

Manual de Prácticas

Administración de Base de Datos

Especialidad Base de Datos

Elaborado por:

M.C. Martha Escamilla Zepeda

Febrero 2010

CONTENIDO

<i>Objetivos Generales del Manual de Prácticas</i>	03
<i>Práctica 1. Conceptos básicos de Linux</i>	04
<i>Práctica 2. Creación de una Base de Datos</i>	09
<i>Práctica 3. Arquitectura de una Base de Datos</i>	23
<i>Práctica 4. Administración de Espacio en Disco</i>	35
<i>Práctica 5. Administración de Usuarios</i>	47
<i>Práctica 6. Manejando Undo Data</i>	65
<i>Práctica 7. Performance y Monitoreo</i>	86
<i>Práctica 8. Respaldo y Recuperación</i>	99
<i>Práctica 9. Moviendo Datos</i>	109

Objetivos Generales del Manual de Prácticas

- ❖ Este Manual de prácticas apoyará al estudiante bajo el nuevo Modelo Educativo por Competencias para mejorar el aprendizaje significativo, dejando que los estudiantes den lo mejor de sí mismo y desarrollar las habilidades prácticas inmediatamente después de adquirir el conocimiento teórico, para su mejor desarrollo Profesional.
- ❖ Desarrollar un manual de prácticas aplicado la tecnología que el mercado actual requiere en la administración y manejo de Base de Datos, en este caso aplicada al manejador de Base de Datos Oracle10g, apoyando a las asignaturas que competen a esta especialidad.
- ❖ Es importante que esta asignatura cuente con un manual de prácticas para poder asimilar de manera más efectiva y eficiente los conocimientos teóricos vistos en el aula de clase, y bajo una guía clara que le permita al mismo tiempo retroalimentar esa información adquirida con la práctica real.
- ❖ Cumplir con los objetivos educacionales desarrollados para el programa de estudios por Competencias.

Beneficios que aportará

- ❖ En Octubre del 2008 la Institución firmó convenio con Oracle Academy para integrar sus programas a los planes de estudio de la Especialidad de Base de Datos. Este documento será concluido y podrá aplicarse a partir del semestre agosto-diciembre del 2010.
- ❖ El Instituto Tecnológico de Toluca contará con un documento que acreciente el acervo bibliográfico como apoyo a los estudiantes y un archivo electrónico al que podrá tener acceso cualquier profesor o estudiante.
- ❖ Contar con un manual de prácticas estructuradas que cubran los temas y subtemas de la Asignatura.
- ❖ Este manual será un apoyo didáctico para los profesores que impartan esta asignatura en un futuro, reafirmando y experimentando los conocimientos vistos en clase.
- ❖ Este manual de prácticas será una guía y servirá de retroalimentación al estudiante para asimilar y comprobar de manera eficiente los conocimientos teóricos aprendidos en el aula.
- ❖ Fortalecerá los conocimientos teóricos que llevados a la práctica son más perdurables, para que en el momento que el estudiante inicie su vida profesional llegue reforzado con toda esta experiencia adquirida dentro de la misma Institución, pues no debe olvidarse que “la práctica hace al maestro”.

- ❖ *Paralelamente se ofrecen las certificaciones para estos programas de estudios, dando la oportunidad al estudiante a obtenerla antes de egresar.*

Martha Escamilla Zepeda

Práctica 01

Conceptos Básicos Linux

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la administración de usuarios básica y configuración del servidor.
- Guiar paso a paso en la configuración y generación de usuarios y del servidor.
- Guiar al estudiante para que pueda realizar la configuración del servidor para permitirle el acceso a través de la red.
- Crear grupos de trabajo y usuarios, así como la asignación de perfiles y privilegios específicos.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Práctica inicial como suplemento de apoyo al temario

Subtema: Conceptos Unix

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Configuración de la red estática
- Monitoreo del Sistema Operativo
- Creación de usuarios
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.
- Siempre se manejarán comandos de Linux o estatutos de SQL.

Desarrollo de la práctica

1. Conectarse como usuario root

```
Login: root
Password:
#
```

Cambiar la dirección IP por 192.168.100.20 y el nombre de host por server1

```
# vi /etc/hosts
```

Editar archivo de configuración de los hosts que queremos agregar al Sistema y agregar la siguiente línea

```
192.168.100.20 server20
```

Levantar la dirección IP 192.168.100.20 en el puerto Ethernet 0 y cambiar el nombre del host por server 20 de la siguiente forma

```
# ifconfig eth0 192.168.100.20 up
#hostname server20
```

Verifica que la dirección IP ingresada esté activa

```
# ifconfig
```

Verifica el nombre del host actual. Se prueba que la nueva configuración funcione.

```
# hostname
#ping server20
```

2. Crear usuario abd01 por medio de los comandos, y crear usuario abd02 modificando los archivos de configuración. Ver los cambios generados.

Agregamos un usuario y le asignamos un password

```
# useradd abd01
# passwd abd01
```

Nos conectamos a la cuenta del nuevo usuario, verificamos su home y que en verdad nos conectamos a esa cuenta, salimos de la cuenta.

```
#su – abd01
$ pwd
$ whoami
$ logout
```

Creando usuario vía archivos de configuración

Editar archivo group para crear un nuevo grupo abd02 para el nuevo usuario, agregando lo siguiente al final de dicho archivo

```
# vi /etc/group
...abd02:x:505:
...
```

Nota: 505 se refiere al id del grupo y puede variar.

Editar archivo passwd para agregar el nuevo usuario abd02, agregar lo siguiente

```
# vi /etc/passwd
abd02:x:505:505::/home/abd02:/bin/bah
```

El primer número 505 indica el id del usuario y el Segundo el id del grupo. La dirección /home/abd02 indica el home del usuario y la dirección /bin/bash el shell de comandos que va utilizar el usuario abd02.

Editar el archivo shadow para crear una contraseña o en su default dejarla por default, agregar

```
#vi /etc/shadow
abd02:!!:.....
```

Los signos !! Indican la contraseña por default.

Creamos directorio de home del usuario y cambiamos el propietario y el grupo de dicho home por el del usuario abd02 y el grupo abd02.

```
#mkdir /home/abd02
#chown abd02.abd02 /home/abd02
```

Ahora copiaremos o crearemos los siguientes archivos que deben encontrarse debajo del home de cualquier usuario sobre el home de abd02

```
.bash_profile
.bash_logout
.bashrc
```

Nota: *En caso de crear los archivos verificar el contenido que deben llevar cada uno. Cambia los permisos de los archivos encontrados bajo el home de abd02.*

```
#cd /home/abd02
```

```
#chown abd02.abd02 .bash_profile .bash_logout .bashrc
```

Nos conectamos a la cuenta del nuevo usuario, verificamos su home y que en verdad nos conectamos a esa cuenta, salimos de la cuenta.

```
#su - abd02  
$pwd  
$whoami  
$logout
```

3. Verificar la versión del software instalado y su versión.

Verificamos los paquetes con extensión rpm que fueron instalados junto con el Sistema, en donde nos aparecerá una lista con el nombre del paquete y su versión, con la instrucción more podemos ir checando uno por uno.

```
#rpm-qa | more
```

4. Verificar la versión del kernel, distribución y distribución de linux por medio de la línea de comandos.

```
#uname -sr
```

Este comando muestra el nombre del kernel (s) así como su versión (r), para ver todas las opciones del kernel checar `uname -a`.

En este caso como tenemos instalado fedora, debemos buscar en el directorio /etc el directorio fedora-release y mostrar su contenido, en el ejemplo se usa cat.

```
#cat /etc/fedora-release
```

5. Monitorear los procesos que están en ejecución y el software que está levantado.

```
#ps -e | more
```

El comando `ps` sirve para ver los procesos y la opción `-e` indica que se muestren todos los procesos en ejecución, indicando el id del proceso, la terminal en la que se encuentra, el tiempo en el que se ejecutó antes de mostrarse la lista y el nombre del proceso o software.

Nota: Es importante leer el manual de `ps` que viene en el Sistema para poder ver las distintas opciones de este comando.

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Es importante que se estudien con mayor detalle los comandos utilizados durante esta práctica y es recomendable utilizar el comando “man” seguido del nombre del comando que se desea verificar, para que se muestre una descripción amplia de las características del comando. Entregar Reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Monitorear usuarios, grupos y características de cada uno*
- 3. Cuales usuarios pertenecen al Sistema Operativo y cuales son del usuario*
- 4. Investigar cómo trabajar en modo gráfico y modo texto.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 02

Creación de una Base de Datos

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la configuración y creación de una instancia de base de datos de manera manual permitiendo un mejor entendimiento de los archivos y procesos involucrados para esta tarea.
- Guiar paso a paso en la configuración y creación de instancia y una base de datos.
- Guiar al estudiante para que pueda crear los archivos de configuración, variables de ambiente y procedimientos requeridos en la creación de la base de datos.
- Aprenderá también a monitorear la instancia de la base de datos y procesos activados.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Configurando Network Environment

Subtema(s): Listeners/ Net Services y conectividad

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Preferentemente trabajar bajo Sistema Operativo Unix (Cualquier Arquitectura)
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Creación de usuarios
- Creación y edición del archivo de parámetros de inicialización
- Creación de una instancia y bases de datos
- Siempre se trabajará con comandos de Linux o estatutos de SQL y Esquema gráfico de Enterprise Manager
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.

Martha Escamilla Zepeda

Desarrollo de la práctica

Ejercicio 1.

1. Bajar el software Oracle 10g. Usando imágenes de cd. Si tienes los cd's de instalación de Oracle entonces móntalos de esta forma

```
# mount -r -o loop=/dev/loop0 ora10gR2.iso Disk1
```

2. Descargando los paquetes. Otra forma es descargarlos de:

<http://www.oracle.com/technology/software/products/database/oracle10g/htdocs/10201linuxsoft.html>

Extracción de los paquetes. Para extraer los paquetes realiza lo siguiente, ubicándote en el directorio donde deseas extraer los archivos

```
# unzip 10201_database_linux32.zip
```

Nos crea una carpeta llamada "db/Disk1" o "database"

3. Crear usuario y grupo

Debes conectarte como root y ejecutar lo siguiente. Así agregarás un grupo dba, crearás un usuario oracle que pertenecerá a dba que tendrá home bajo /opt/oracle y le asignarás una contraseña al usuario oracle.

```
# groupadd dba
# useradd -d /opt/oracle -g dba oracle
# passwd oracle
```

4. Agregar variables de ambiente

Conectarse como oracle y editar el archivo .bash_profile ubicado en el home de oracle.

```
# su - oracle
$ vi .bash_profile

### Agregar las Variables de ambiente para Oracle 10g ###

ORACLE_BASE=/opt/oracle
ORACLE_HOME=$ORACLE_BASE/10gR2
ORACLE_SID=MY_ORACLE
LD_LIBRARY_PATH=$ORACLE_HOME/lib
PATH=$PATH:$ORACLE_HOME/bin

export ORACLE_BASE ORACLE_HOME ORACLE_SID LD_LIBRARY_PATH PATH
```

Guarda el archivo `.bash_profile` y ejecuta el comando para cargar las nuevas variables de entorno.

```
$ . .bash_profile
```

5. Instalar librerías `compat-gcc` y paquetes necesarios

Terminar primero la sesión de `oracle($)` y verificar que te encuentres como `root(#)`. Instalar las librerías `compat-gcc`. Verifica que los paquetes estén instalados

```
# rpm -q setarch- tcl- libXau-devel- libXp- compat-db- compat-libstdc++-33
compat-libf2c-34- compat-gcc-34- libaio- compat-gcc-34-c++- compat-libstdc++-
296 compat-libgcc-296
```

En caso de no tener instaladas algunas de las librerías, **descarga las versiones correspondientes**, e instalarlas con el siguiente comando: **`rpm -ivh <nombre_del_paquete>`**. De ser necesario debes forzar la instalación de estas librerías para que no interfieran con versiones más recientes que son necesarias para otros programas.

Ejemplo:

```
# rpm -ivh setarch-2.0-1.1.i386.rpm
# rpm -ivh tcl-8.4.13-3.fc6.i386.rpm
# rpm -ivh libXau-devel-1.0.1-3.1.i386.rpm
# rpm -ivh libXp-1.0.0-8.i386.rpm
# rpm -ivh compat-db-4.2.52-5.1.i386.rpm
# rpm -ivh compat-libstdc++-33-3.2.3-61.i386.rpm
# rpm -ivh compat-libf2c-34-3.4.6-4.i386.rpm
# rpm -ivh compat-gcc-34-3.4.6-4.i386.rpm
# rpm -ivh libaio-0.3.106-3.2.i386.rpm
# rpm -ivh compat-gcc-34-c++-3.4.6-4.i386.rpm
# rpm -ivh compat-libstdc++-296-2.96-139.i386.rpm
# rpm -ivh compat-libgcc-296-2.96-139.i386.rpm
```

Instala el paquete requerido adicionalmente usando el mismo comando

```
# rpm -ivh openmotif21-2.1.30-14.i386.rpmA
```

6. Cambiar parámetros del kernel

Editar el archivo `sysctl.conf`

```
# vi /etc/sysctl.conf
#Agregar
kernel.shmall = 2097152
```

```

kernel.shmmax                =                2147483648
kernel.shmmni                =                4096
kernel.sem                   =      250      32000      100      128
fs.file-max                  =                65536
net.ipv4.ip_local_port_range =                1024      65000
net.core.rmem_default=262144
net.core.rmem_max=262144
net.core.wmem_default=262144
net.core.wmem_max=262144

```

Nota: Es necesario ejecutar "sysctl -p" o reiniciar el sistema para aplicar los ajustes.

```
# /sbin/sysctl -p
```

Editar el archivo limits.conf

```

# vi /etc/security/limits.conf

###Agregar:

* soft nproc2047
* hard nproc16384
* soft nofile1024
* hard nofile65536

```

Editar el archivo login. Editamos el archivo login y agregamos la siguiente línea al archivo login, si es que no existe

```

#vi /etc/pam.d/login

###agregamos si es que no existe###
session required /lib/security/pam_limits.so

```

7. Deshabilitar SELINUX

Deshabilitamos la seguridad de Linux editando el siguiente archivo

```

# vi /etc/selinux/config

###editamos
SELINUX=disabled

```

Dado que el software de Oracle Database 10g no tiene soporte directo para las distribuciones de Fedora Core, y es requerido por el instalador reconocer el SO que se tiene, editamos el siguiente archivo

```
# vi /etc/redhat-release

### Replezamos la línea
Fedora Core release 6 (Zod)
### por
redhat-4
```

8. Configurar el DISPLAY

Para que se pueda ejecutar el instalador de oracle, estando como root, deberás hacer lo siguiente

```
#xhost +<nombre_servidor>
```

Por default es localhost, si no se agrega nada después del signo + se permite acceso a todos los puertos.

Nos conectamos como el usuario oracle y exportamos la variable DISPLAY

```
# su - oracle
$ export DISPLAY=<nombre_servidor>:0.0
```

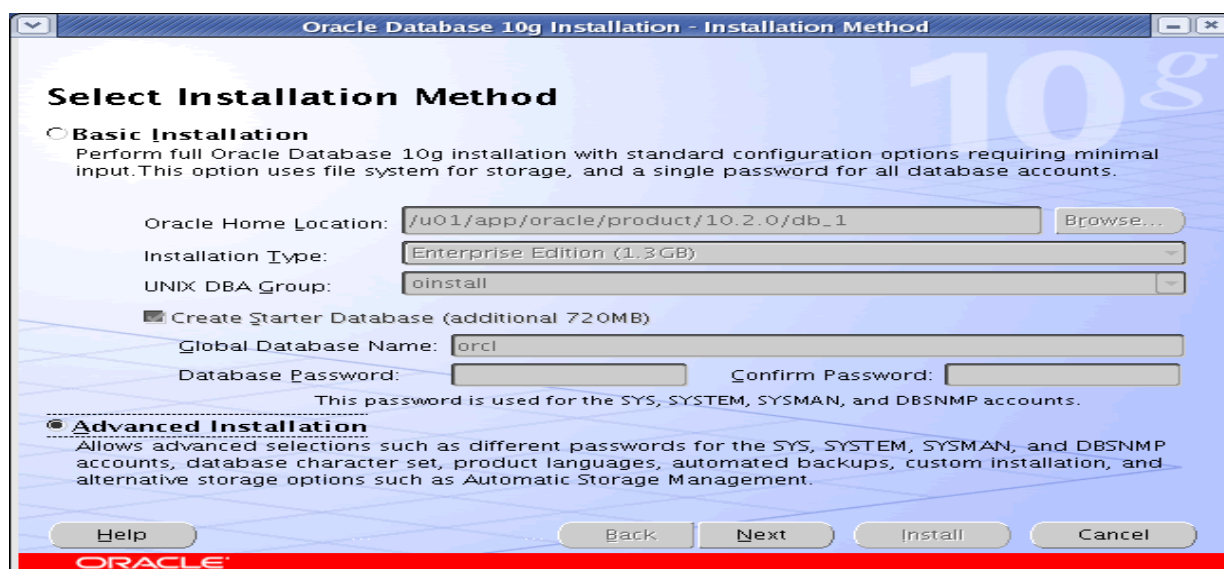
9. Iniciar la instalación

Iniciar el instalador gráfico. Ya hecho correctamente lo anterior, ejecutar conectado como usuario oracle (\$) y ubicado bajo "db/Disk1" o "database"(En la ruta donde descomprimiste los paquetes), lo siguiente

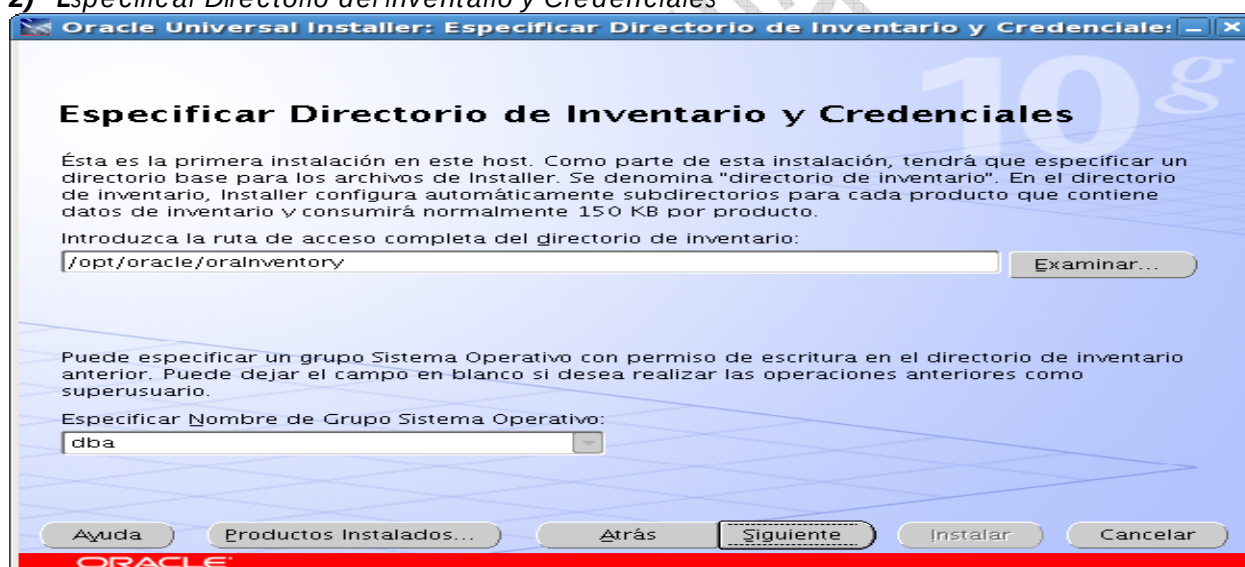
```
$ ./runInstaller
```

Una vez iniciado la interfaz gráfica de instalación sólo deberás seguir los pasos que se te van pidiendo.

