y actividad de los mismos, y decidir como repartir SW (3Gb) y BD(4Gb).

```
$ df -h
S.ficheros          Tamaño Usado  Disp Uso% Montado en
/dev/mapper/VolGroup00-LogVol100
                    20G   17G   1,8G  91% /
/dev/hda3           99M   31M   64M  33% /boot
none                2,0G    0   2,0G  0% /dev/shm
/dev/hda1           99M   22M   72M  24% /antiguaFC3/boot
/dev/hda2           50G   42G   5,7G  88% /antiguaFC3/raiz
/dev/hdc2           99G   83G   11G  89% /antiguaFC3/home
/dev/hdc3           30G   18G   11G  61% /u01

/dev/hda2           50G   22G   26G  46% /
/dev/hda1           99M   11M   83M  12% /boot
none                2,0G    0   2,0G  0% /dev/shm
/dev/hdc2           99G   48G   46G  51% /home
```

(Como se observa, en /antiguaFC3/home hay 11Gb libres, casi el doble de espacio q en el otro disco (/antiguaFC3/raiz); por tanto, si nos ceñimos al espacio libre, yo pondría la BD en /antiguaFC3/home. Para tu información, actualmente aquí es donde están instaladas las bbdd de prácticas.).

```
$ iostat 5 5
...
Device:            tps    Blq_leid/s    Blq_escr/s    Blq_leid    Blq_escr
hda                 2,40         41,58         48,50    341770897    398710926
hdc                 17,93         32,68         466,08    268667482    3831279320
...
hda                 1,00          0,00          16,03         0           80
hdc                 12,83          0,00         426,45         0          2128
```

(Con "iostat 5 5" hemos pedido al SO las estadísticas de E/S en 5 intervalos de 5 segundos, teniendo en cuenta q en el primer intervalo nos mostrará las estadísticas acumuladas desde el último arranque hasta ese momento, y en los otros 4 intervalos sólo

mostrará los datos de cada uno de ellos respectivamente. Si tenemos en cuenta sólo la carga de E/S, montaríamos la BD en hda, es decir, bajo el /antiguaFC3/raiz y no bajo /antiguaFC3/home. Si además vemos los datos del espacio libre, como en /antiguaFC3/raiz tenemos 5,7Gb libres y nos basta con 4Gb para la supuesta BD, por lo optaríamos por /antiguaFC3/raiz para la BD.)

Vamos a separar el software de la BD, dedicando la partición con más E/S para el SW (/antiguaFC3/home) y la que tiene menos, para la BD (/antiguaFC3/raiz). De esta forma, **definiríamos los directorios /antiguaFC3/home/u01 (creando el enlace /u01) para el SW y /antiguaFC3/raiz/u02 y u03 para la BD (creando el enlace /u02 y /u03)**. Además, en cuanto a la BD se refiere, separaremos el redo de los datos, creando /antiguaFC3/home/u04 (y el enlace /u04) para los ficheros redo log.

- Diseñar, sobre el papel, la BD ALUxy. En mi opinión, inicialmente yo intentaría separar:
 - Redo del resto de la BD.
 - Datos de índices.
 - Datos de undo.
 - Datos de temp.
 - SYSTEM de SYSAUX.

Vamos a separar datos, índices y redo; en /u02, /u03 y /u04, respectivamente. Además, el tablespace SYSTEM (que contiene el DD) lo vamos a ubicar con los índices (/u03), separado del SYSAUX (/02); TEMP separado de los datos (/u03); y UNDO_RBS separado de los datos y del redo (/u03). Estamos suponiendo que la mayor actividad de E/S se va a presentar en datos, índices y redo, pero podría no ser así; de hecho una vez creada la BD deberemos revisar el estado de la E/S (iostat).

