

Prácticas de

Administración de Bases de Datos

Grado en Ingeniería Informática

**PRÁCTICA 2**

Arquitectura del SGBD Oracle

(inicio y parada de la base de datos, estructura física)

**Ejercicios**

# Inicio y parada de la base de datos

1. Desde SQL\*Plus, parar la base de datos ONUBA. Para saber si la base de datos se ha detenido correctamente deben aparecer los mensajes: “Base de datos cerrada”, “Base de datos desmontada” e “Instancia ORACLE cerrada”

**Para pararla dependerá de lo que se requiera, los comandos a utilizar serán los siguientes:**

**En la CMD 🡪 sqlplus /NOLOG**

**En SQLPlus 🡪 CONNECT / AS SYSDBA**

**SHUTDOWN IMMEDIATE**

1. Desde SQL\*Plus, arrancar de nuevo la instancia ONUBA, montando y abriendo la base de datos.

STARTUP MOUNT;

ALTER DATABASE OPEN;

1. Puesto que las operaciones de arrancar y parar una instancia son realizadas por el administrador con cierta frecuencia, la mejor opción es diseñar unos scripts. Diseñar un script que arranque la base de datos ONUBA y otro que la pare. Para lanzar estos scripts es necesario crear un fichero .sql con las sentencias a ejecutar y un fichero por lotes (.bat) que contenga la llamada al SQL\*Plus que ejecute el fichero

.sql

Ejemplo:

C:\app\Uhu\product\11.2.0\dbhome\_1\BIN\sqlplus /nolog @C:\app\Uhu\scripts\initONUBA.sql

esta sentencia corresponde al fichero por lotes (**.bat)**, la cual realiza una llamada al programa *sqlplus* sin usuario (/nolog) y ejecuta el script initONUBA.sql.

Fichero initOnuba.bat

sele

1. Comprobemos las distintas fases del arranque de una base de datos.
   * Arrancar sólo la instancia (NOMOUNT) y consultar (con SQL\*Plus) algún parámetro con **show parameter nombre\_parámetro**. ¿Qué ocurre al acceder a la vista V$CONTROLFILE y por qué?

**Con el parámetro por ejemplo “memory\_max\_target” al poner la sentencia:**

**show parameter memory\_max\_target**

**La base de datos es capaz de mostrar el valor del parámetro ya que esta accediendo a un fichero binario. No obstante, si intentamos acceder a la vista V$CONTROLFILE:**

**select \* from v$controlfile;**

**La base de datos no es capaz de mostrar nada porque no esta montada la instancia Onuba, entonces no hay nada que mostrar.**

* + Montar la BD y volver a consultar V$CONTROLFILE. ¿Qué sucede al consultar la tabla DBA\_USERS?

**Montamos la BD con la sentencia:**

**alter database mount;**

**Ahora al consultar la vista V$CONTROLFILE de nuevo, si muestra contenido.**

**Al consultar la vista DBA\_USERS con la siguiente sentencia:**

**select \* from dba\_users;**

**nos dice que la base de datos no esta abierta, y que por tanto solo se permiten consultas en tablas/vistas fijas.**

* + Abrir la BD y volver a consultar la tabla DBA\_USERS

**Abrimos la BD con:**

**alter database open;**

**Cuando consultamos de nuevo DBA\_USERS, ahora si muestra todo el contenido de DBA\_USERS.**

1. Cerrar la BD y abrirla en modo READ ONLY. Crear una tabla y comprobar qué ocurre. Activar el modo READ WRITE y volver a crear la tabla.

**Con las sentencias:**

**startup mount;**

**alter database open read only;**

**Abrimos la base de datos en modo de solo lectura.**

**Intentaremos crear una tabla.**

1. Veamos ahora cómo se puede restringir el acceso a los usuarios normales y que sólo accedan los administradores.
   * Arrancar la base de datos en modo restringido
   * Intentar conectaros desde el usuario ***scott***
   * Sin parar la base de datos, modificarla para que no esté restringida y volver a intentar conectaros con el usuario ***scott***

# Fichero de Control

1. Localizar el fichero de control desde el S.O y desde la propia BD.
2. Consultar la información de la BD relativa al contenido del fichero de control (V$DATABASE). Forzar un checkpoint (*alter system checkpoint*) y volver a consultar las columnas relativas al fichero de control.
3. Añadir una réplica del fichero de control de la BD en

$ORACLE\_BASE\oradata\$ORACLE\_SID. Comprobar que se ha hecho correctamente a nivel de sistema operativo y a nivel de base de datos

# Ficheros de Rehacer

1. Localizar los ficheros *Redo Log* de la BD. ¿Cuántos grupos hay y cuántos miembros tiene cada grupo? ¿están correctamente distribuidos?
2. Comprobar cual es el fichero *Redo Log* activo. ¿Qué ocurre al forzar un *log switch* (ALTER SYSTEM SWITCH LOGFILE;)? ¿y al forzar un *checkpoint* (ALTER SYSTEM CHECKPOINT;)?
3. Añadir un miembro más a cada grupo: "redo11.log" al Grupo 1, "redo12.log" al Grupo 2 y "redo13.log" al Grupo 3. Situarlo en una carpeta simulando que se están guardando en un disco distinto. Consultar las vistas V$LOGFILE y V$LOG
4. Añadir un nuevo grupo (grupo 4) con dos miembros. Un miembro estará en la ubicación por defecto y el otro en la ubicación que hayáis elegido en el ejercicio

anterior. Los ficheros se llamarán "redo04.log" y "redo14.log". El tamaño del nuevo grupo debe ser el mismo que el de los grupos 1, 2 y 3

1. Con este ejercicio vamos a aprender a aumentar el tamaño de los Redo Log. Para ello hay que crear los nuevos grupos con el tamaño deseado y eliminar los antiguos.
   * Añadir los grupos 5, 6 y 7 con 2 miembros cada uno y con un tamaño de 100M. La ubicación de los miembros debe ser la misma que en los ejercicios anteriores
   * Eliminar los miembros del grupo 1, de uno en uno. ¿Qué ocurre al eliminar el último?
   * Eliminar los grupos 1, 2, 3 y 4. ATENCIÓN: no se puede eliminar el grupo ACTIVO (current) ni los grupos que estén activos (ACTIVE), que están esperando un checkpoint
2. Cambiar el nombre de los miembros de redo de los grupos 5, 6 y 7 a redo5a.log, redo5b.log, redo6a.log, redo6b.log, redo7a.log y redo7b.log. Comprobar los cambios en la vista V$LOGFILE