1) Seleccionamos el Método de Instalación Avanzado



2) Especificar Directorio del Inventario y Credenciales



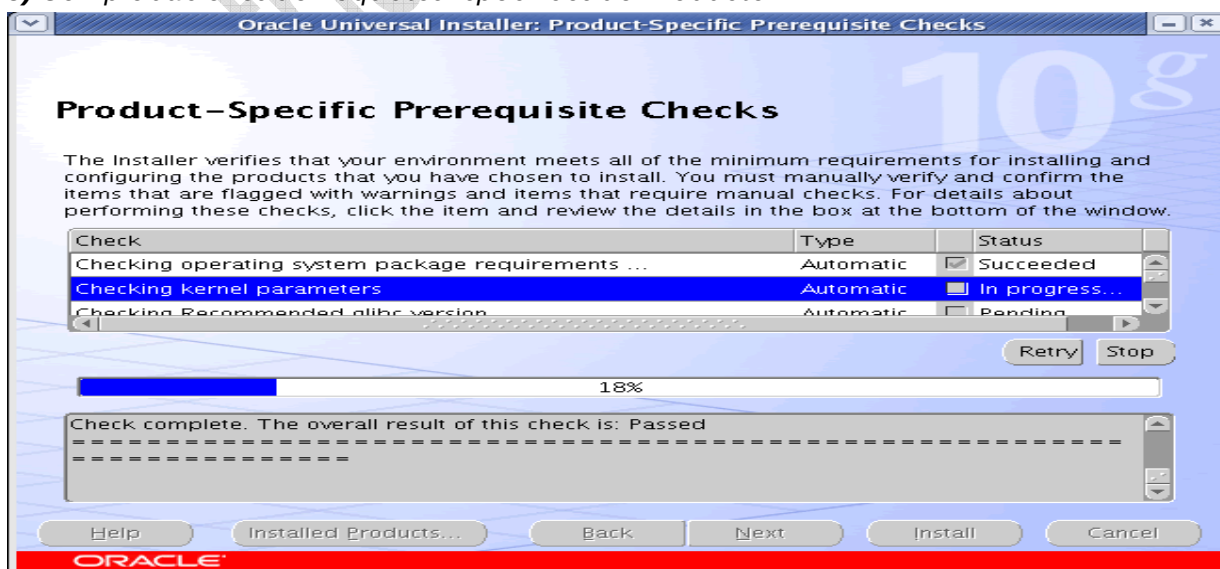
3) Seleccionar el Tipo de Instalación



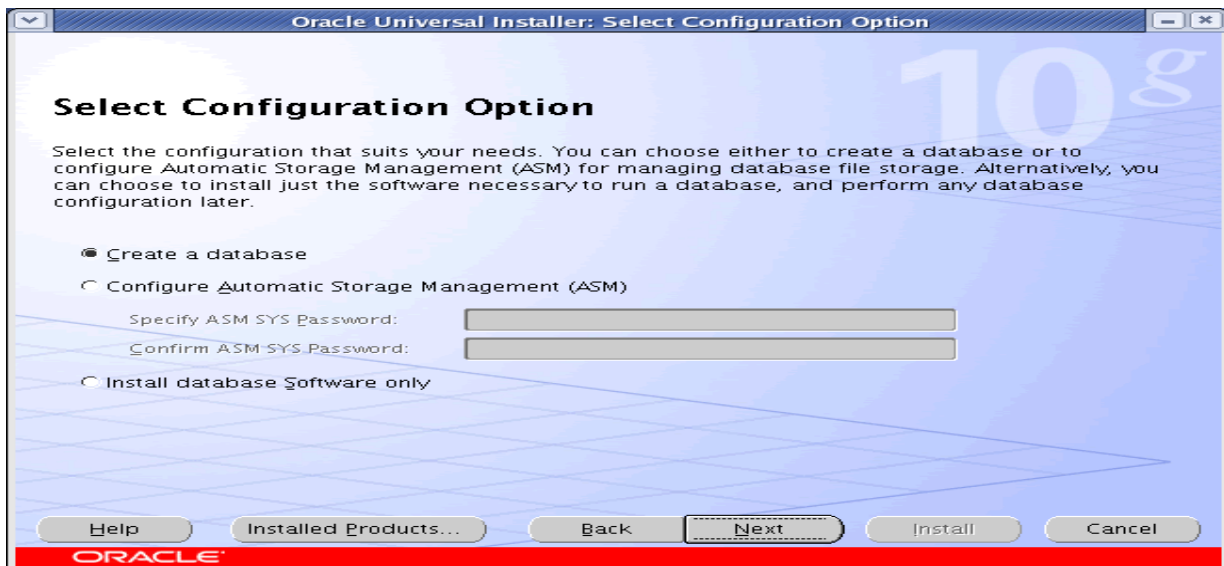
4) Especificar Detalles del Directorio Raíz



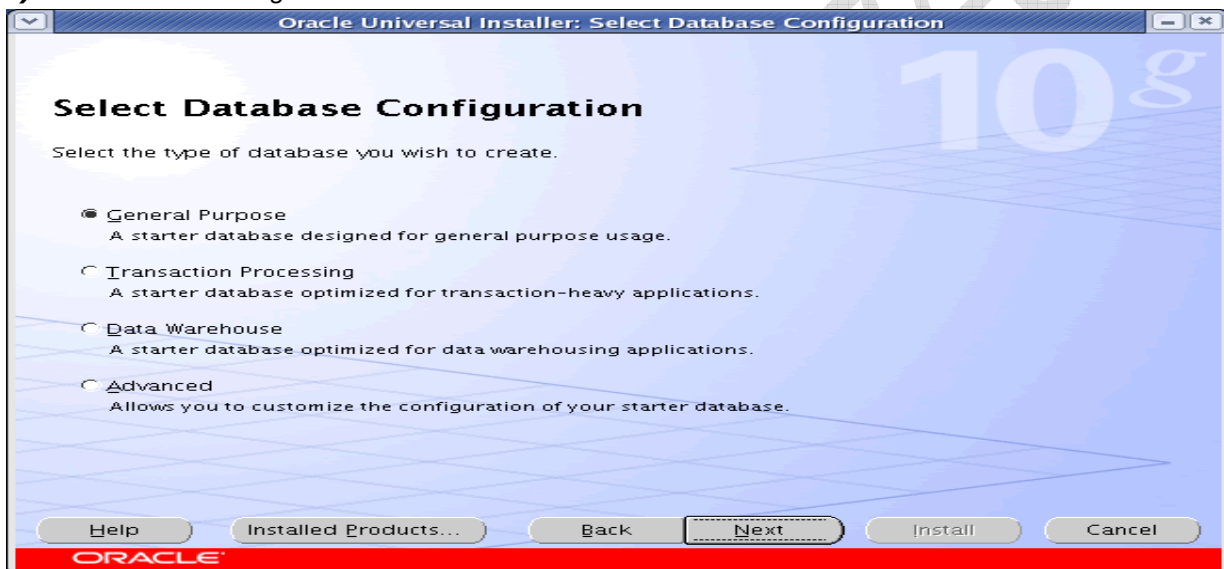
5) Comprobaciones de Requisitos Específicos del Producto



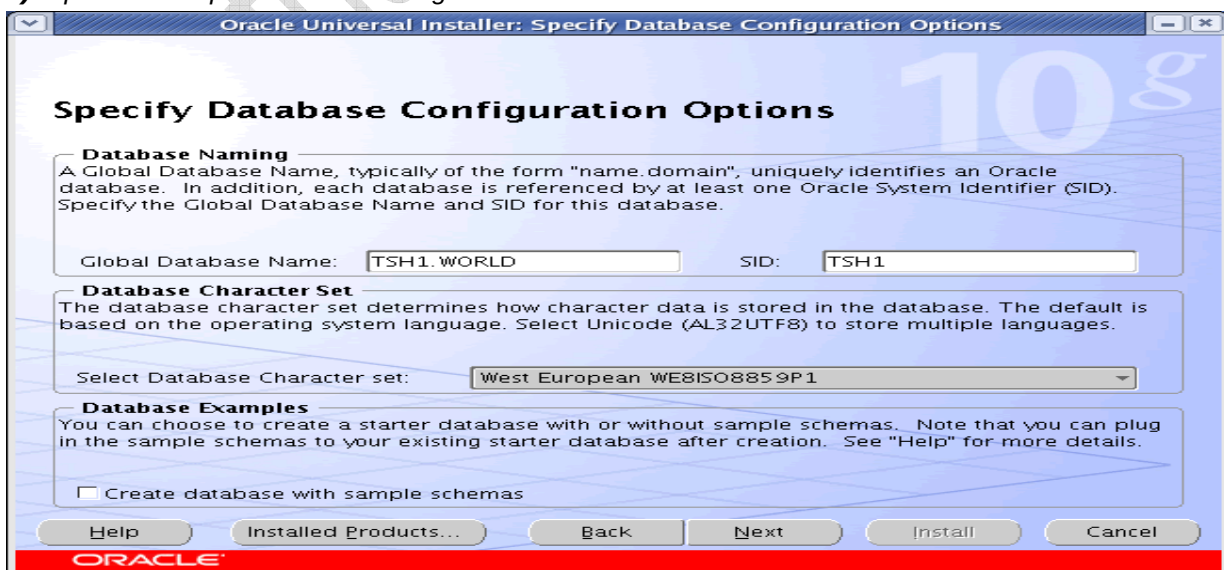
6) Seleccionar Opción de Configuración



7) Seleccionar Configuración de Base de Datos



8) Especificar Opciones de Configuración de Base de Datos



9) Seleccionar Opción de Gestor de Base de Datos

Select Database Management Option

Each Oracle Database 10g may be managed centrally using the Oracle Enterprise Manager 10g Grid Control or locally using the Oracle Enterprise Manager 10g Database Control. For Grid Control, specify the Oracle Management Service through which you will centrally manage your database. For Database Control, you may additionally indicate whether you want to receive email notifications for alerts.

Select the management options for your instance.

☐ Use Grid Control for Database Management

Management Service:

☒ Use Database Control for Database Management

☐ Enable Email Notifications

Outgoing Mail (SMTP) Server:

Email Address:

Help Installed Products... Back Next Install Cancel

ORACLE

10) Especificar Opción de Almacenamiento de Base de Datos

Specify Database Storage Option

Select the storage mechanism you would like to use for database creation.

☒ File System

Use the file system for database storage. For best database organization and performance, Oracle recommends installing database files and Oracle software on separate disks.

Specify Database file location: Browse...

☐ Automatic Storage Management (ASM)

Automatic Storage Management simplifies database storage administration and optimizes database layout for I/O performance.

☐ Raw Devices

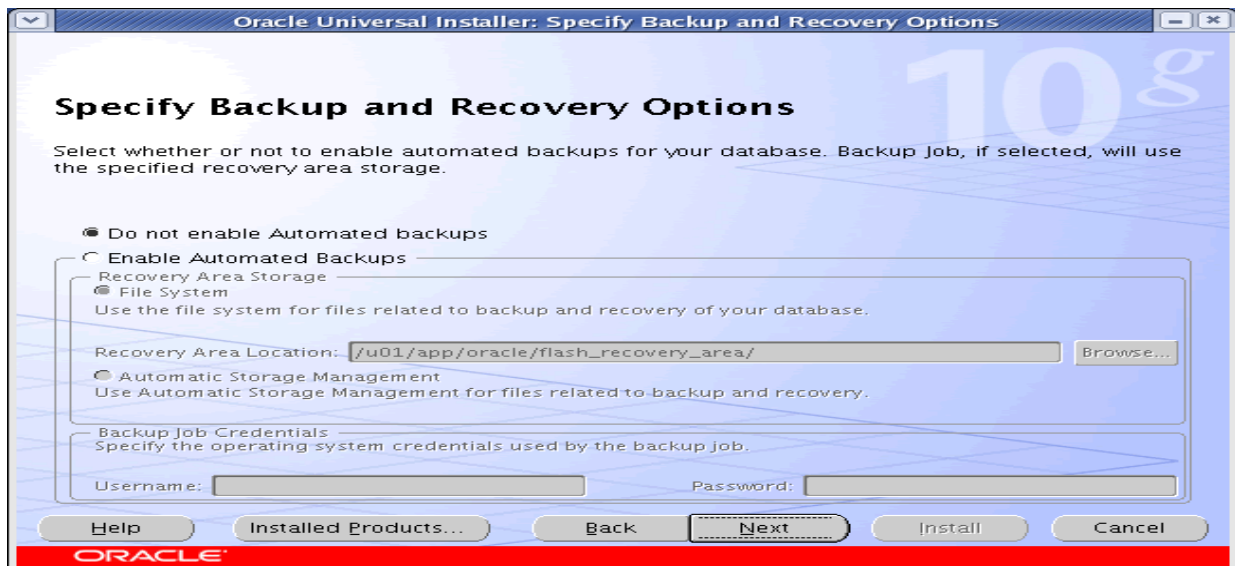
Raw partitions can also provide the required shared storage for Real Application Clusters (RAC) databases. You will need to create one raw device for each data file, control file, and log file for the starter database and then provide a file that maps specific tablespaces, control files, and log files to rawvolumes.

Specify Raw Devices mapping file: Browse...

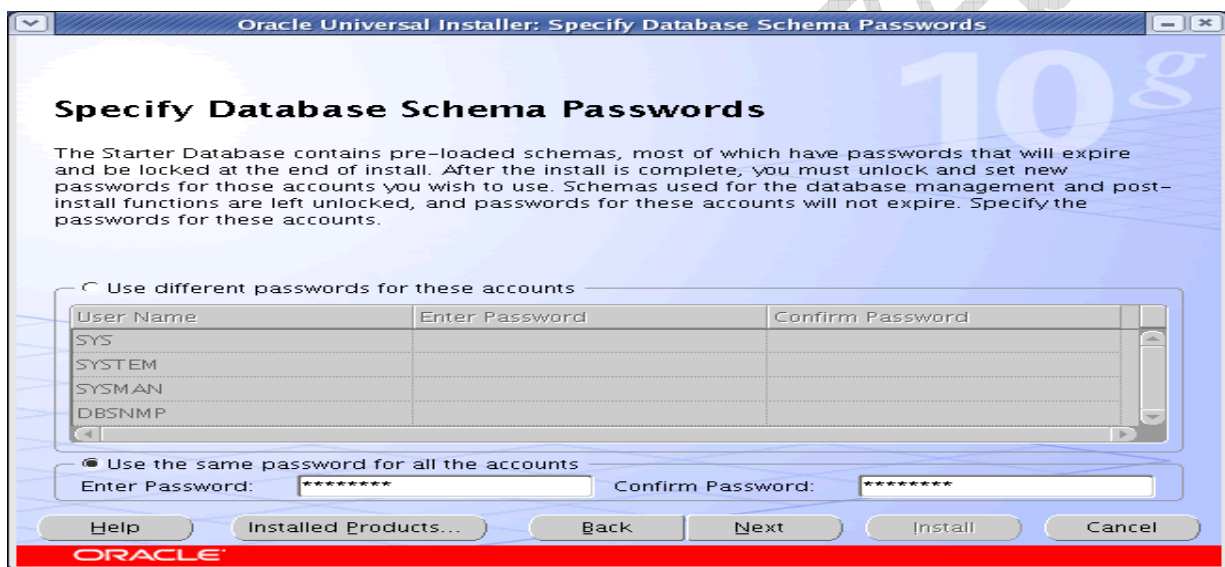
Help Installed Products... Back Next Install Cancel