Resumiendo, los tablespaces de ALUxy quedaría repartidos, por ejemplo, así:

```
/u02/oradata/ALUxy -> DATOS, ESTATICO_DAT, SYSAUX, USERS
/u03/oradata/ALUxy -> INDICES, ESTATICO_IND, SYSTEM,
                    TEMP, UNDO_RBS
/u04/oradata/ALUxy -> Redo
```

2.2. Crear un init.ora, diseñando requerimientos de memoria, suponiendo 4Gb de RAM y 21 instancias con 10 usuarios cada una. Se puede usar un máximo del 64% de la RAM para las SGAs de todas las bases de datos. Los checkpoints sucederán cuando se llene el fichero redolog y, como máximo, cada 30 minutos. La gestión de rollback será automática. Tendremos dos copias del fichero de control en /u02 y /u03.

Solución:

Comprobamos la memoria y su estado con el comando “free -m”:

```
$ free -m
              total          used          free   shared    buffers     cached
Mem:          3928           3898             30         0         120       3041
-/+ buffers/cache:           736         3191
Swap:          6143           253         5889
```

Oracle recomienda destinar un máximo del 80% de la RAM para Oracle, en nuestro caso, el 80% de

3928M son 3142M.

Ahora debemos restar lo q ocupe el núcleo de Oracle:

```
$ size $ORACLE_HOME/bin/oracle
text      data      bss       dec       hex       filename
75275107  288744   121304    75685155  482dd23  /u01/app/oracle/product/10.2.0.1/bin/oracle
```

Por tanto, quitando 71M a los 3142M, nos quedan 3071M. De estos últimos, podemos dedicar el 80% a SGAs (2456M) y el 20% a PGAs (615M). Como tenemos 21 instancias, para cada una de ellas:

- SGA_MAX_SIZE = 2456/21 = 116M. Podemos empezar con SGA_TARGET=100M.
- PGA_AGGREGATE_TARGET = 615/21 = 29M

Como vamos a usar gestión automática de la SGA (con sga_target, y dejando shared_pool_size=db_cache_size=0), y no necesitamos ni Java ni MTS, haremos java_pool_size=0 y large_pool_size=0. Nos queda log_buffer, que le podemos asignar 1M (1048576), y si queremos q los checkpoints sucedan como máximo cada 30minutos (o al llenarse los redolog) usaremos log_checkpoint_timeout=1800 y log_checkpoint_interval=0. Resumiendo, en cuanto a asignación de memoria se refiere:

```
sga_max_size=116M
sga_target=100M
shared_pool_size=0
db_cache_size=0
large_pool=0
java_pool=0
log_buffer=1048576
log_checkpoint_timeout=1800
log_checkpoint_interval=0
```

Por otro lado, partimos del supuesto de que el número de sesiones simultáneas será 10; por tanto fijaremos processes=10 + nºprocesos_background +1. Los procesos background mínimos para Oracle 10g son trece (DBWR, LGWR, SMON, PMON, CKPT, RECO, PSP0, MMAN, MMON, MMML, QMNC, Q000 y Q001); y sumamos uno por el proceso PSEUDO (utilizado para la transición entre procesos); en total tenemos (10+13+1):

```
processes=24
```

Como queremos rollback automático, y si llamamos al tablespace de undo UNDO_RBS, asignamos los parámetros:

```
undo_management=auto
undo_tablespace=undo_rbs
```

El nombre de la base de datos es ALUxy, donde xy es el número que incluye tu nombre de usuario linux (por ejemplo, para curso30 sería ALU30):

```
db_name=ALUxy
```

Como tenemos instalado Oracle10g (10.2.0.2), y queremos utilizar todas sus novedades, añadimos:

```
compatible=10.2.0
```

Para tener dos copias del fichero de control en /u02 y /u03, asignamos:

```
control_files=(/u02/oradata/ALUxy/control1.ctl, /u03/oradata/ALUxy/control2.ctl)
```