ORACLE

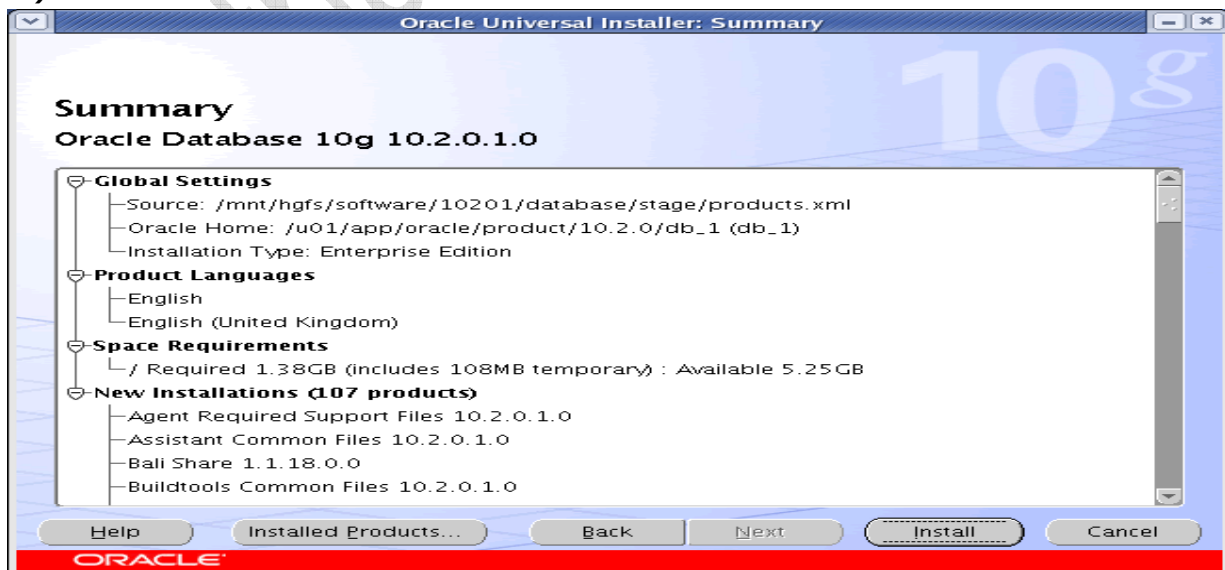
11) Especificar Opciones de Copia de Seguridad y Recuperación



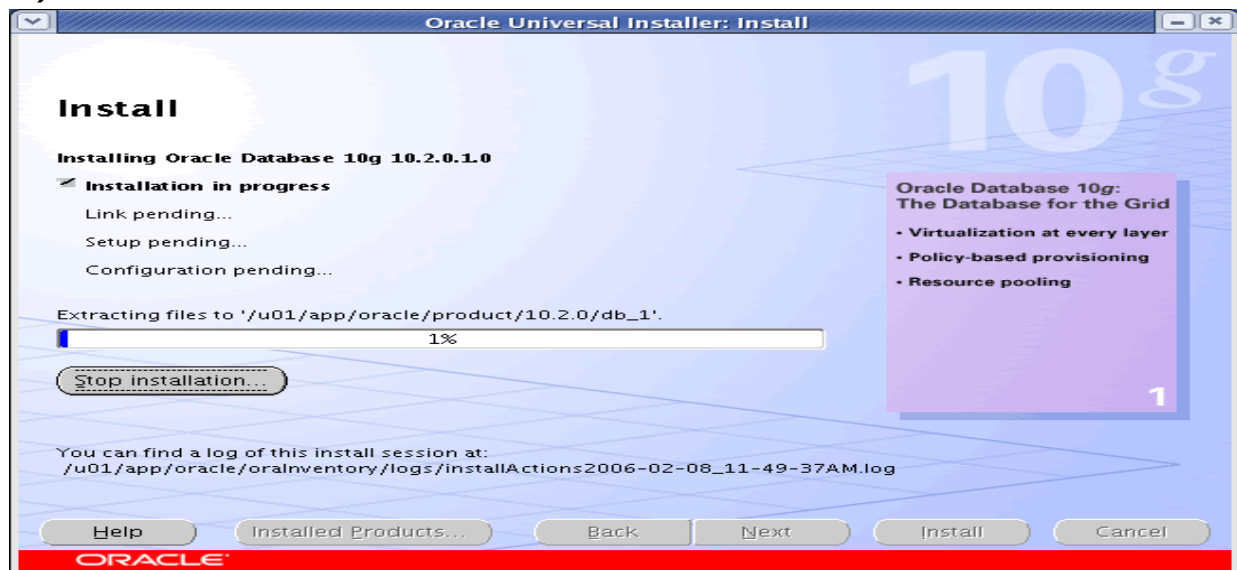
12) Especificar Contraseñas de Esquema de Base de Datos



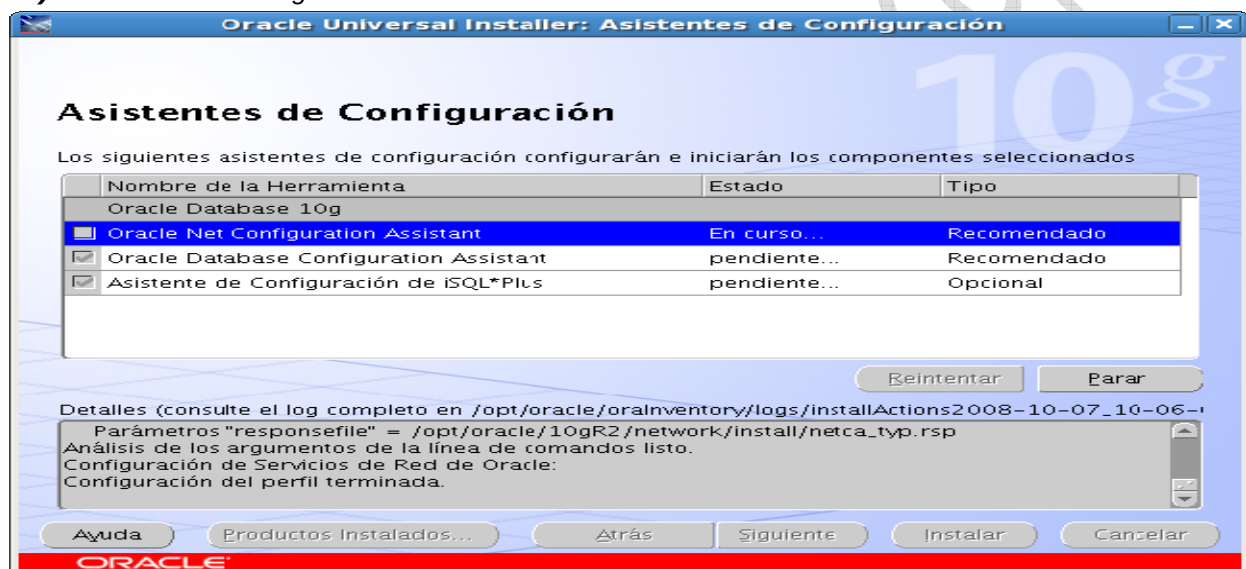
13) Resumen



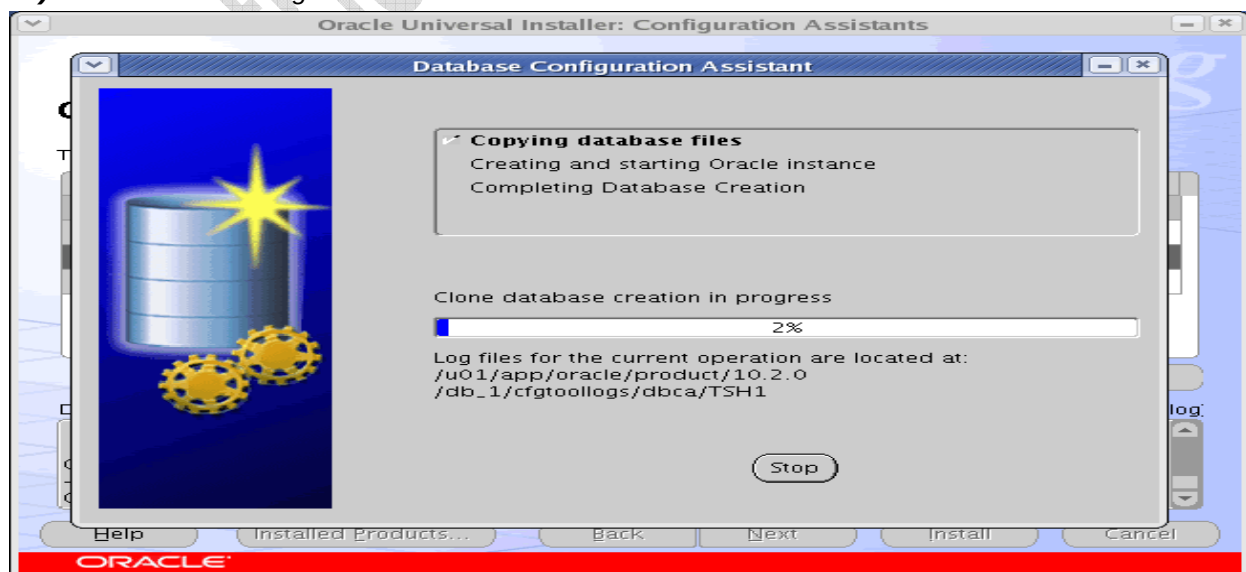
14) Instalación



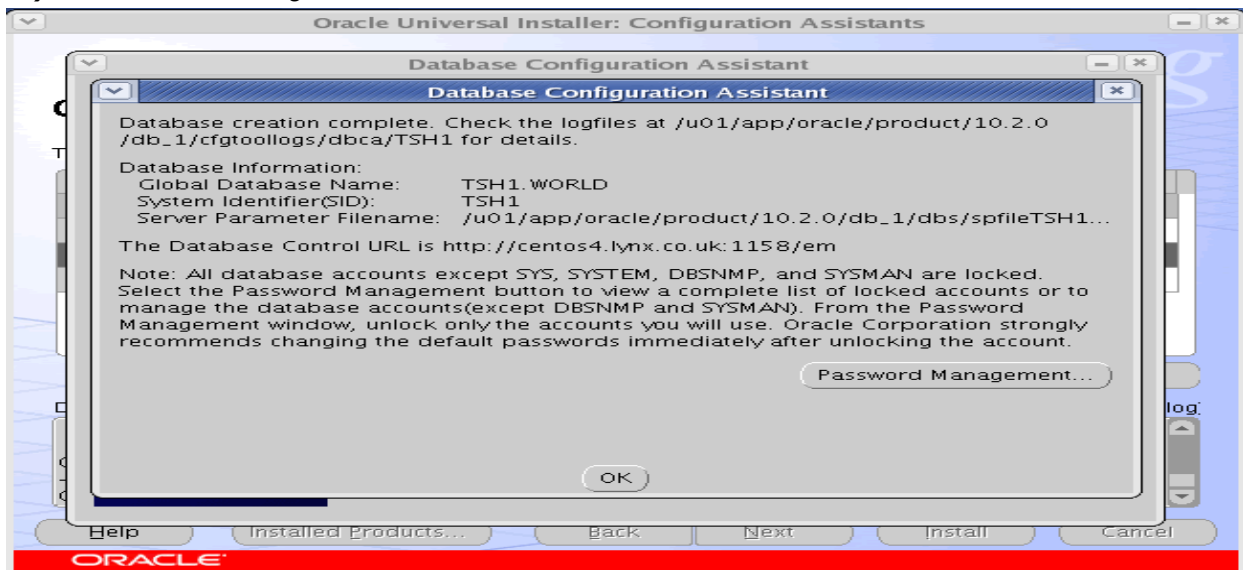
15) Asistente de Configuración



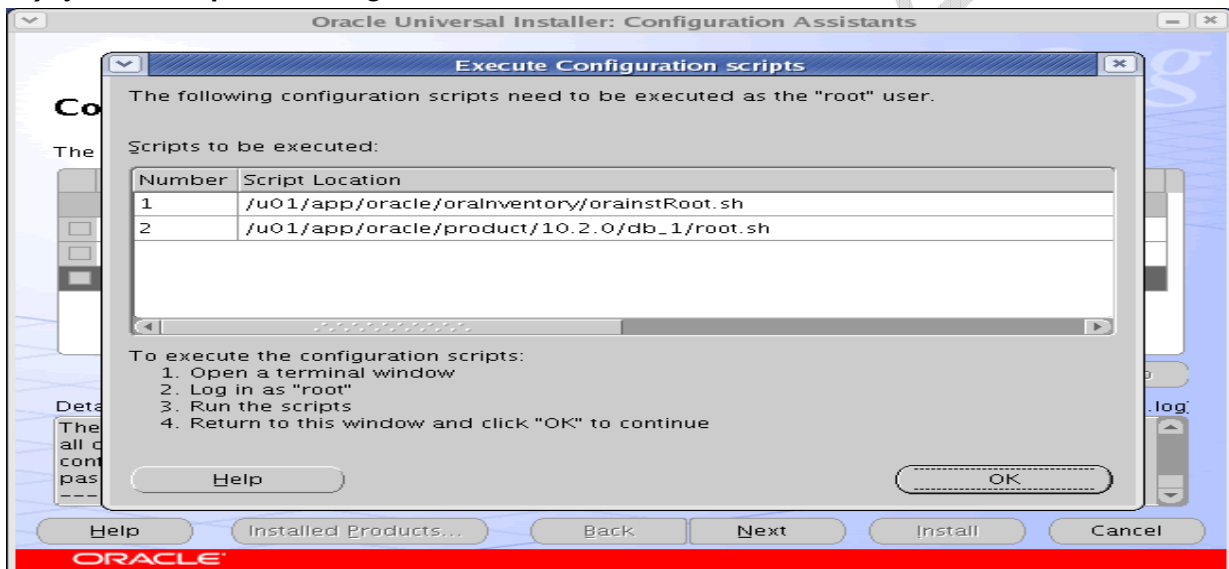
16) Asistente de Configuración de Base de Datos



17) Asistente de Configuración de Base de Datos de Gestión de Contraseñas



18) Ejecutar Scripts de Configuración



La interfaz pide que se ejecute 2 shells desde terminal conectado como root. primero

```
# /opt/oracle/orainventory/orainstRoot.sh
```

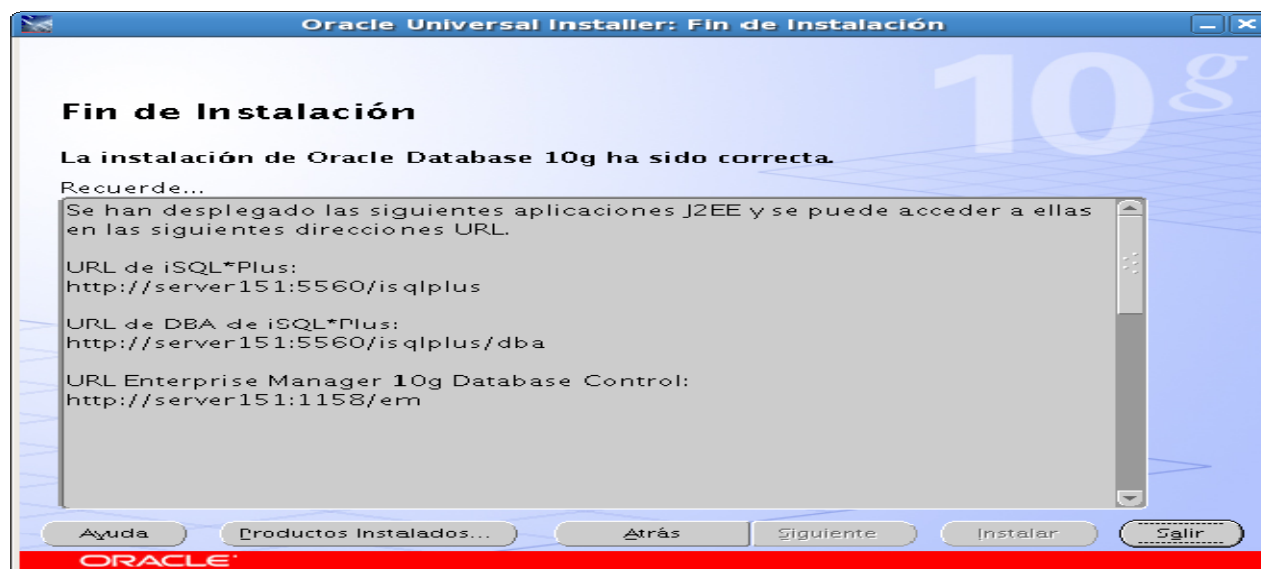
Y después

```
# /opt/oracle/10gR2/root.sh
Running Oracle9 root.sh script...
\nThe following environment variables are set as:
ORACLE_OWNER= oracle
ORACLE_HOME= /opt/oracle/oracle10g
```

Enter the full pathname of the local bin directory: [/usr/local/bin]:

Presionamos enter para tomar el path sugerido entre corchetes (/usr/local/bin/), y proseguimos con la instalación.

19) Fin de la Instalación



Tareas Post-Instalación

Habilitar SELINUX

```
# vi /etc/selinux/config  
  
###editamos  
SELINUX=permissive
```

Editamos el archivo redhat-release para regresar el nombre de nuestra Distribución

```
# vi /etc/redhat-release  
  
### Remplazamos la línea  
redhat-4  
### por  
Fedora Core release 6 (Zod)
```

Instalar rlwrap (Opcional)

Paso Opcional: Podemos usar el rlwrap para trabajar cómodamente con sqlplus. Este paquete RPM para Fedora Core descargar en la página del software. Para instalarlo ejecute el comando rpm

```
# rpm -ivh rlwrap-0.24.fedora.i386.rpm
```

Y agregue al `.bash_profile` del usuario `oracle` la siguiente línea.

```
# cd /opt/oracle/  
# vi .bash_profile  
  
###Agregamos  
alias sqlplus='rlwrap sqlplus'  
  
# ..bash_profile
```

10. Iniciando con Oracle 10g

Ya terminada la instalación deberás verificar que el servicio esté levantado ejecutando

```
$ netstat -na | more
```

Deberá estar el puerto 1521 en modo LISTEN. En caso de que no esté levantado ejecutar

```
$ lsnrctl start
```

Ahora ingresa a `sqlplus` para iniciar

```
$ sqlplus /nolog
```

Conéctate como `sysdba` e inicializa la instancia

```
SQL> connect / as sysdba  
SQL> startup
```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Monitoreo de puertos y procesos levantados por el manejador de Base de Datos.*
- 3. Verificar la ubicación y actualización de archivos de acuerdo a estándares.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 03

Arquitectura de una Base de Datos

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a los componentes de la Arquitectura de una base de datos Oracle, y tenga la capacidad de monitorear cada uno de ellos y el impacto que puedan tener en el comportamiento del rendimiento del manejador
- Identificar claramente cada componente de la Arquitectura de la base de datos.
- Guiar paso a paso en cómo monitorear cada componente de la arquitectura y observar los posibles valores obtenidos.
- Entender el impacto que cada uno de estos componentes tiene en el rendimiento de la base de datos.
- Permitir con el reporte del estudiante el alumno afirme aún más dicho conocimiento.