Y, finalmente, las trazas las vamos a generar en /u01/app/oracle/admin/ALUxy, en los directorios bdump, cdump y udump; para el alert, cores y trazas de usuario, respectivamente, limitando el

tamaño del fichero a 5Mb (en bloques del SO, 10240):

```
background_dump_dest=/u01/app/oracle/admin/ALUxy/bdump
core_dump_dest=/u01/app/oracle/admin/ALUxy/cdump
user_dump_dest=/u01/app/oracle/admin/ALUxy/udump
max_dump_file_size=10240
```

De esta forma el init podría quedar así:

```
db_name=ALUxy
compatible = 10.2.0
control_files = (/u02/oradata/ALUxy/control1.ctl, /u03/oradata/ALUxy/control2.ctl)
undo_management = auto
undo_tablespace = undo_rbs
background_dump_dest = /u01/app/oracle/admin/ALUxy/bdump
core_dump_dest = /u01/app/oracle/admin/ALUxy/cdump
user_dump_dest = /u01/app/oracle/admin/ALUxy/udump
max_dump_file_size = 10240
sga_max_size = 116M
sga_target = 100M
db_cache_size = 0
shared_pool_size = 0
large_pool_size = 0
java_pool_size = 0
log_buffer = 1048576
log_checkpoint_interval = 0
log_checkpoint_timeout = 1800
processes = 24
```

Lo guardamos como initALUxy.ora, en tu directorio personal \$HOME.

2.3. Crear la BD que hemos diseñado anteriormente, y para la que también hemos creado un init.ora. Hacerlo manualmente con CREATE DATABASE. El nombre de la BD será ALUxy. Antes de lanzar el comando de creación, ponte de acuerdo con un compañero para que no lo hagais a la vez.

Solución:

Con el comando CREATE DATABASE vamos a crear una BD con los tablespaces básicos: SYSTEM, TEMP y UNDO_RBS (y uno “de trabajo” para usuarios q lo necesiten: USERS); más dos copias del fichero de control y tres ficheros redolog (tres grupos de un sólo miembro cada uno).

Según el diseño anterior, la distribución de ficheros sería la siguiente:

```
/u02/oradata/ALUxy -> SYSAUX, USERS, control1
/u03/oradata/ALUxy -> SYSTEM, TEMP, UNDO_RBS, control2
/u04/oradata/ALUxy -> Redo
```

Debemos tener el fichero init.ora anteriormente creado en:

\$HOME/initALUxy.ora

El comando CREATE DATABASE podría quedar así:

```

CREATE DATABASE "ALUxy"
  maxdatafiles 254
  maxinstances 1
  maxlogfiles 32
  character set WE8ISO8859P15
DATAFILE '/u03/oradata/ALUxy/system01.dbf' SIZE 260M
  AUTOEXTEND ON NEXT 10M MAXSIZE 400M
  EXTENT MANAGEMENT LOCAL
SYSAux DATAFILE '/u02/oradata/ALUxy/sysaux01.dbf' SIZE 10M
  AUTOEXTEND ON NEXT 10M MAXSIZE 400M
UNDO TABLESPACE UNDO_RBS
  DATAFILE '/u03/oradata/ALUxy/undo_rbs01.dbf' SIZE 10M
  AUTOEXTEND ON NEXT 1M MAXSIZE 20M
DEFAULT TABLESPACE USERS
  DATAFILE '/u02/oradata/ALUxy/users01.dbf' SIZE 10M
  AUTOEXTEND ON NEXT 1M MAXSIZE 20M
  segment space management auto
DEFAULT TEMPORARY TABLESPACE TEMP
  TEMPFILE '/u03/oradata/ALUxy/temp01.dbf' SIZE 10M REUSE
  AUTOEXTEND ON NEXT 10M MAXSIZE 50M
  EXTENT MANAGEMENT LOCAL UNIFORM SIZE 64K
logfile '/u04/oradata/ALUxy/redo01.log' SIZE 10M,
  '/u04/oradata/ALUxy/redo02.log' SIZE 10M,
  '/u04/oradata/ALUxy/redo03.log' SIZE 10M;