Tema: Introducción

SubTema(s): Arquitectura del Manejador de Base de Datos/Instalación de la Base de Datos/Creación de una Base de Datos/Manejo y administración de una instancia para Oracle

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Preferentemente trabajar bajo Sistema Operativo Unix (Cualquier Arquitectura)
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Configuración de la red estática
- Monitoreo del Sistema Operativo
- Creación y edición del archivo de parámetros de inicialización
- Creación de una instancia y bases de datos.
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el

manejador de Base de Datos Oracle.

- Siempre se manejarán comandos Unix o estatutos SQL

Martha Escamilla Zepeda

Desarrollo de la práctica

1. Comprobar las variables de entorno necesarias para conectarnos a la BD.

En primer lugar, hay que dar de alta la base de datos. Para ello nos conectamos como el usuario oracle. Accesando SQLPLUS y nos conectamos como SYSDBA:

```
# su - oracle

$ sqlplus
SQL*Plus: Release 9.2.0.1.0 Production on Wed Oct 20 12:39:05 2004
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
Enter username: / as sysdba
Connected to an idle instance.
SQL>
```

Levantamos la BD con el comando STARTUP:

```
SQL> startup
ORACLE instance started.
Total System Global Area 34148352 bytes
Fixed Size 450560 bytes
Variable Size 29360128 bytes
Database Buffers 4194304 bytes
Redo Buffers 143360 bytes
Database mounted.
Database opened.
```

Ahora las variables de ambiente necesarias para conectarnos a la BD son \$ORACLE_BASE, \$ORACLE_HOME, \$ORACLE_SID, \$LD_LIBRARY_PATH y \$PATH. Para conocer su valores usamos el comando Unix echo desde una terminal

```
$ echo $ORACLE_BASE
/opt/oracle
$ echo $ORACLE_HOME
/opt/oracle/oracle10g
echo $ORACLE_SID
prod
```

2. Identificar los procesos que componen instancia.

Realizamos una consulta sobre la vista dinámica V\$PROCESS (o directamente en V\$BGPROCESS), en la BD en cuestión

```
SQL> select username, program from v$process where background is not null;
USERNAME  PROGRAM
SYSTEM    oracle@server151  ###(PMON)
SYSTEM    oracle@server151  ###(DBW0)
SYSTEM    oracle@server151  ###(LGWR)
SYSTEM    oracle@server151  ###(CKPT)
SYSTEM    oracle@server151  ###(SMON)
SYSTEM    oracle@server151  ###(RECO)
6 rows selected.

SQL> select name, description from v$process a, v$bgprocess b where
a.ADDR=b.PADDR;
NAME DESCRIPTION
PMON process cleanup
DBW0 db writer process 0
LGWR Redo etc.
CKPT checkpoint
SMON System Monitor Process
RECO distributed recovery
6 rows selected.
```

3. Ver el tamaño de la SGA de la BD.

Hay dos vistas dinámicas de la BD que nos dan información sobre el tamaño y la estructura de la SGA: V\$SGA y V\$SGASTAT.

```
SQL> select * from v$sga;
NAME          VALUE
Fixed Size    450560
Variable Size 29360128
Database Buffers 4194304
Redo Buffers   143360

### v$sgastat
SQL> select * from v$sgastat
2 where name in ('library cache','dictionary cache','sql area',
3 'buffer_cache','log_buffer');
```

POOL	NAME	BYTES
	buffer_cache	25165824
	log_buffer	656384
shared pool	sql area	2343928
shared pool	library cache	1834420
shared pool	dictionary cache	1610880

4. Comprobar valores de parámetros del init relacionados con el tamaño de la SGA.

Los parámetros de inicialización más importantes que afectan al tamaño de la SGA son: *shared_pool_size*, *db_cache_size*, *db_block_size*, *log_buffer*, *large_pool_size* y *java_pool_size*.

```
SQL> show parameter shared_pool_size
```

NAME	TYPE	VALUE

shared_pool_size	big integer	12582912

```
SQL> show parameter db_cache_size
```

NAME	TYPE	VALUE

db_cache_size	big integer	4194304

```
SQL> show parameter db_block_size
```

NAME	TYPE	VALUE

db_block_size	integer	2048

```
SQL> show parameter log_buffer
```

NAME	TYPE	VALUE

log_buffer	integer	1048576

```
SQL> show parameter large_pool_size
```

NAME	TYPE	VALUE

large_pool_size	big integer	0

```
SQL> show parameter java_pool_size
```

NAME	TYPE	VALUE

java_pool_size	big integer	0

5. Comprobar archivos que componen la BD y ubicarlos en la estructura OFA.

Los archivos que componen la BD los podemos consultar desde el propio Sistema Operativo o desde la BD. Desde Unix y teniendo en cuenta la estructura OFA los encontraremos en */opt/oracle/oradata/\$ORACLE_SID*. Por otra parte podemos localizar cada tipo de archivo por la extensión, *.dbf, *.ctl, *.log. Otra opción es consultar las vistas dinámicas *V\$DATAFILE*, *V\$TEMPFILE*, *V\$CONTROLFILE* y *V\$LOGFILE* (Nota: Los resultados de las siguientes Consultas son ejemplo, es posible que no coincidan con las obtenidas durante la práctica)

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

NAME

/OPT/ORACLE/ORADATA/BDUNIX/SYSTEM01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/UNDOTBS01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/CWMLITE01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/DRSYS01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/EXAMPLE01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/INDX01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/ODM01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/TOOLS01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/USERS01.DBF
/OPT/ORACLE/ORADATA/BDUNIX/XDB01.DBF

10 rows selected.

SQL> select name from **v\$tempfile**;

NAME

/OPT/ORACLE/ORADATA/BDUNIX/TEMP01.DBF

SQL> select member from **v\$logfile**;

MEMBER

/OPT/ORACLE/ORADATA/BDUNIX/REDO03.LOG
/OPT/ORACLE/ORADATA/BDUNIX/REDO02.LOG
/OPT/ORACLE/ORADATA/BDUNIX/REDO01.LOG
/OPT/ORACLE/ORADATA/BDUNIX/REDO11.LOG
/OPT/ORACLE/ORADATA/BDUNIX/REDO12.LOG
/OPT/ORACLE/ORADATA/BDUNIX/REDO13.LOG

6 rows selected.

SQL> select name from **v\$controlfile**;

NAME

/OPT/ORACLE/ORADATA/BDUNIX/CONTROL01.CTL
/OPT/ORACLE/ORADATA/BDUNIX/CONTROL02.CTL
/OPT/ORACLE/ORADATA/BDUNIX/CONTROL03.CTL

6. Identificar la estructura lógica de la BD: tablespaces, segmentos, extensiones

En el DD de la BD tenemos vistas que nos permiten comprobar la estructura lógica de la BD: DBA_TABLESPACES, DBA_DATA_FILES, DBA_SEGMENTS y DBA_EXTENTS.

SQL> select tablespace_name from **dba_tablespaces** order by tablespace_name;
TABLESPACE_NAME

CWMLITE
 DRSYS
 EXAMPLE
 INDX
 ODM
 SYSTEM
 TEMP
 TOOLS
 UNDOTBS1
 USERS
 XDB

SQL> select tablespace_name, file_name from **dba_data_files** order by
 tablespace_name, file_name;

TABLESPACE_NAME	FILE_NAME
-----	-----
CWMLITE	/OPT/ORACLE/ORADATA/BDUNIX/CWMLITE01.DBF
DRSYS	/OPT/ORACLE/ORADATA/BDUNIX/DRSYS01.DBF
EXAMPLE	/OPT/ORACLE/ORADATA/BDUNIX/EXAMPLE01.DBF
INDX	/OPT/ORACLE/ORADATA/BDUNIX/INDX01.DBF
ODM	/OPT/ORACLE/ORADATA/BDUNIX/ODM01.DBF
SYSTEM	/OPT/ORACLE/ORADATA/BDUNIX/SYSTEM01.DBF
TOOLS	/OPT/ORACLE/ORADATA/BDUNIX/TOOLS01.DBF
UNDOTBS1	/OPT/ORACLE/ORADATA/BDUNIX/UNDOTBS01.DBF
USERS	/OPT/ORACLE/ORADATA/BDUNIX/USERS01.DBF
XDB	/OPT/ORACLE/ORADATA/BDUNIX/XDB01.DBF

10 rows selected.

SQL> select tablespace_name, segment_type, count(*) segmentos from
 dba_segments

group by tablespace_name, segment_type;
 TABLESPACE_NAME SEGMENT_TYPE SEGMENTOS

TABLESPACE_NAME	SEGMENT_TYPE	SEGMENTOS
-----	-----	-----
ODM	INDEX	41
ODM	TABLE	39
ODM	LOBINDEX	7
ODM	LOBSEGMENT	7
XDB	INDEX	34
XDB	TABLE	27
XDB	LOBINDEX	271
XDB	LOBSEGMENT	271
DRSYS	INDEX	97
DRSYS	LOBSEGMENT	2
TOOLS	INDEX	66
TOOLS	TABLE	30
SYSTEM	CACHE	1
SYSTEM	INDEX	500
SYSTEM	TABLE	436
SYSTEM	CLUSTER	10
SYSTEM	LOBINDEX	63
SYSTEM	ROLLBACK	1

SYSTEM	LOBSEGMENT	63
SYSTEM	NESTED TABLE	1
SYSTEM	INDEX PARTITION	24
SYSTEM	TABLE PARTITION	27
EXAMPLE	INDEX	132
EXAMPLE	TABLE	61
EXAMPLE	LOBINDEX	23
EXAMPLE	LOBSEGMENT	23
EXAMPLE	NESTED TABLE	3
EXAMPLE	INDEX PARTITION	104
EXAMPLE	TABLE PARTITION	28
UNDOTBS1	TYPE2 UNDO	10

34 rows selected.

```
SQL> select tablespace_name, segment_type, count(*)
2 from dba_extents
3 group by tablespace_name, segment_type;
```

TABLESPACE_NAME SEGMENT_TYPE COUNT(*)

ODM	INDEX	41
ODM	TABLE	78
ODM	LOBINDEX	7
ODM	LOBSEGMENT	7
XDB	INDEX	34
XDB	TABLE	30
XDB	LOBINDEX	271
XDB	LOBSEGMENT	271
DRSYS	INDEX	97
DRSYS	TABLE	53
DRSYS	LOBINDEX	2
DRSYS	LOBSEGMENT	2
TOOLS	INDEX	66
TOOLS	TABLE	30
SYSTEM	CACHE	1
SYSTEM	INDEX	782
SYSTEM	TABLE	840
SYSTEM	CLUSTER	64
SYSTEM	LOBINDEX	63
SYSTEM	ROLLBACK	6
SYSTEM	LOBSEGMENT	106
SYSTEM	NESTED TABLE	1
SYSTEM	INDEX PARTITION	24
SYSTEM	TABLE PARTITION	27
CWMLITE	INDEX	92
CWMLITE	TABLE	57
EXAMPLE	INDEX	199
EXAMPLE	TABLE	147
EXAMPLE	LOBINDEX	23
EXAMPLE	LOBSEGMENT	74
EXAMPLE	NESTED TABLE	3

EXAMPLE	INDEX PARTITION	313
EXAMPLE	TABLE PARTITION	433
UNDOTBS1	TYPE2 UNDO	59

34 rows selected.

7. Consultar información sobre la base de datos (v\$database) y la instancia (v\$instance)

Podemos obtener información de la base de datos y de la instancia, de las vistas V\$DATABASE y V\$INSTANCE, respectivamente.

```
SQL> select name, created, log_mode, checkpoint_change#, open_mode from
v$database;
```

NAME	CREATED	LOG_MODE	CHECKPOINT_CHANGE#	OPEN_MODE
CURSOxy	13-AUG-04	NOARCHIVELOG	753958	READ WRITE

```
SQL> select instance_name, host_name, version, startup_time,
status, archiver, logins, database_status from v$instance;
```

INSTANCE_NAME	HOST_NAME	VERSION	STARTUP_TIME	STATUS	ARCHIV	LOGINS	DATAB ASE_STATUS
BDUNIX	server151	9.2.0.6.0	20-OCT-04	OPEN	STOPPED	ALLOWED	ACTIVE

8. Localizar el proceso "servidor" asociado a mi sesión (v\$process y v\$session). ¿Es un servidor dedicado o compartido?

Toda sesión tiene dos vertientes: cliente y servidor. En el cliente tenemos el proceso de usuario que inicia la sesión y en el servidor de BD tendremos el proceso que sirve las peticiones de dicha sesión; que puede ser un servidor dedicado o compartido. En las vistas V\$SESSION y V\$PROCESS tenemos toda la información relativa a sesiones y procesos, respectivamente.

```
SQL> show user
```

USER is "SYSTEM"

```
SQL> select a.SERVER, a.username dbuser,a.OSUSER, a.PROCESS user_process,
2 a.machine, a.terminal, a.program user_program,
3 b.spid server_process, b.program server_program
4 from v$session a, v$process b
5 where a.username=USER and a.PADDR=b.ADDR;
```

SERVER	DBUSER	OSUSER	USER_PROCESS	MACHINE
DEDICATED	SYSTEM	ORACLE	30580	SERVER151

TERMINAL	USER_PROGRAM	SERVER_PROCE

```
SERVER_PROGRAM
```

```
-----  
3088:3104 CASHITA \BLANKITA
```

```
BLANKITA
```

```
sqlplus.exe 3772
```

```
ORACLE.EXE
```

9. ¿Cuanto ocupa la Dictionary cache y la Library cache en tu BD? (v\$sghastat)

En la vista V\$SGSTAT hay información detallada sobre las partes de la SGA

```
SQL> select * from v$sgstat where name like '%cache';
```

POOL	NAME	BYTES
	buffer_cache	4194304
	shared pool library cache	1567084
	shared pool dictionary cache	1610880

10. Ver la actividad de la Library Cache (v\$librarycache)

En la vista V\$LIBRARYCACHE podemos ver los ratios

```
SQL> select namespace, gethitratio from v$librarycache;
```

NAMESPACE	GETHITRATIO
SQL AREA	.806060606
TABLE/PROCEDURE	.713888889
BODY	0
TRIGGER	1
INDEX	.46969697
CLUSTER	.953020134
OBJECT	1
PIPE	1
JAVA SOURCE	1
JAVA RESOURCE	1
JAVA DATA	1

11 rows selected.

11. Ver las sentencias SQL que guarda la SharedPool (v\$sqlarea)

En la vista V\$SQLAREA tenemos el contenido del "área SQL" de la sharedpool

```
SQL> select sql_text from v$sqlarea;  
SQL_TEXT
```

```

commit
ALTER DATABASE OPEN
grant sysdba to system
ALTER DATABASE MOUNT
SELECT USER FROM DUAL
select o.owner#,o.obj#,decode(o.linkname,null, decode(u.name,null,'SYS',u.name),
o.remoteowner), o.name,o.linkname,o.namespace,o.subname from user$ u, obj$ o
whe
re u.user#(+)=o.owner# and o.type#=:1 and not exists (select p_obj# from depende
ncy$ where p_obj# = o.obj#) for update
...

```

12. Crear un archivo de autenticación y activar su uso.

Los pasos a seguir son:

Crear archivo en \$ORACLE_HOME/dbs con la utilidad orapwd (si ya existe y queremos volver a crearlo, le cambiamos el nombre). Añadir el parámetro remote_login_passwordfile al init.ora. En este caso lo haremos con el comando Unix echo:

```

$ orapwd file=$ORACLE_HOME/dbs/orapwUNIXxy password=miclave entries=5
$ chmod g+w $ORACLE_HOME/dbs/orapwCURSOxy
$ echo "remote_login_passwordfile=EXCLUSIVE" >>
$ORACLE_BASE/admin/$ORACLE_SID/pfile/initUNIXxy.ora
$ echo >> $ORACLE_BASE/admin/$ORACLE_SID/pfile/initUNIXxy.ora

```

Parar y arrancar la BD.

```

SQL> CONNECT / AS SYSDBA
SQL> SHUTDOWN IMMEDIATE
SQL> exit
SQL> CONNECT SYS AS SYSDBA
Enter password: miclave
SQL> STARTUP

```

13. Subir el tamaño de la sharedpool de 12 a 14M y comprobar cómo aumenta el espacio libre.

Primero comprobamos el valor del parámetro shared_pool_size, así como el espacio libre en la shared_pool

```

SQL> show parameter shared_pool_size
NAME                TYPE        VALUE
-----
shared_pool_size    big integer 12582912
SQL> select sum(bytes) from v$sqlstat where pool='shared pool' and name like
'%free%';
SUM(BYTES)

```

Asignarle 14M y volver a comprobar el valor del parámetro, así como el espacio libre que tenga ahora la *shared_pool*

```
SQL> alter system set shared_pool_size=14M;
System altered.
SQL> show parameter shared_pool_size
```

NAME	TYPE	VALUE

shared_pool_size	big integer	16777216

```
SQL> select sum(bytes) from v$$sgastat where pool='shared pool' and name like
'%free%';
SUM(BYTES)
12582912
```

Finalmente, volvemos a dejar la *shared_pool* con 12M y comprobar de nuevo los valores anteriores

```
SQL> alter system set shared_pool_size=12M;
System altered.
SQL> show parameter shared_pool_size
```

NAME	TYPE	VALUE
shared_pool_size	big integer	12582912

```
SQL> select sum(bytes) from v$$sgastat where pool='shared pool' and name like
'%free%';
SUM(BYTES)
10544868
```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Definir cada componente de la Arquitectura de la Base de Datos Oracle 10g*
- 3. Definir el impacto que tiene cada componente en el performance.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 04

Administración de Espacio en Disco

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la creación y administración del espacio en disco y tablespaces generados para una instancia de base de datos para el mejor su mejor rendimiento.
- Guiar al estudiante paso a paso en la configuración, creación, modificación y eliminación de los diferentes tipos de tablespaces alojados en la instancia de base de datos.
- Aprender y poner en práctica la manera de monitorear las diferentes características de los tablespaces.
- Aprender a realizar dichas tareas a través de estatutos de SQL
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Administración de Espacio en Disco

SubTema(s): Estructuras de Almacenamiento/Creación y Manejo de Tablespaces y Datafiles/Manejando Schema Objects/Manejando Datos y Concurrencia

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Preferentemente trabajar bajo Sistema Operativo Unix (Cualquier Arquitectura)
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Creación de instancias
- Creación de tablespaces
- Operaciones de tablespaces
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.
- Siempre se manejarán comandos de Unix o estatutos de SQL.

Martha Escamilla Zepeda

Desarrollo de la práctica

1. Consultar los tablespaces que componen la BD. Comprobar los archivos que tiene cada uno. ¿Cuales son 'locales' y cuáles no?

Conectados como SYSDBA realizamos la siguiente consulta

```
SQL> SELECT tablespace_name, extent_management FROM dba_tablespaces;
```

TABLESPACE_NAME	EXTENT_MAN
SYSTEM	LOCAL
UNDOTBS1	LOCAL
TEMP	LOCAL
CWMLITE	LOCAL
DRSYS	LOCAL
EXAMPLE	LOCAL
INDX	LOCAL
ODM	LOCAL
TOOLS	LOCAL
USERS	LOCAL
XDB	LOCAL

11 filas seleccionadas.

Se consulta los archivos correspondientes a cada tablespace

```
SQL> SELECT rpad(TABLESPACE_NAME,10)
2 tablespace, rpad(FILE_NAME,40) ARCHIVOS,
3 to_char(BYTES/1024/1024,
'999.99') MB,
4 to_char(increment_by*8/1024, '99.99') nextmb,
5 to_char(MAXBYTES/1024/1024, '999999.99') maxmb
6 FROM DBA_DATA_FILES;
```

TABLESPACE	ARCHIVOS	MB	NEXTMB	MAXMB
USERS	/opt/oracle/oradata/dbunix/users01.dbf	5.00	1.25	32767.98
SYSAUX	/opt/oracle/oradata/dbunix/sysaux01.dbf	230.00	10.00	32767.98
UNDOTBS1	/opt/oracle/oradata/dbunix/undotbs01.dbf	30.00	5.00	32767.98
SYSTEM	/opt/oracle/oradata/dbunix/system01.dbf	480.00	10.00	32767.98
EXAMPLE	/opt/oracle/oradata/dbunix/example01.dbf	100.00	.63	32767.98

Se consulta los archivos correspondientes a cada tablespace y la fecha cuando se llevo a cabo

```
SQL> SELECT tablespace_name, extent_management FROM dba_tablespaces
       2 WHERE tablespace_name='DATACURSOXY';
```

```
TABLESPACE_NAME      EXTENT_MAN
-----
DATACURSOXY          LOCAL
```

Creamos las tablas: TABLA01, TABLA02 y TABLA03.

```
SQL> CREATE TABLE TABLA01 (C1 VARCHAR2(4000))
       2 tablespace DATACURSOxy storage (initial 256K minextents 1);
```

Table created.

```
SQL> CREATE TABLE TABLA02 (C1 VARCHAR2(4000))
       2 tablespace DATACURSOxy storage (initial 256K minextents 1);
```

Table created.

```
SQL> CREATE TABLE TABLA03 (C1 VARCHAR2(4000))
       2 tablespace DATACURSOxy storage (initial 256K minextents 1);
```

Table created.

un checkpoint.

2. **Crea el tablespace DATACURSOxy, NO manejado localmente, con el archivo \$ORACLE_BASE/oradata/\$ORACLE_SID/datacursoxy01.dbf, con un tamaño de 1M. Créalo de nuevo, manejado localmente. Crea 3 tablas (TABLA01, TABLA02, TABLA03) de 256K sobre dicho tablespace. Borra la tabla TABLA02 y crea una tabla TABLA04 de 352K. ¿Qué ocurre y por qué?** Los tablespaces 'manejados localmente' gestionan mejor los huecos libres, eliminando la fragmentación que, en el caso de tablespaces tradicionales, provoca el borrado de objetos. El espacio libre inicial en los 'tablespaces locales' es algo menor que en los 'tradicionales', ya que se pierde espacio para el bitmap. Si el tablespace SYSTEM ha sido creado como LOCAL, ya no

```
SQL> CREATE TABLESPACE DATACURSOxy
       2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/datacursoxy01.dbf' size 1M
reuse
```

```
       3 extent management dictionary;
create tablespace DATACURSOxy
*
```

ERROR at line 1:

ORA-12913: Cannot create dictionary managed tablespace

Se crea el tablespace manejado localmente.

```
SQL> CREATE TABLESPACE DATACURSOxy
       2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/datacursoxy01.dbf' size 1M
reuse
       3 extent management local uniform size 128K;
```

Tablespace created.

se podrá crear ningún tablespace manejado por DICCIONARIO en la BD. Consultamos
Borramos la tabla TABLA02.

```
SQL> DROP TABLE tabla02;

Table dropped.
```

Consultamos el espacio libre en el tablespace.

```
SQL> SELECT bytes FROM dba_free_space WHERE tablespace_name='DATACURSOXY';

      BYTES
-----
    131072
    131072
    131072
```

Volvemos a crear la tabla TABLA02, y consultamos el espacio libre en el tablespace.

```
SQL> create table TABLA02 (C1 VARCHAR2(4000))
      2 tablespace DATACURSOxy storage (initial 352K minextents 1);

Table created.

SQL> SELECT bytes FROM dba_free_space WHERE tablespace_name='DATACURSOXY';

no rows selected
```

Como se observa, siendo el tablespace local, se aprovechan los huecos que se generan con el borrado de tablas. La cuestión clave, en este caso, es encontrar un tamaño adecuado para "uniform size". Una posibilidad es tener varios tablespaces: pequeño, mediano y grande con uniform size, por ejemplo, de 8K, 256K y 8M:

- DATOS8K: para guardar tablas de menos de 1M de tamaño.
- DATOS256K: para tablas de entre 1M hasta 32M.
- DATOS8M: para aquellas mayores de 32M.

3. Poner el tablespace DATACURSOxy en modo READONLY. Inserta una fila en alguna de sus tablas, ¿qué ocurre?. Borrar la tabla TABLA01, ¿por qué se puede borrar?. Deja el tablespace DATACURSOxy en modo READWRITE. Repite el insert, ahora sobre TABLA02.

```
SQL> ALTER TABLESPACE DATACURSOxy read only;

Tablespace altered.

### Insertamos
SQL> INSERT INTO tabla01 values ('PRIMERA FILA');
INSERT INTO tabla01 values ('PRIMERA FILA')
      *
ERROR at line 1:
ORA-00372: file 6 cannot be modified at this time
ORA-01110: data file 6: '/opt/oracle/oradata/dbunix/datacursoxy01.dbf'
```

Colocamos el tablespace en modo *READONLY*, e insertamos

Borramos la tabla TABLA01

```
SQL> DROP TABLE TABLA01;  
Table dropped.
```

Ponemos el tablespace *DATACURSOxy* en modo *READWRITE*. y Repetimos el insert, ahora sobre

```
SQL> ALTER TABLESPACE DATACURSOxy read write;  
Tablespace altered.  
  
### Insertamos en la tabla TABLA02  
SQL> INSERT INTO tabla02 values ('PRIMERA FILA');  
1 row created.  
  
SQL commit;
```

TABLA02.

En resumen, en un tablespace en modo *read only* no se pueden actualizar datos (*INSERT*, *UPDATE*, *DELETE*), pero sí se pueden borrar objetos (*DROP TABLE*), ya que el borrado de objetos sólo afecta al *DD* (y no al tablespace).

4. Crea las tablas TABLA04 y TABLA05 en el tablespace DATACURSOxy, ¿qué ocurre y por qué?. Activa el autoextend de su archivo, ajustando next 256K y maxsize 2M. Consultamos el espacio libre del tablespace DATACURSOxy. Y creamos la tabla TABLA04.

Cuando el tablespace se llena no será posible crear más objetos, a no ser que esté activado el 'autoextend', de forma que permita la ampliación del tablespace lo suficiente para albergar el nuevo objeto.

Alteramos la BD, Activamos el autoextend del archivo, ajustando next 256K y maxsize 2M. Y volvemos a crear la tabla TABLA05.

```
SQL> ALTER DATABASE DATAFILE  
'$ORACLE_BASE/oradata/$ORACLE_SID/datacursoxy01.dbf'  
2 autoextend on next 256K maxsize 2M;  
  
Database altered.  
  
SQL> create table TABLA05 (C1 VARCHAR2(4000))  
2 tablespace DATACURSOxy storage (initial 256K minextents 1);  
  
Table created.
```

Consultamos el espacio libre del tablespace DATACURSOXY.

```
SQL> SELECT bytes FROM dba_free_space WHERE tablespace_name='DATACURSOXY';

      BYTES
-----
    131072
    131072

### creamos la tabla TABLA04
SQL> create table TABLA04 (C1 VARCHAR2(4000))
  2 tablespace DATACURSOxy storage (initial 256K minextents 1);

Table created.

### creamos la tabla TABLA05
SQL> create table TABLA05 (C1 VARCHAR2(4000))
  2 tablespace DATACURSOxy storage (initial 256K minextents 1);
create table TABLA05 (C1 VARCHA|R2(4000))
*
ERROR at line 1:
ORA-01658: unable to create INITIAL extent for segment in tablespace
DATACURSOXY
```

```
SQL> select bytes from dba_free_space where tablespace_name='DATACURSOXY';

no rows selected

SQL> SELECT rpad(segment_name,20) segmento, segment_type tipo, bytes from
dba_extents
  2 where tablespace_name='DATACURSOXY';

SEGMENTO      TIPO      BYTES
-----
TABLA04      TABLE    131072
TABLA04      TABLE    131072
TABLA02      TABLE    131072
TABLA02      TABLE    131072
TABLA02      TABLE    131072
TABLA03      TABLE    131072
TABLA03      TABLE    131072
TABLA05      TABLE    131072
TABLA05      TABLE    131072

9 rows selected.
```

5. Crear el tablespace INDCURSOxy de 1M con el archivo '\$ORACLE_BASE/oradata/\$ORACLE_SID/indcursoxy01.dbf'. Moverlo al directorio '\$ORACLE_BASE/oradata/\$ORACLE_SID_2'.

Creamos el tablespace INDCURSOxy.

```
SQL> CREATE TABLESPACE INDCURSOxy datafile
  2  '$ORACLE_BASE/oradata/$ORACLE_SID/indcursoxy01.dbf' size 1M reuse
  3  extent management local uniform size 64K;

Tablespace created.
```

Para mover o renombrar un archivo de un tablespace, éste último debe estar OFFLINE.

```
SQL> ALTER TABLESPACE indcursoxy offline;
Tablespace altered.
```

Creamos el directorio destino, y movemos el datafile.

```
SQL> $ORACLE_BASE/oradata/$ORACLE_SID/indcursoxy01.dbf
SQL> !mv $ORACLE_BASE/oradata/$ORACLE_SID/indcursoxy01.dbf
$ORACLE_BASE/oradata/ $ORACLE_SID-2/indcursoxy01.dbff
```

```
SQL> ALTER TABLESPACE indcursoxy
  2  RENAME DATAFILE '$ORACLE_BASE/oradata/$ORACLE_SID/indcursoxy01.dbf' TO
  3  '$ORACLE_BASE/oradata/$ORACLE_SID-2/indcursoxy01.dbf';
```

Tablespace altered.

```
SQL> ALTER TABLESPACE indcursoxy online;
Tablespace altered.
```

Alteramos el tablespace y lo ponemos online

Realizamos una Consulta para verificar

```
SQL> SELECT file_name FROM dba_data_files;

FILE_NAME
-----
/opt/oracle/oradata/dbunix/users01.dbf
/opt/oracle/oradata/dbunix/sysaux01.dbf
/opt/oracle/oradata/dbunix/undotbs01.dbf
/opt/oracle/oradata/dbunix/system01.dbf
/opt/oracle/oradata/dbunix/example01.dbf
/opt/oracle/oradata/dbunix/datacursoxy01.dbf
/opt/oracle/oradata/dbunix-2/indcursoxy01.dbf

7 rows selected.
```

6. Redimensionar el tablespace DATACURSOxy a 1M ¿qué ocurre y por qué? Redimensionar el tablespace INDCURSOxy a 512K, y añadirle otro archivo de 512K.

Redimensionamos el tablespace DATACURSOxy.

```
SQL> ALTER DATABASE DATAFILE
  2 '$ORACLE_BASE/oradata/$ORACLE_SID/datacursoxy01.dbf' resize 1M;
ALTER DATABASE DATAFILE
'$ORACLE_BASE/oradata/$ORACLE_SID/datacursoxy01.dbf' resize 1M
*
ERROR at line 1:
ORA-03297: file contains used data beyond requested RESIZE value
```

Se puede redimensionar un archivo reduciendo su tamaño sólo si el espacio recortado está libre. En el caso de DATACURSOxy no es posible reducirlo a 1M porque tiene espacio ocupado en el espacio recortado. Sin embargo, INDCURSOxy sí que puede ser reducido a 512K porque está vacío.

Añadimos un datafile indcursoxy02.db de 512K al tablespace INDCURSOxy

```
SQL> ALTER TABLESPACE INDCURSOxy ADD DATAFILE
  2 '$ORACLE_BASE/oradata/$ORACLE_SID-2/indcursoxy02.dbf' size 512K;
Tablespace altered.
```

```
SQL> ALTER DATABASE DATAFILE
  2 '$ORACLE_BASE/oradata/$ORACLE_SID-2/indcursoxy01.dbf' resize 512K;

Database altered.
```

Consultamos para verificar

```
SQL> SELECT rpad(file_name,50),bytes
  2 FROM dba_data_files

RPAD(FILE_NAME, 50)                                BYTES    3  WHERE
tablespace_name='INDCURSOXY';

-----
/opt/oracle/oradata/dbunix-2/indcursoxy01.dbf      524288
/opt/oracle/oradata/dbunix-2/indcursoxy02.dbf      524288
```

7. Crear un tablespace de "undo" UNDO_CURSOxy. Convertirlo en el tablespace de "undo" activo. Crear una tabla TABLA07 en el nuevo tablespace, ¿qué ocurre? Consultar la vista V\$UNDOSTAT.

Podemos tener varios tablespaces de "undo", pero sólo uno de ellos estará activo en un momento dado (el indicado en el parámetro undo_tablespace). No se pueden crear objetos sobre un tablespace de "undo" (como puede ser una tabla). Al cambiar de tablespace "undo" activo

(con undo_tablespace), los segmentos de rollback que contiene el nuevo tablespace pasan a estar “online”, mientras que los del tablespace anterior se ponen offline.

Consultamos

```
SQL> SHOW PARAMETER UNDO_TABLESPACE
```

NAME	TYPE	VALUE
undo_tablespace	string	UNDOTBS1

```
SQL> SELECT SEGMENT_NAME, TABLESPACE_NAME, STATUS FROM DBA_ROLLBACK_SEGS;
```

SEGMENT_NAME	TABLESPACE_NAME	STATUS
SYSTEM	SYSTEM	ONLINE
_SYSSMU1\$	UNDOTBS1	ONLINE
_SYSSMU2\$	UNDOTBS1	ONLINE
_SYSSMU3\$	UNDOTBS1	ONLINE
_SYSSMU4\$	UNDOTBS1	ONLINE
_SYSSMU5\$	UNDOTBS1	ONLINE
_SYSSMU6\$	UNDOTBS1	ONLINE
_SYSSMU7\$	UNDOTBS1	ONLINE
_SYSSMU8\$	UNDOTBS1	ONLINE
_SYSSMU9\$	UNDOTBS1	ONLINE
_SYSSMU10\$	UNDOTBS1	ONLINE
_SYSSMU11\$	UNDO_CURSOXY	OFFLINE
_SYSSMU12\$	UNDO_CURSOXY	OFFLINE
_SYSSMU13\$	UNDO_CURSOXY	OFFLINE
_SYSSMU14\$	UNDO_CURSOXY	OFFLINE
_SYSSMU15\$	UNDO_CURSOXY	OFFLINE
_SYSSMU16\$	UNDO_CURSOXY	OFFLINE
_SYSSMU17\$	UNDO_CURSOXY	OFFLINE

18 rows selected.

Creamos segmentos en un tablespace de Undo

```
SQL> CREATE UNDO TABLESPACE undo_cursoxy  
2 DATAFILE '$ORACLE_BASE/oradata/$ORACLE_SID-2/undo_cursoxy01.dbf' SIZE  
1M;
```

Tablespace created.