```

Asignamos la variable de entorno ORACLE_SID=ALUxy, para que Oracle sepa el nombre de la instancia que vamos a crear:

```

/home/CURSO/cursoXY (CURSOxy)> export ORACLE_SID=ALUxy
/home/CURSO/cursoXY (ALUxy)>

```

Antes de crear la BD debemos crear los directorios necesarios:

```

$ mkdir /u02/oradata/ALUxy
$ mkdir /u03/oradata/ALUxy
$ mkdir /u04/oradata/ALUxy
$ mkdir /u01/app/oracle/admin/ALUxy
$ mkdir /u01/app/oracle/admin/ALUxy/bdump
$ mkdir /u01/app/oracle/admin/ALUxy/cdump
$ mkdir /u01/app/oracle/admin/ALUxy/udump

```

Antes de crear la BD, debemos dar permiso de escritura al usuario oracle (bajo el q correrán los procesos de la instancia) sobre los directorios que hemos creado. Lo haremos dando permiso de escritura al grupo:

```

$ chmod g+w /u0?/oradata/ALUxy
$ chmod -R g+w /u01/app/oracle/admin/ALUxy

```

Ahora sí que podemos crear la BD. **Pregunta a tu compañero si ha iniciado el proceso de creación, de forma que si lo ha hecho debes esperarte a que el suyo termine:**

```

/home/CURSO/cursoXY (ALUxy)> sqlplus
...
Enter user-name: / as sysdba
Connected to an idle instance.
SQL> startup nomount pfile=initALUxy.ora
ORACLE instance started.
Total System Global Area  121634816 bytes
Fixed Size                 1259504 bytes
Variable Size              71305232 bytes

```

Administración Avanzada de Oracle10g

```
Database Buffers          46137344 bytes
Redo Buffers              2932736 bytes
```

```
SQL> CREATE DATABASE "ALUxy"
      maxdatafiles 254
      maxinstances 1
      maxlogfiles 32
      character set WE8ISO8859P15
DATAFILE '/u03/oradata/ALUxy/system01.dbf' SIZE 260M
      AUTOEXTEND ON NEXT 10M MAXSIZE 400M
      EXTENT MANAGEMENT LOCAL
SYSAUX DATAFILE '/u02/oradata/ALUxy/sysaux01.dbf' SIZE 10M
      AUTOEXTEND ON NEXT 10M MAXSIZE 400M
UNDO TABLESPACE UNDO_RBS
      DATAFILE '/u03/oradata/ALUxy/undo_rbs01.dbf' SIZE 10M
      AUTOEXTEND ON NEXT 1M MAXSIZE 20M
DEFAULT TABLESPACE USERS
      DATAFILE '/u02/oradata/ALUxy/users01.dbf' SIZE 10M
      AUTOEXTEND ON NEXT 1M MAXSIZE 20M
      segment space management auto
DEFAULT TEMPORARY TABLESPACE TEMP
      TEMPFILE '/u03/oradata/ALUxy/temp01.dbf' SIZE 10M REUSE
      AUTOEXTEND ON NEXT 10M MAXSIZE 50M
      EXTENT MANAGEMENT LOCAL UNIFORM SIZE 64K
logfile '/u04/oradata/ALUxy/redo01.log' SIZE 10M,
        '/u04/oradata/ALUxy/redo02.log' SIZE 10M,
        '/u04/oradata/ALUxy/redo03.log' SIZE 10M;
Database created.
```