Consultamos el parámetro `UNDO_TABLESPACE` . Consultamos la Vista `V$UNDOSTAT` para ver los cambios en el tablespace `TABLESPACE`.

```
SELECT to_char(BEGIN_TIME, 'dd/mm/yyyy hh24:mi') BEGIN_TIME, b.name
TABLESPACE,
2 UNDOBLKS, TXNCOUNT
3 FROM V$UNDOSTAT a, v$tablespace b
4 WHERE A.UNDOTSN=B.TS#
5 ORDER BY BEGIN_TIME;
```

BEGIN_TIME	TABLESPACE	UNDOBLKS	TXNCOUNT
03/12/2008 05:27	UNDOTBS1	844	2556
03/12/2008 05:37	UNDOTBS1	93	410
03/12/2008 05:47	UNDOTBS1	3	47
03/12/2008 05:57	UNDOTBS1	13	132
03/12/2008 06:07	UNDOTBS1	3	48
03/12/2008 06:17	UNDOTBS1	8	82
03/12/2008 06:27	UNDOTBS1	17	170
03/12/2008 06:37	UNDOTBS1	16	239
03/12/2008 06:47	UNDOTBS1	2	54
03/12/2008 06:57	UNDOTBS1	74	187
03/12/2008 07:07	UNDOTBS1	11	199
03/12/2008 07:17	UNDOTBS1	14	87
03/12/2008 07:27	UNDOTBS1	2	69
03/12/2008 07:37	UNDOTBS1	4	54
03/12/2008 07:47	UNDOTBS1	12	79
03/12/2008 07:57	UNDOTBS1	56	117
03/12/2008 08:07	UNDOTBS1	6	147
03/12/2008 08:17	UNDOTBS1	2	53
03/12/2008 08:27	UNDOTBS1	4	80
03/12/2008 08:37	UNDOTBS1	7	82
03/12/2008 08:47	UNDOTBS1	2	50
03/12/2008 08:57	UNDOTBS1	37	214
03/12/2008 09:07	UNDOTBS1	8	92
03/12/2008 09:17	UNDOTBS1	3	79
03/12/2008 09:27	UNDOTBS1	7	54
03/12/2008 09:37	UNDOTBS1	2	52

26 rows selected

En la vista `V$UNDOSTAT` se guarda información valiosa cuando se usa "undo automático", pues podemos ver estadísticas (cada 10 minutos) del tablespace de undo usado (`UNDOTSN`), bloques de rollback consumidos (`UNDOBLKS`) y el n° total acumulado de transacciones de dicho periodo (`TXNCOUNT`).

8. Crear un tablespace temporal `TEMP_CURSOxy` de 1M. ¿Qué ocurre al ponerlo en modo `read only`? Haz que sea el tablespace temporal por defecto de la base de datos ¿que pasa al ponerlo `offline`? ¿Y si lo borramos?

Creamos el tablespace temporal `TEMP_CURSOxy`

```
SQL> CREATE TEMPORARY TABLESPACE temp_cursoxy
2 TEMPFILE '$ORACLE_BASE/oradata/$ORACLE_SID-2/temp_cursoxy01.dbf'
3 SIZE 1M EXTENT MANAGEMENT LOCAL uniform size 64K;
```

Tablespace created.

Alteramos el tablespace TEMP_CURSOxy a modo read only.

```
SQL> ALTER TABLESPACE temp_cursoxy read only;  
ORA-03217: opción no válida para ALTER de TEMPORARY TABLESPACE
```

Alteramos el tablespace TEMP_CURSOxy a modo offline.

```
SQL> ALTER TABLESPACE temp_cursoxy offline;  
ORA-03217: opción no válida para ALTER de TEMPORARY TABLESPACE
```

Un tablespace temporal no puede ponerse en modo read only, y tampoco se puede poner offline.

```
SQL> ALTER TABLESPACE DEFAULT TEMPORARY TABLESPACE temp_cursoxy;  
  
Database altered.
```

Hacemos que sea el tablespace temporal por defecto de la BD.

Borramos el tablespace TEMP_CURSOxy.

```
SQL> DROP TABLESPACE temp_cursoxy;  
  
ORA-12906: no se puede borrar el tablespace temporal por defecto
```

El tablespace temporal por defecto de la BD, no puede ser borrado.

Para poder borrar el tablespace TEMP_CURSOxy, hay que asignar otro tablespace como tablespace predefinido para la BD. En este caso regresamos el tablespace TEMP.

```
SQL> ALTER TABLESPACE DEFAULT TEMPORARY TABLESPACE temp;  
  
Database altered.  
  
### Y ahora borramos el tablespace temporal TEMP_CURSOxy.  
SQL> drop tablespace temp_cursoxy;  
  
Tablespace borrado.
```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa.*
- 2. Monitorear la base de datos creada y tablespaces.*
- 3. Monitorear de cada tablespace, su tamaño, tipo y características generales.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 05

Administración de Usuarios

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la seguridad en la base de datos a través de la creación de usuarios, privilegios y roles para los diferentes niveles de acceso a los datos.
- Conocer cuáles son las vistas y tablas donde la información de usuarios es guardada y poder consultar las características almacenadas.
- Guiar al estudiante paso a paso en la creación, modificación y eliminación de usuarios y asignación de privilegios, así como los roles que puedan ser asignados a los usuarios.
- Asignar que a un usuario a un tablespace específico para que ahí mantenga sus datos específicos
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Administración de Usuarios

SubTema(s): Responsabilidad DBA/Creando y manejando cuentas de usuarios/Seguridad de Base de Datos

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Creación de usuarios y roles
- Asignación de Perfiles
- Asignación y eliminación de Privilegios y roles
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.

- Siempre se manejarán comandos de Linux o estatutos de SQL.

Martha Escamilla Zepeda

Desarrollo de la práctica

1. **Identificar las Vistas que contienen información sobre Usuarios, Perfiles, Privilegios Y Roles. Y checar la información que nos proporcionan.** Dentro de las Vista del Diccionario de Datos de nuestra base encontramos las siguientes:

```
SQL> DESC DBA_PROFILES
SQL> DESC DBA_ROLES
SQL> DESC DBA_USERS
SQL> DESC DBA_ROLE_PRIVS
SQL> DESC DBA_TAB_PRIVS
SQL> DESC DBA_SYS_PRIVS
```

2. **Crear un usuario llamado "administrador" autenticado por la base de datos. Indicar como "tablespace" por default USERS y como "tablespace" temporal TEMP; asignar una cuota de 500K en el "tablespace" USERS.**

Para ello debemos estar conectados como el usuario SYSTEM a la BD.

```
SQL> CREATE USER ADMINISTRADOR IDENTIFIED BY ADMIN
2 DEFAULT TABLESPACE USERS
3 TEMPORARY TABLESPACE TEMP
4 QUOTA 500K ON USERS;
```

User created.

Verificamos en la tabla de usuarios el nuevo usuario.

```
SQL> SELECT USERNAME FROM DBA_USERS WHERE USERNAME='ADMINISTRADOR';
```

3. **Abrir una sesión sqlplus y conectarse como usuario "administrador". En caso de error, verificar los privilegios asignados al usuario, y asignarle los privilegios requeridos para poder conectarse.**

Abrimos otra sesión conectados como el usuario Oracle, abrimos sqlplus.

```
$ sqlplus
SQL*Plus: Release 9.2.0.1.0 - Production on Mon Nov 22 12:50:48 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Enter user-name: administrador
Enter password: ADMIN
ERROR:
ORA-01045: user ADMINISTRADOR lacks CREATE SESSION privilege; logon
denied
```

Como podemos observar el acceso se nos fue negado por falta de privilegios. Por lo tanto, verificamos cuales son los privilegios que tiene el usuario "administrador", desde la terminal de system.

Asignamos el privilegio de "CREATE SESSION" al usuario "administrador" y reintentamos la

```
SQL> grant CREATE SESSION to administrador;

Grant succeeded.

SQL> connect administrador
Enter password:
Connected.

SQL> SELECT * FROM dba_role_privs WHERE grantee='ADMINISTRADOR';

no rows selected

SQL> SELECT * FROM dba_tab_privs WHERE grantee='ADMINISTRADOR';

no rows selected

SQL> SELECT * FROM dba_sys_privs WHERE grantee='ADMINISTRADOR';

no rows selected
```

conexión

4. Conectado como usuario "administrador", crear un usuario llamado "prueba00" que tenga como "tablespace" por default USERS y como "tablespace" temporal TEMP; asignar una cuota de 0K en el "tablespace" USERS. En caso de error otorgar el privilegio faltante.

```
### Verificamos el usuario actual con el cual estamos conectados
SQL> show user
USER is "ADMINISTRADOR"
```

Procedemos a crear el usuario "prueba00"

```
SQL> create user prueba00 identified by prueba00
      2 default tablespace users
      3 temporary tablespace temp
      4 quota 0k on users;

create user prueba00 identified by prueba00
*

ERROR at line 1:
ORA-01031: insufficient privileges
```

Conectado como usuario SYSTEM, otorgar el privilegio "CREATE USER" al usuario "administrador" y repetir el paso anterior.

```
SQL> grant create user to administrador;
```

Grant succeeded.

Desde la terminal de administrador:

```
SQL> create user prueba00 identified by prueba00
2 default tablespace users
3 temporary tablespace temp
4 quota 0k on users
```

User created.

5. Investigar que usuarios de la base de datos tienen asignado los privilegios de "CREATE USER" y "CREATE SESSION" de forma directa.

La vista que nos proporciona información sobre este privilegio es la dba_sys_privs, así que

```
SQL> SELECT * FROM dba_sys_privs WHERE privilege = 'CREATE USER';
```

GRANTEE	PRIVILEGE	ADM
DBA	CREATE USER	YES
SYS	CREATE USER	NO
MDSYS	CREATE USER	YES
WKSYS	CREATE USER	NO
CTXSYS	CREATE USER	NO
ADMINISTRADOR	CREATE USER	NO
IMP_FULL_DATABASE	CREATE USER	NO

7 filas seleccionadas.

```
SQL> SELECT * FROM dba_sys_privs WHERE privilege = 'CREATE SESSION';
```

GRANTEE	PRIVILEGE	ADM
DBA	CREATE SESSION	YES
ODM	CREATE SESSION	NO
SYS	CREATE SESSION	NO
XDB	CREATE SESSION	NO
MDSYS	CREATE SESSION	YES
WKSYS	CREATE SESSION	NO
CTXSYS	CREATE SESSION	NO
ORDSYS	CREATE SESSION	NO
CONNECT	CREATE SESSION	NO
ODM_MTR	CREATE SESSION	NO
WKPROXY	CREATE SESSION	NO
ANONYMOUS	CREATE SESSION	NO
ORDPLUGINS	CREATE SESSION	NO
ADMINISTRADOR	CREATE SESSION	NO
RECOVERY_CATALOG_OWNER	CREATE SESSION	NO

15 filas seleccionadas.

realizamos la consulta.

6. Crear un rol llamado "ADMIN", asignarle los privilegios "CREATE SESSION","CREATE USER" y "CREATE ROLE y asignarlo al usuario "administrador".

Creamos el Rol conectados como SYSTEM, ya que el usuario "administrador" no cuenta con los

```
SQL> CREATE ROLE admin;  
Role created.
```

privilegios para crear Roles.

Asignamos los privilegios al Rol "ADMIN"

```
SQL> GRANT create session, create user, create role TO admin;  
Role created.
```

Asignamos el Rol "ADMIN" al usuario "administrador"

```
SQL> GRANT admin TO administrador;
```

7. Consultar los privilegios de Sistema que tiene asignados el usuario "administrador", revocarlos y asignarle el rol "admin".

Consultamos los privilegios actuales del usuario "administrador"

```
SQL> SELECT * FROM dba_sys_privs WHERE grantee ='ADMINISTRADOR'
```

GRANTEE	PRIVILEGE	ADM
ADMINISTRADOR	CREATE USER	NO
ADMINISTRADOR	CREATE SESSION	NO

Revocamos los privilegios

```
SQL> REVOKE create session, create user FROM administrador;  
  
Revoke succeeded.
```

Consultamos los privilegios y no debe de tener ninguno. Y asignamos el Rol

```
SQL> GRANT admin TO administrador;  
  
Grant succeeded.
```

8. Conectado como SYSTEM, crear un usuario llamado "prueba01" autenticado por la BD al que no se le asigne "tablespace" por default ni temporal. Y verificar cuales son los tablespaces que usa el usuario prueba01.

Creamos el usuario prueba01

```
SQL> CREATE USER prueba01 IDENTIFIED BY prueba01;

User created.
```

Consultamos en la vista correspondiente los "tablespaces" que tiene los usuarios SYS, SYSTEM, "administrador", "prueba00" y "prueba01".

```
SQL> SELECT username, default_tablespace, temporary_tablespace
2 FROM dba_users
3 WHERE username in
('SYS', 'SYSTEM', 'ADMINISTRADOR', 'PRUEBA00', 'PRUEBA01');
USUARIO          DEFAULT_TABLESPACE  TEMPORARY_TABLESPACE
-----
PRUEBA01          SYSTEM              TEMP
PRUEBA00          USERS              TEMP
ADMINISTRADOR     USERS              TEMP
SYSTEM            SYSTEM              TEMP
SYS               SYSTEM              TEMP
```

9. Crear dos "tablespace" llamados NOMINA y ACADEMO, que contendrán datos relativos a las aplicaciones de nomina y datos académicos de los empleados de una empresa, según las siguientes características

Parámetros	ACADEMO	NOMINA
Tamaño inicial	1M	1M
Autoextensible	SI	SI
Extensión	200k	100k
Tamaño máximo	1400k	1500k
Initial	16k	16k
Next	16k	16k

Parámetros	ACADEMO	NOMINA
Minextents	1	1
Maxextents	3	3
Localización	\$ORACLE_BASE/oradata/\$ORACLE_SID/	\$ORACLE_BASE/oradata/\$ORACLE_SID/

Creamos los tablespaces

```
SQL> CREATE TABLESPACE academo DATAFILE
'$ORACLE_BASE/oradata/$ORACLE_SID/academo01.dbf' SIZE 1M
2 AUTOEXTEND ON NEXT 200k MAXSIZE 1400K
3 DEAFULT STORAGE (initial 16k NEXT 16k
4 MINEXTENTS 1 MAXEXTENTS 3);
```

Tablespace created.

```
SQL> CREATE TABLESPACE nomina DATAFILE
'$ORACLE_BASE/oradata/$ORACLE_SID/academo01.dbf' size 1M
2 AUTOEXTEND ON NEXT 100K MAXSIZE 1500K
3 DEAFULT STORAGE (initial 16k NEXT 16k
4 MINEXTENTS 1 MAXEXTENTS 3);
```

10. Crear dos "tablespace" temporales, manejados de forma local, llamados TEMP_NOMINA y TEMP_ACADEMO con las siguientes características:

Parámetros	TEMP_ACADEMO	TEMP_NOMINA
Tamaño inicial	500k	600k
Autoextensible	SI	SI
Extensión	50k	50k
Tamaño máximo	600k	700k
Localización	\$ORACLE_BASE/oradata/\$ORACLE_SID/	\$ORACLE_BASE/oradata/\$ORACLE_SID/

Creamos los tablespaces temporales

```
SQL> CREATE TEMPORARY TABLESPACE temp_academo
  2 TEMPFILE '$ORACLE_BASE/oradata/$ORACLE_SID/temp_academo01.dbf'
  3 SIZE 500k AUTOEXTEND ON NEXT 50k MAXSIZE 600k
  4 EXTENT MANAGEMENT LOCAL UNIFORM SIZE 100k;
```

Tablespace created.

```
SQL> CREATE TEMPORARY TABLESPACE temp_nomina
  2 TEMPFILE '$ORACLE_BASE/oradata/$ORACLE_SID/temp_nomina01.dbf'
  3 SIZE 600k AUTOEXTEND ON NEXT 50k MAXSIZE 700k
  4 EXTENT MANAGEMENT LOCAL UNIFORM SIZE 100k;
```

Tablespace created.

11. Crear un usuario llamado "prueba02" autenticado por base de datos, asignando como "tablespace" por default NOMINA y como "tablespace" temporal TEMP_NOMINA (no se le asignará cuota en NOMINA).

```
SQL> CREATE USER prueba02 IDENTIFIED BY prueba02
  2 DEFAULT TABLESPACE nomina
  3 TEMPORARY TABLESPACE temp_nomina;
```

User created.

Conectado como SYSTEM, creamos el usuario con sus especificaciones de la siguiente manera

12. Asignar al usuario "prueba01" los "tablespace" ACADEMO y TEMP_ACADEMO como "tablespace" de trabajo y temporal respectivamente (sin especificar cuota).

Asignamos los tablespaces al usuario "prueba01"

```
SQL> ALTER USER prueba01 TEMPORARY TABLESPACE temp_academo;
User altered.
```

```
SQL> ALTER USER prueba01 DEFAULT TABLESPACE academo;
User altered.
```

13. Crear un rol llamado "CONEXIÓN", con el privilegio "CREATE SESSION", y asignárselo a los usuarios "prueba00", "prueba01" y "prueba02".

Primero creamos el Rol y después le asignamos el privilegio, les asignamos el Rol a los usuarios

```
SQL> CREATE ROLE conexion;
Role created.

SQL> GRANT create session TO conexion;

Grant succeeded.

SQL> GRANT conexion TO prueba00, prueba01, prueba02;

Grant succeeded.
```

Comprobamos en la vista correspondiente cuales son los roles asignados usuarios "prueba00",

```
SQL> SELECT * FROM dba_role_privs WHERE grantee in
('PRUEBA00', 'PRUEBA01', 'PRUEBA02');
```

GRANTEE	GRANTED_RO	ADM	DEF
PRUEBA00	CONEXION	NO	YES
PRUEBA01	CONEXION	NO	YES
PRUEBA02	CONEXION	NO	YES

"prueba01" y "prueba02".

14. Conectarse como usuario "prueba01" y crear la tabla siguiente en el "tablespace" ACADEMO

¿Es posible hacerlo?, ¿Falta algún permiso? Nos conectamos como el usuario "prueba01" y

```
SQL>connect prueba01/prueba01
Connected.

SQL> CREATE TABLE CODIGOS
2 (CODIGO varchar2(3),
3 DESCRIPCION varchar2(20))
4 TABLESPACE ACADEMO
5 STORAGE (INITIAL 64K
6 NEXT 64K
7 MINEXTENTS 5
8* MAXEXTENTS 10)
CREATE TABLE CODIGOS
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

creamos la tabla.

No podemos crear la tabla, dado que el usuario prueba01 no tiene privilegios para ello.

15. Crear un rol llamado "DESARROLLO" y asignarle los permisos "CREATE SEQUENCE", "CREATE SESSION", "CREATE SYNONYM", "CREATE TABLE" y "CREATE VIEW". Asignar el rol "DESARROLLO" a los usuarios "prueba00", "prueba01" y "prueba02". Asignar cuota ilimitada al usuario "prueba01" en el "tablespace" ACADEMO.

Nos conectamos como SYSTEM y creamos el Rol.

```
SQL> connect system/*****
Connected.

SQL> CREATE ROLE desarrollo;
Role created.
```

Asignamos los privilegios al Rol

```
SQL> GRANT create sequence, create session, create synonym, create table,
create view
TO desarrollo;

Grant succeeded.
```

Asignamos el Rol a los usuarios Asignamos cuota ilimitada al usuario "prueba01"

```
SQL> alter user prueba01 quota unlimited on academo;

User altered.
```

```
SQL> grant desarrollo to prueba00, prueba01, prueba02;

Grant succeeded.
```

16. Volver a conectarse como usuario "prueba01" y crear la tabla ACADEMO del punto 14.

```
SQL> conn prueba01/prueba01
Connected.
SQL>

CREATE TABLE CODIGOS
2 (CODIGO varchar2(3),
3 DESCRIPCION varchar2(20))
4 TABLESPACE ACADEMO
5 STORAGE (INITIAL 64K
6 NEXT 64K
7 MINEXTENTS 5
8 MAXEXTENTS 10);

Table created.
```

17. Asignar cuota ilimitada al usuario "prueba02" en el "tablespace" NOMINA. Obtener información sobre roles, privilegios de Sistema, "tablespace" y cuotas para los usuarios "prueba00", "prueba01" y "prueba02".

Asignando cuota ilimitada al usuario "prueba02"

```
SQL> conn system/system
Connected.

SQL> alter user prueba02 quota unlimited on academo;

User altered.
```

Consultamos la Vista dba_role_privs

```
SQL> SELECT * FROM dba_role_privs WHERE grantee in
('PRUEBA00', 'PRUEBA01', 'PRUEBA02');
GRANTEE      GRANTED_ROLE ADM DEF
-----
PRUEBA00     CONEXION      NO  YES
PRUEBA00     DESARROLLO   NO  YES
PRUEBA01     CONEXION      NO  YES
PRUEBA01     DESARROLLO   NO  YES
PRUEBA02     CONEXION      NO  YES
PRUEBA02     DESARROLLO   NO  YES

SQL> SELECT * FROM dba_sys_privs WHERE ('PRUEBA00', 'PRUEBA01', 'PRUEBA02');

no rows selected

SQL> SELECT USERNAME , TABLESPACE_NAME , BYTES FROM dba_ts_quotas WHERE
2 username in ('PRUEBA00', 'PRUEBA01', 'PRUEBA02');
USERNAME      TABLESPACE      BYTES
-----
PRUEBA02      ACADEMO          0
PRUEBA01      ACADEMO          327680
```

18. Asignar cuota cero en el "tablespace" por default para el usuario "prueba01". Verificar si siguen estando sus objetos, y si es posible crear algún otro (probar creando una tabla).

Realizamos el ALTER USER conectados como SYSTEM

```
SQL> alter user prueba01 quota 0k on academo;

User altered.

SQL> SELECT owner, table_name FROM dba_tables WHERE owner='PRUEBA01';

OWNER TABLE_NAME
-----
PRUEBA01 CODIGOS
```

Nos conectamos como prueba01 para crear la tabla

```
SQL> CONECT prueba01/prueba01
Connected.

SQL> CREATE TABLE CODIGOS2(CODIGO varchar2(3),
DESCRIPCION varchar2(20))
TABLESPACE ACADEMO
STORAGE (INITIAL 64K
NEXT 64K
MINEXTENTS 5
MAXEXTENTS 10);
CREATE TABLE CODIGOS2(CODIGO varchar2(3),
*
ERROR at line 1:
ORA-01536: space quota exceeded for tablespace 'ACADEMO'
```

19. Asignar el permiso DROP USER al rol ADMIN. Averiguar que usuarios o roles de base de datos tienen asignado el privilegio DROP USER.

```
SQL> grant drop user to admin;
Grant succeeded.
SQL> SELECT * FROM dba_sys_privs WHERE privilege='DROP USER';
```

GRANTEE	PRIVILEGE	ADM
DBA	DROP USER	YES
SYS	DROP USER	NO
MDSYS	DROP USER	YES
WKSYS	DROP USER	NO
CTXSYS	DROP USER	NO
MP_FULL_DATABASE	DROP USER	NO

20. Conectado como usuario "administrador", crear el usuario "prueba03" autenticado, con cuotas en el "tablespace" ACADEMO (500K) y NOMINA (200K). Su "tablespace" temporal será

```
SQL> connec administrador/admin
Connected.

SQL> CREATE USER prueba03 IDENTIFIED BY prueba03
DEFAULT TABLESPACE academo
TEMPORARY TABLESPACE temp
QUOTA 500k ON academo
QUOTA 200k ON nomina;

User created.
```

TEMP.

21. Averiguar que usuarios de base de datos o que roles tienen asignado el privilegio "CREATE PROFILE". Conectados como SYSTEM realizamos una consulta a la vista dba_sys_privs

```
SQL> SELECT * FROM dba_sys_privs WHERE privilege='CREATE PROFILE';
```

GRANTEE	PRIVILEGE	ADM
DBA	CREATE PROFILE	YES
SYS	CREATE PROFILE	NO
MDSYS	CREATE PROFILE	YES
WKSYS	CREATE PROFILE	NO
CTXSYS	CREATE PROFILE	NO
IMP_FULL_DATABASE	CREATE PROFILE	NO

22. Asignar el permiso "CREATE PROFILE" al rol ADMIN.

```
SQL> grant create profile to admin;
```

Grant succeeded.

23. Averiguar que perfiles están definidos en la base de datos y que límites de recursos fija cada uno de ellos. Y consultar que perfiles tiene asignados cada usuario de la base de datos.

```
SQL> SELECT profile perfil, resource_name recurso, resource_type,
```

```
2 limit limite FROM dba_profiles
```

```
3 order by profile, resource_name;
```

PERFIL	RECURSO	RESOURCE	LIMITE
DEFAULT	COMPOSITE_LIMIT	KERNEL	UNLIMITED
DEFAULT	CONNECT_TIME	KERNEL	UNLIMITED
DEFAULT	CPU_PER_CALL	KERNEL	UNLIMITED
DEFAULT	CPU_PER_SESSION	KERNEL	UNLIMITED
DEFAULT	FAILED_LOGIN_ATTEMPTS	PASSWORD	UNLIMITED
DEFAULT	IDLE_TIME	KERNEL	UNLIMITED
DEFAULT	LOGICAL_READS_PER_CALL	KERNEL	UNLIMITED
DEFAULT	LOGICAL_READS_PER_SESSION	KERNEL	UNLIMITED
DEFAULT	PASSWORD_GRACE_TIME	PASSWORD	UNLIMITED
DEFAULT	PASSWORD_LIFE_TIME	PASSWORD	UNLIMITED
DEFAULT	PASSWORD_LOCK_TIME	PASSWORD	UNLIMITED
DEFAULT	PASSWORD_REUSE_MAX	PASSWORD	UNLIMITED
DEFAULT	PASSWORD_REUSE_TIME	PASSWORD	UNLIMITED
DEFAULT	PASSWORD_VERIFY_FUNCTION	PASSWORD	NULL
DEFAULT	PRIVATE_SGA	KERNEL	UNLIMITED
DEFAULT	SESSIONS_PER_USER	KERNEL	UNLIMITED

Realizamos una consulta a la vista `dba_profiles` para ver los perfiles definidos en la BD

Consultamos los perfiles de los usuarios ADMINISTRADOR, PRUEBA00, PRUEBA01, PRUEBA02, PRUEBA03, SYSTEM, SYS

```
SQL> SELECT username, profile FROM dba_users WHERE username in
('ADMINISTRADOR', 'PRUEBA00', 'PRUEBA01', 'PRUEBA02', 'PRUEBA03', 'SYSTEM',
'SYS') order by username;
USERNAME          PROFILE
-----
ADMINISTRADOR     DEFAULT
PRUEBA00          DEFAULT
PRUEBA01          DEFAULT
PRUEBA02          DEFAULT
PRUEBA03          DEFAULT
SYS               DEFAULT
SYSTEM            DEFAULT
```

24. Crear un perfil llamado “DESARROLLO” con las siguientes especificaciones y asignar el perfil a los usuarios “prueba00”, “prueba01”, “prueba02” y “prueba03”.

- Sessions_per_user 2
- Cpu_per_session unlimited
- Cpu_per_call 6000
- Connect_time 480
- Idle_time 2
- Failed_login_attempts 2
- Password_life_time 120

Creamos el perfil de la siguiente manera

```
SQL> create profile desarrollo
2 limit
3 sessions_per_user 2
4 cpu_per_session unlimited
5 cpu_per_call 6000
6 connect_time 480
7 idle_time 2
8 failed_login_attempts 2
9 password_life_time 120

Profile created
```

Y lo asignamos a los usuarios

```
SQL> alter user prueba00 profile desarrollo;
User altered.

alter user prueba01 profile desarrollo;
User altered.

alter user prueba02 profile desarrollo;
User altered.

alter user prueba03 profile desarrollo;
User altered.
```

25. Intentar conectarse dos veces como usuario “prueba01” fallando la contraseña, y ver lo que sucede?. Comprobar si la cuenta ha sido bloqueada en la vista de base de datos correspondiente.

Fallamos el password del usuario prueba01 para ver lo que sucede

```
SQL> CONN prueba01/abcde
ERROR:
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.

SQL> CONN prueba01/123456
ERROR:
ORA-28000: the account is locked
```

Ahora consultamos la vista dba_users

```
SQL> SELECT username, lock_date FROM dba_users WHERE username like
'PRUEBA%';
USERNAME    LOCK_DAT
-----
PRUEBA00
PRUEBA02
PRUEBA03
PRUEBA01    29/10/08
```

26. Asignar los privilegios ALTER PROFILE y DROP PROFILE al rol ADMIN.

Conectados como SYSDBA o SYSTEM

```
SQL> grant alter profile to admin;
Grant succeeded.

grant drop profile to admin;
Grant succeeded.
```

27. Asignar el privilegio “GRANT ANY ROLE” al rol “ADMIN.”. Podemos verificar con la vista

```
SQL> SELECT * FROM dba_sys_privs WHERE privi
lege='GRANT ANY ROLE';
GRANTEE    PRIVILEGE          ADM
-----
DBA        GRANT ANY ROLE     YES
SYS        GRANT ANY ROLE     NO
ADMIN      GRANT ANY ROLE     NO
MDSYS      GRANT ANY ROLE     YES
WKSYS      GRANT ANY ROLE     NO
CTXSYS     GRANT ANY ROLE     NO
```

Conectado como SYSTEM:

```
SQL> grant grant any role to admin;  
Grant succeeded.
```

28. Conectarse como usuario SYSTEM y otorgar al usuario "prueba02" el permiso para seleccionar datos de la tabla códigos (pertenece al usuario "prueba01"). ¿Qué sucede?, ¿por qué?

Conectado como SYSTEM otorgamos los permisos a prueba02

```
SQL>SQL> grant select on prueba01.codigos to prueba02;  
grant select on prueba01.codigos to prueba02  
*  
ERROR en línea 1:  
ORA-01031: privilegios insuficientes
```

29. Conectarse como usuario "prueba01" y otorgar al usuario "prueba02" el permiso para seleccionar datos de la tabla códigos; hacerlo de forma que "prueba02" también pueda otorgar el permiso a otros usuarios (opción ADMIN).

Conectado como prueba01 otorgamos el permiso

```
SQL> connect prueba01/prueba01  
Connected.  
  
SQL> grant select on prueba01.codigos to prueba02 with grant  
option;  
Grant succeeded.
```

30. Conectarse como usuario "prueba02" y otorgar al usuario "prueba03" el permiso para seleccionar datos de la tabla códigos.

Conectado como prueba02

```
SQL> connect prueba02/prueba02
Connected.

SQL> grant select on prueba01.codigos to prueba03 ;
Grant succeeded.

SQL> connect prueba03/prueba03
Connected.

SQL> select * FROM prueba01.codigos;
no rows selected
```

31. Conectarse como usuario "prueba01" y revocar al usuario "prueba02" el permiso para seleccionar datos de la tabla códigos.

Conectado como prueba01

```
SQL> connect prueba01/prueba01
Connected.

SQL> revoke select on prueba01.codigos FROM prueba02;
Revoke succeeded.

## Crear table códigos##
```

32. Conectarse como usuario "prueba03" e intentar consultar la tabla códigos. ¿Qué ocurre?, ¿por qué?.

Conectado como prueba03

```
SQL> connect prueba03
Connected.
```

Intentamos consultar la tabla códigos, y nos dice que la tabla no existe porque revocamos el

```
SQL> SELECT * FROM prueba01.codigos;
SELECT * FROM prueba01.codigos
*
ERROR at line 1:
ORA-00942: table or view does not exist
```

permiso de consulta

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa.*
- 2. Identificar tablas relacionadas con usuarios, roles y privilegios.*
- 3. Monitorear los valores asignados a cada tabla.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 06

Manejando UNDO Data

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referente a la administración de los espacio de UNDO, que serán fundamentales en la recuperación de datos manteniendo la consistencia de la base.
- Guiar al estudiante paso a paso en la configuración, creación, modificación y eliminación de las áreas de UNDO.
- Conocer cuáles son las vistas y tablas donde la información de las áreas de UNDO quedan almacenadas y poder consultar sus características específicas.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Manejando UNDO Data

SubTema(s): Funcionamiento de Undo Data/Administración Undo Data

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Preferentemente trabajar bajo Sistema Operativo Unix (Cualquier Arquitectura)
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Edición de archivos de inicialización
- Asignación de espacio UNDO
- Borrado de espacio UNDO
- Configuración de la red estática
- Monitoreo del Sistema Operativo
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.

- Siempre se manejarán comandos de Linux o estatutos de SQL.

Martha Escamilla Zepeda

Desarrollo de la práctica

1. Consultar los archivos de inicialización de la base de datos y averiguar cuáles de los parámetros de "undo" están definidos y su valor actual.

Dentro de una sesión conectado como el usuario oracle, consultamos el archivo `init$ORACLE_SID.ora`:

```
$ cd $ORACLE_BASE/admin/$ORACLE_SID/pfile
$ grep undo init$ORACLE_SID.ora
undo_management = auto
undo_tablespace = undo_rbs1
```

2. Averiguar cuántos espacios de "undo" están definidos en la base de datos (nombre, características de almacenamiento etc.)

Realizamos una consulta a la tabla `DBA_TABLESPACE`

```
SQL> SELECT tablespace_name, initial_extent, next_extent, min_extents,
2 max_extents, status, contents FROM dba_tablespaces WHERE contents='UNDO'
TABLESPACE_NAME INITIAL_EXTENT NEXT_EXTENT MIN_EXTENTS MAX_EXTENTS STATUS
CONTENTS
-----
---
UNDO_RBS1          65536              1  2147483645 ONLINE      U
NDO
```

3. ¿En qué Sistema/s de archivos están definidos los anteriores espacios de almacenamiento?, ¿cuál es su tamaño?

Realizamos la siguiente consulta

```
SQL> SELECT tablespace_name, file_name, file_id, bytes
2 FROM dba_data_files WHERE tablespace_name='UNDO_RBS1';

TABLESPACE_NAME FILE_NAME                                FILE_ID    BYTES
-----
UNDO_RBS1        $ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf          3    4194304
```

4. Consultar el valor actualmente activo de los parámetros de "undo" en la base de datos.

Consultamos los parámetros "undo" de la siguiente manera

```
SQL> show parameters undo

NAME                                TYPE        VALUE
-----
undo_management                     string      AUTO
undo_retention                       integer     900
undo_suppress_errors                 boolean     FALSE
undo_tablespace                     string      undo_rbs1
```

5. Intentar asignar el segmento de "rollback" r01 a la transacción actual. ¿Se genera error?, en caso afirmativo ¿cuál? ¿Cuál es el valor de undo_suppress_errors?

Asignamos el segmento de rollback r01 a la transacción actual

```
SQL> set transaction use rollback segment r01;
set transaction use rollback segment r01
*
ERROR at line 1:
ORA-30019: Illegal rollback Segment operation in Automatic Undo mode
```

6. Modificar dinámicamente el valor del parámetro "undo_suppress_errors" y volver a realizar el ejercicio anterior. ¿Cuál es el resultado?

Modificamos la BD a través de comando ALTER SYSTEM

```
SQL> ALTER system set undo_suppress_errors = TRUE;
System altered.

SQL> set transaction use rollback segment r01;
Transaction set.
```

7. Repetir el ejercicio número 4 y observar el cambio de valor de los parámetros.

```
SQL> show parameters undo
NAME                                TYPE          VALUE
-----
undo_management                     string        AUTO
undo_retention                      integer       900
undo_suppress_errors               boolean     TRUE
undo_tablespace                    string        undo_rbs
```

Consultamos los parámetros

8. Crear un segundo espacio de almacenamiento de "undo" con las siguientes características

Nombre: UNDO02

Localización: \$ORACLE_BASE/oradata/\$ORACLE_SID/undo02.dbf

Tamaño: 1M

Autoextensible: 2M, hasta 3M

Creamos el tablespace undo02

```
SQL> create undo tablespace undo02
2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/undo02.dbf'
3 size 1M
4 autoextend on next 2M maxsize 3M;
Tablespace created.
```

Verificamos con la consulta

```
SQL> SELECT tablespace_name, file_name
       2 FROM dba_data_files WHERE tablespace_name like 'UNDO%';
```

TABLESPACE_NAME	FILE_NAME
UNDO_RBS1	\$ORACLE_BASE/oradata/\$ORACLE_SID/rbs01.dbf
UNDO02	\$ORACLE_BASE/oradata/\$ORACLE_SID/undo02.dbf

9. Crear un espacio de almacenamiento de "undo" con las siguientes características:

Nombre: UNDO03

Localización: \$ORACLE_BASE/oradata/\$ORACLE_SID/undo03.dbf

Tamaño :1M

Definir la cláusula "storage" por default: initial 100k, next 100k y maxextents 50.

Intentamos crear el espacio undo

```
SQL> create undo tablespace undo03
       2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/undo03.dbf'
       3 size 1M
       4 default storage (initial 100k next 100k maxextents 50)
default storage (initial 100k next 100k maxextents 50)
*
ERROR at line 4:
ORA-30024: Invalid specification for CREATE UNDO TABLESPACE
```

10. Modificar el espacio de "undo" por default, de forma que sea a partir de este momento UNDO02. Comprobarlo antes y después de hacerlo.

Verificamos cual es el espacio undo asignado actualmente

```
SQL> show parameters undo
```

NAME	TYPE	VALUE
undo_management	string	AUTO
undo_retention	integer	900
undo_suppress_errors	boolean	TRUE
undo_tablespace	string	undo_rbs1

Modificamos el espacio UNDO por default

```
SQL> ALTER system set undo_tablespace=undo02;
System altered.
```

Volvemos a comprobar

```
SQL> show parameters undo
```

NAME	TYPE	VALUE
undo_management	string	AUTO
undo_retention	integer	900
undo_suppress_errors	boolean	TRUE
undo_tablespace	string	UNDO02

11. Modificar el archivo de inicialización comentando la línea correspondiente a la variable "undo_tablespace". ¿Cuál es el espacio de "undo" que se habilita? (recuerde que al cerrar la base de datos era UNDO02, ver ejercicio anterior)

Iniciamos sesión en sqlplus y damos de baja la bd

```
$ sqlplus /nolog
SQL*Plus: Release 9.2.0.1.0 - Production on Thu Nov 18 14:40:33 2004
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
SQL> connect / as sysdba;
Connected.