Vamos a comprobar los ficheros que componen las BD que hemos creado:

```
SQL> select name from v$datafile;
NAME
```

```
-----
/u03/oradata/ALUxy/system01.dbf
/u03/oradata/ALUxy/undo_rbs01.dbf
/u02/oradata/ALUxy/sysaux01.dbf
/u02/oradata/ALUxy/users01.dbf
```

```
SQL> select name from v$tempfile;
NAME
```

```
-----
/u03/oradata/ALUxy/temp01.dbf
```

```
SQL> select member from v$logfile;
MEMBER
```

```
-----
/u04/oradata/ALUxy/redo01.log
/u04/oradata/ALUxy/redo02.log
/u04/oradata/ALUxy/redo03.log
```

```
SQL> select name from v$controlfile;
NAME
```

```
-----
/u02/oradata/ALUxy/control11.ctl
/u03/oradata/ALUxy/control2.ctl
```

Finalmente, cerramos la BD:

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
```

Las vistas del DD no las vamos a crear porque tardan unos 15 minutos, y habría que hacerlo uno por uno (alumno por alumno). Como sois 20 alumnos, necesitaríamos 300 minutos (5 horas) para completarlo.

2.4. Crear la misma BD anterior, esta vez usando OMF, y cambiando el nombre por OMFxy, creando los ficheros bajo /u02/oradata/OMFxy y usando dos destinos para redolog (en /u03 y /u04).

Solución:

En primer lugar, creamos el init.ora en nuestro directorio particular, copia del anterior, pero cambiando ALU por OMF:

```
$ cp /home/CURSO/cursoXY/initALUxy.ora /home/CURSO/cursoXY/initOMFxy.ora
```

Tenemos que cambiar algunas cosas, fundamentalmente añadir los parámetros de activación de OMF: db_create_file_dest, db_create_online_log_dest_1 y db_create_online_log_dest_2; y modificar db_name (también cambiar ALU por OMF en todas las líneas donde aparezca):

```
db_name=OMFxy
compatible = 10.2.0
control_files = (/u02/oradata/OMFxy/control1.ctl,
/u03/oradata/OMFxy/control2.ctl)undo_management = auto
undo_tablespace = undo_rbs
DB_CREATE_FILE_DEST=/u02/oradata/OMFxy
DB_CREATE_ONLINE_LOG_DEST_1=/u03/oradata/OMFxy
DB_CREATE_ONLINE_LOG_DEST_2=/u04/oradata/OMFxy
background_dump_dest = /u01/app/oracle/admin/OMFxy/bdump
core_dump_dest = /u01/app/oracle/admin/OMFxy/cdump
user_dump_dest = /u01/app/oracle/admin/OMFxy/udump
max_dump_file_size = 10240
sga_max_size = 116M
sga_target = 100M
db_cache_size = 0
shared_pool_size = 0
large_pool_size = 0
java_pool_size = 0
log_buffer = 1048576
log_checkpoint_interval = 0
log_checkpoint_timeout = 1800
processes = 24
```

Cambiamos la variable ORACLE_SID=OMFxy:

```
$ export ORACLE_SID=OMFxy
/home/CURSO/cursoXY (OMFxy)>
```

Vamos a crear los directorios necesarios para los ficheros de la BD:

```
/home/CURSO/cursoXY (OMFxy)> mkdir /u02/oradata/OMFxy
/home/CURSO/cursoXY (OMFxy)> mkdir /u03/oradata/OMFxy
/home/CURSO/cursoXY (OMFxy)> mkdir /u04/oradata/OMFxy
/home/CURSO/cursoXY (OMFxy)> mkdir /u01/app/oracle/admin/OMFxy
/home/CURSO/cursoXY (OMFxy)> mkdir /u01/app/oracle/admin/OMFxy/bdump
/home/CURSO/cursoXY (OMFxy)> mkdir /u01/app/oracle/admin/OMFxy/cdump
/home/CURSO/cursoXY (OMFxy)> mkdir /u01/app/oracle/admin/OMFxy/udump
```


Antes de crear la BD, debemos dar permiso de escritura al usuario oracle sobre los directorios que hemos creado. Lo haremos dando permiso de escritura al grupo:

```
$ chmod g+w /u02/oradata/OMFxy
$ chmod -R g+w /u01/app/oracle/admin/OMFxy
```

Ahora construimos la sentencia de creación de la BD, que es muy simple al usar OMF:

```
CREATE DATABASE OMFxy
DEFAULT TABLESPACE USERS
DEFAULT TEMPORARY TABLESPACE TEMP
UNDO TABLESPACE UNDO_RBS;
```

Vamos a crear la BD. Antes, igual que en el caso anterior, ponte **de acuerdo con tu compañero de forma que no lancéis el comando de creación a la vez.**

```
home/CURSO/cursoXY (OMFxy)> sqlplus
```

```
...
Enter user-name: / as sysdba
Connected to an idle instance.
SQL> startup nomount pfile=initOMFxy.ora
ORACLE instance started.
Total System Global Area 121634816 bytes
Fixed Size 1259504 bytes
Variable Size 71305232 bytes
Database Buffers 46137344 bytes
Redo Buffers 2932736 bytes
```

```
SQL> show parameter db_create
```

NAME	TYPE	VALUE
db_create_file_dest	string	/u02/oradata/OMFxy
db_create_online_log_dest_1	string	/u03/oradata/OMFxy
db_create_online_log_dest_2	string	/u04/oradata/OMFxy

```
SQL> CREATE DATABASE "OMFxy"
2 DEFAULT TEMPORARY TABLESPACE TEMP
3* UNDO TABLESPACE UNDO_RBS;
Database created.
```

```
SQL> select name,bytes from v$datafile;
NAME
```

```
-----
BYTES
-----
/u02/oradata/OMFxy/datafile/o1_mf_system_2tsng5vm_.dbf
104857600
/u02/oradata/OMFxy/datafile/o1_mf_undo_rbs_2tsngdyk_.dbf
10485760
/u02/oradata/OMFxy/datafile/o1_mf_sysaux_2tsngf8b_.dbf
104857600
/u02/oradata/OMFxy/datafile/o1_mf_users_2tsngjb4_.dbf
104857600
```

```
SQL> select name,bytes from v$tempfile;
NAME
```

```
-----
BYTES
-----
/u02/oradata/OMFxy/datafile/o1_mf_temp_2tsngj5k_.tmp
104857600
```

```
SQL> SELECT name FROM V$CONTROLFILE;
```

NAME

```
-----  
/u02/oradata/OMFxy/control11.ctl  
/u03/oradata/OMFxy/control12.ctl
```

```
SQL> select a.member, b.bytes from v$logfile a, v$log b where a.group#=b.group#;  
MEMBER
```

```
-----  
      BYTES
```

```
-----  
/u03/oradata/OMFxy/onlinelog/o1_mf_1_2tsnfvor_.log  
104857600  
/u04/oradata/OMFxy/onlinelog/o1_mf_1_2tsnfw1b_.log  
104857600  
/u03/oradata/OMFxy/onlinelog/o1_mf_2_2tsnfwdh_.log  
104857600  
/u04/oradata/OMFxy/onlinelog/o1_mf_2_2tsnfyx7_.log  
104857600
```

Como se observa, Oracle ha creado automáticamente el directorio OMFxy bajo el que hemos indicado con `db_create_file_dest`, por lo q habría bastando con asignar la variable `db_create_file_dest=/u02/oradata`.

Finalmente cerramos la BD, como en el ejercicio anterior, con “shutdown immediate”:

```
SQL> shutdown immediate  
Database closed.  
Database dismounted.  
ORACLE instance shut down.
```

2.5. Consultar el “Simulador de DBCA”.

Se trata de un documento en PDF con las capturas de pantalla de la ejecución de una sesión del DataBase Configuration Assitant (dbca):

<http://cursos.atica.um.es/DBA10g2/pdf/contenidos/SimulacionDBCA.pdf>