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

En otra sesión, editamos el archivo de inicialización, comentando la línea undo_tablespace

```
$ VI $ORACLE_BASE/admin/$ORACLE_SID/pfile/init$ORACLE_SID.ora
...
# undo_tablespace = undo_rbs1
...
```

Levantamos la BD en modo normal, nos conectamos como SYSTEM, y consultamos

```
SQL> startup
ORACLE instance started.
Total System Global Area ...
...
Database mounted.
Database opened.

SQL> connect system/*****
Connected.

SQL> show parameters undo
NAME                                TYPE                                VALUE
-----
undo_management                     string                             AUTO
undo_retention                      integer                            900
undo_suppress_errors                boolean                             FALSE
undo_tablespace                     string                             undo_rbs1
```

12. Modificar el archivo de inicialización y asignar a la variable "undo_tablespace" el valor UNDO02. Parar y arrancar la base de datos.

Editamos el archivo de inicialización, asignando "UNDO02" a al parámetro undo_tablespace, y reiniciamos la BD. Conectados como System

```
$ VI $ORACLE_BASE/admin/$ORACLE_SID/pfile/
undo_management = auto
undo_tablespace = UNDO02

$ sqlplus /nolog
SQL*Plus: Release 9.2.0.1.0 - Production on Thu Nov 18 14:49:42 2004
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
SQL> connect /as sysdba;
Connected.

    ### Damos de Baja la BD
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.

    ### Levantamos la BD
SQL> startup
ORACLE instance started.
...
...
Database mounted.
Database opened.
```

13. Borrar el espacio de "undo" UNDO02. Intentamos borrar el tablespace UNDO02

No podemos borrar ese tablespace porque está siendo utilizado como espacio undo, por lo tanto primero tenemos que asignar otro tablespace como nuevo espacio undo, y después podremos borrar el espacio UNDO02

```
### Cambiamos de espacio UNDO a UNDO_RBS1
SQL> ALTER system set undo_tablespace='UNDO_RBS1';
System altered.
```

```
### Verificamos
SQL> show parameters undo
NAME                                TYPE                                VALUE
-----
undo_management                     string                              AUTO
undo_retention                      integer                             900
undo_suppress_errors                boolean                             FALSE
undo_tablespace                    string                             UNDO_RBS1
```

```
### Borramos el espacio UNDO02
SQL> drop tablespace undo02;
Tablespace dropped.
```

```
SQL CONN system/***** |
Connected.
SQL> show parameters undo
```

```
NAME                                TYPE                                VALUE
-----
undo_management                     string                              AUTO
undo_retention                      integer                             900
undo_suppress_errors                boolean                             FALSE
undo_tablespace                     string                              UNDO02
```

14. Parar y arrancar la base de datos en modo normal y ver lo que sucede.

Conectados como SYSDBA damos de baja la BD, y la levantamos en modo normal

```
SQL> connect / as sysdba;
Connected.
```

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

```
SQL> startup
ORACLE instance started.
Total System Global Area 34148352 bytes
Fixed Size 450560 bytes
Variable Size 29360128 bytes
Database Buffers 4194304 bytes
Redo Buffers 143360 bytes
Database mounted.
ORA-01092: ORACLE instance terminated. Disconnection forced
```

Debido a que en el archivo `init$ORACLE_SID.ora` el parámetro `"undo_tablespace"` tiene el valor `"UNDO02"`, nos manda este mensaje.

15. Comentar el parámetro `"undo_tablespace"` en el archivo de inicialización, y arrancar la base de datos. ¿Qué espacio de `"undo"` tiene asignado?

Desde una terminal editamos el archivo de inicialización

```
$ VI init$ORACLE_SID.ora
### Comentamos el parámetro
...
undo_management = auto
#undo_tablespace = UNDO02
...
```

Arrancamos la BD y consultamos cual es el espacio undo utilizado por la BD

```
SQL> connect / as sysdba;
Connected to an idle instance.

SQL> startup
ORACLE instance started.
Total System Global Area 34148352 bytes
Fixed Size 450560 bytes
Variable Size 29360128 bytes
Database Buffers 4194304 bytes
Redo Buffers 143360 bytes
Database mounted.
Database opened.

SQL> show parameters undo
NAME                                TYPE                                VALUE
-----
undo_management                     string                             AUTO
undo_retention                      integer                            900
undo_suppress_errors                boolean                             FALSE
undo_tablespace                     string                             UNDO_RBS1
```

16. Comentar todos los parámetros correspondientes a `"undo"` en el archivo de inicialización, parar y arrancar la B.D. Comprobar los valores de `"undo"`, ¿qué observa?

Comentamos los parámetros undo

```
$ VI init$ORACLE_SID.ora
### Comentamos el parámetro
...
#undo_management = auto
#undo_tablespace = UNDO02
...
```

Reiniciamos la BD y consultamos los valores tomados por los parámetros undo

```
SQL> connect /as sysdba;
Connected.

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

SQL> startup
ORACLE instance started.
...
...
Database opened.

###Consultamos
SQL> show parameters undo
NAME                                TYPE                                VALUE
-----                                -
undo_management                     string                             MANUAL
undo_retention                       integer                             900
undo_suppress_errors                 boolean                             FALSE
undo_tablespace                      string
```

17. Borrar el espacio de "undo" UNDO_RBS1. Habilitar de nuevo la gestión automática de "undo" y, parar y arrancar la base de datos. Comprobar los valores de "undo", ¿qué espacio de "undo" se usa?

Comprobamos lo valores iniciales

```
SQL> SELECT * FROM dba_data_files WHERE tablespace_name='UNDO_RBS1'

FILE_NAME                                FILE_ID TABLESPACE_NAME  BYTES  BLOCKS STAT
US
-----
--
$ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf 3 UNDO_RBS1      4194304    2048 AVAILA
BLE

RELATIVE_FNO AUT MAXBYTES MAXBLOCKS INCREMENT_BY USER_BYTES USER_BLOCKS
...
3 NO          0          0          0      4128768      2016
```

Borramos el espacio UNDO_RBS1

```
SQL> drop tablespace UNDO_RBS1;
Tablespace dropped.
```

Damos de baja la BD y habilitamos la gestión automática de undo y Reiniciamos la BD y verificamos los parámetros undo

```
SQL> connect /as sysdba;
Connected.

SQL> shutdown immediate
Database closed.

Database dismounted.
ORACLE instance shut down.

$ VI init$ORACLE_SID.ora
### Descomentamos el parámetro
...
undo_management = auto
#undo_tablespace = UNDO002
...
```

```
SQL> startup
ORACLE instance started.
...
...
Database opened.

SQL> show parameters undo
NAME                                TYPE        VALUE
-----
undo_management                     string      AUTO
undo_retention                      integer     900
undo_suppress_errors                boolean     FALSE
undo_tablespace                     string
```

18. Crear un tablespace llamado PRUEBA, que no sea de "undo", e intentar asignarlo como espacio de "undo".

Creamos el tablespace con las siguientes características:

Nombre: PRUEBA

Localización: \$ORACLE_BASE/oradata/\$ORACLE_SID/prueba01.dbf

Tamaño: 1M

Creamos el tablespace y después lo intentamos asignar como espacio undo

```
SQL> create tablespace prueba datafile
  2 '$ORACLE_BASE/oradata/$ORACLE_SID/prueba01.dbf' size 1M

Tablespace created.

SQL> ALTER system set undo_tablespace=PRUEBA;
ALTER system set undo_tablespace=PRUEBA
*
ERROR at line 1:
ORA-02097: parameter cannot be modified because specified value is invalid
ORA-30012: undo tablespace 'PRUEBA' does not exist or of wrong type
```

Como vemos no podemos asignar un tablespace como espacio undo.

19. Crear de nuevo el espacio de “undo” UNDO_RBS1 y asignarlo como tal.

Creamos el espacio undo con las siguientes características:

Nombre: UNDO_RBS1

Localización: \$ORACLE_BASE/oradata/\$ORACLE_SID/rbs01.dbf

Tamaño: 1M

Siguiente extensión: 500k

Tamaño máximo: 2M

```
SQL> create undo tablespace undo_rbs1
  2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf'
  3 size 1M autoextend on next 500K maxsize 2M;
create undo tablespace undo_rbs1
datafile '$ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf'
*
ERROR at line 2:
ORA-01119: error in creating database file
'$ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf'
ORA-27038: skgfrcre: file exists
```

Nota: Cuando se borró el espacio UNDO_RBS1, no se eliminó el archivo correspondiente y se genera un error. Es necesario previamente borrar o renombrar "rbs01.dbf"; se elige renombrar el archivo por ser una operación más segura en este caso.

Renombramos el archivo rbs01.dbf desde SQLPLUS. Creamos el espacio undo y lo asignamos

```
SQL> ! cd $ORACLE_BASE/oradata/$ORACLE_SID/
mv rbs01.dbf rbs01.dbf.antiguo
exit

SQL> create undo tablespace undo_rbs1
  2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/rbs01.dbf'
  3 size 1M autoextend on next 500K maxsize 2M;
```

Asignamos "rbs01" como espacio undo, y verificamos

```
SQL> ALTER system set undo_tablespace=undo_rbs1
System altered.
```

```
SQL> show parameters undo
```

NAME	TYPE	VALUE
undo_management	string	AUTO
undo_retention	integer	900
undo_suppress_errors	boolean	FALSE
undo_tablespace	string	UNDO_RBS1

20. Obtener la descripción completa de las vistas siguientes.

- DBA_UNDO_EXTENTS
- V\$UNDOSTAT
- V\$ROLLSTAT

Consultamos las Vistas usando el comando desc

dba_undo_extents: Especialmente interesantes son las columnas "owner" (propietario del espacio de "undo"), tablespace_name (nombre del espacio de "undo"), y "status" (estado de la transacción en la extensión: Active, contiene transacciones activas, expired, el segmento no se utiliza, o unexpired, no hay transacciones activas pero contiene datos que pueden ser requeridos por otras transacciones).

```
SQL> desc dba_undo_extents
```

Name	Null?	Type
OWNER		CHAR(3)
SEGMENT_NAME	NOT NULL	VARCHAR2(30)
TABLESPACE_NAME	NOT NULL	VARCHAR2(30)
EXTENT_ID		NUMBER
FILE_ID	NOT NULL	NUMBER
BLOCK_ID		NUMBER
BYTES		NUMBER
BLOCKS		NUMBER
RELATIVE_FNO		NUMBER
COMMIT_JTIME		NUMBER
COMMIT_WTIME		VARCHAR2(20)
STATUS		VARCHAR2(9)

v\$undostat: Muestra datos estadísticos relativos a consumo de espacio de "undo", longitud de las consultas ejecutadas, Se devuelven valores nulos si se trabaja en modo manual. Cada fila de la vista corresponde a un intervalo de 10 minutos (en total 1008 filas, correspondientes a un periodo de siete días).

```
SQL> desc v$undostat
```

Name	Null?	Type
BEGIN_TIME		DATE
END_TIME		DATE
UNDOTSN		NUMBER
UNDOBLKS		NUMBER
TXNCOUNT		NUMBER
MAXQUERYLEN		NUMBER
MAXCONCURRENCY		NUMBER
UNXPSTEALCNT		NUMBER
UNXPBLKRELCNT		NUMBER
UNXPBLKREUCNT		NUMBER
EXPSTEALCNT		NUMBER
EXPBLKRELCNT		NUMBER
EXPBLKREUCNT		NUMBER
SSOLDERRCNT		NUMBER
NOSPACEERRCNT		NUMBER

v\$rollstat: En modo automático refleja el comportamiento de los segmentos de "undo" en el

```
SQL> desc v$rollstat
```

Name	Null?	Type
USN		NUMBER
LATCH		NUMBER
EXTENTS		NUMBER
RSSIZE		NUMBER
WRITES		NUMBER
XACTS		NUMBER
GETS		NUMBER
WAITS		NUMBER
OPTSIZE		NUMBER
HWMSIZE		NUMBER
SHRINKS		NUMBER
WRAPS		NUMBER
EXTENDS		NUMBER
AVESHRINK		NUMBER
AVEACTIVE		NUMBER
STATUS		VARCHAR2(15)
CUREXT		NUMBER
CURBLK		NUMBER

espacio de "undo".

21. Habilitar el modo de gestión manual de "undo". Parar y arrancar la base de datos.

Editamos el archivo `init$ORACLE_SID.ora` y reiniciamos la BD

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

SQL> startup
ORACLE instance started.
...
...
Database mounted.
Database opened.
```

Consultamos los parámetros undo

```
SQL> show parameters undo
NAME                                TYPE                                VALUE
-----                                -
undo_management                     string                             MANUAL
undo_retention                      integer                           900
undo_suppress_errors                boolean                           FALSE
undo_tablespace                     string                             UNDO_RBS1
```

22. Obtener la descripción completa de las vistas siguiente

- DBA_SEGMENTS
- DBA_ROLLBACK_SEGS
- V\$ROLLNAME
- V\$ROLLSTAT

dba_segments ([más información](#))

```
SQL> desc dba_segments
Name                                Null?                                Type
-----                                -
OWNER                              VARCHAR2(30)
SEGMENT_NAME                        VARCHAR2(81)
PARTITION_NAME                      VARCHAR2(30)
SEGMENT_TYPE                        VARCHAR2(18)
TABLESPACE_NAME                     VARCHAR2(30)
HEADER_FILE                         NUMBER
HEADER_BLOCK                       NUMBER
BYTES                              NUMBER
BLOCKS                             NUMBER
EXTENTS                            NUMBER
INITIAL_EXTENT                     NUMBER
NEXT_EXTENT                        NUMBER
MIN_EXTENTS                        NUMBER
MAX_EXTENTS                        NUMBER
PCT_INCREASE                       NUMBER
FREELISTS                          NUMBER
FREELIST_GROUPS                    NUMBER
RELATIVE_FNO                       NUMBER
BUFFER_POOL                         VARCHAR2(7)
```

dba_rollback_segs

```
SQL> desc dba_rollback_segs
```

Name	Null?	Type
SEGMENT_NAME	NOT NULL	VARCHAR2(30)
OWNER		VARCHAR2(6)
TABLESPACE_NAME	NOT NULL	VARCHAR2(30)
SEGMENT_ID	NOT NULL	NUMBER
FILE_ID	NOT NULL	NUMBER
BLOCK_ID	NOT NULL	NUMBER
INITIAL_EXTENT		NUMBER
NEXT_EXTENT		NUMBER
MIN_EXTENTS	NOT NULL	NUMBER
MAX_EXTENTS	NOT NULL	NUMBER
PCT_INCREASE		NUMBER
STATUS		VARCHAR2(16)
INSTANCE_NUM		VARCHAR2(40)
RELATIVE_FNO		NOT NULL NUMBER

V\$rollname / v\$rollstat

```
SQL> desc v$rollstat
```

Name	Null?	Type
USN		NUMBER
LATCH		NUMBER
EXTENTS		NUMBER
RSSIZE		NUMBER
WRITES		NUMBER
XACTS		NUMBER
GETS		NUMBER
WAITS		NUMBER
OPTSIZE		NUMBER
HWMSIZE		NUMBER
SHRINKS		NUMBER
WRAPS		NUMBER
EXTENDS		NUMBER
AVESHRINK		NUMBER
AVEACTIVE		NUMBER
STATUS		VARCHAR2(15)
CUREXT		NUMBER
CURBLK		NUMBER

23. Averiguar cuántos espacios de "undo" dedicados a gestión automática existen en la base de datos y ponerlos fuera de línea (comprobar este punto).

Realizamos una consulta a la vista `dba_tablespaces`

```
SQL> SELECT tablespace_name, contents, status  
2 FROM dba_tablespaces WHERE contents='UNDO';
```

TABLESPACE_NAME	CONTENTS	STATUS
UNDO_RBS1	UNDO	ONLINE
UNDO02	UNDO	ONLINE

Ponemos los espacio undo fuera de línea, y comprobamos

```
SQL> desc V$rollname
```

Name	Null?	Type
USN		NUMBER
NAME	NOT NULL	VARCHAR2(30)

```
SQL> ALTER tablespace undo_rbs1 offline;  
Tablespace altered.
```

```
SQL> ALTER tablespace undo02 offline;  
Tablespace altered.
```

```
SQL> SELECT tablespace_name, contents, status  
2 FROM dba_tablespaces WHERE contents='UNDO';
```

TABLESPACE_NAME	CONTENTS	STATUS
UNDO_RBS1	UNDO	OFFLINE
UNDO02	UNDO	OFFLINE

24. Averiguar de cuántos segmentos de "rollback" dispone la base de datos y cuáles de ellos están activos.

Consultamos la vista `dba_rollback_segs`

```
### Consultamos Cuantos estan Online
SQL> SELECT count(*) FROM dba_rollback_segs WHERE status='ONLINE';
COUNT(*)
```

1

```
### Consultamo cuales estan Activos
SQL> SELECT owner, SEGMENT_NAME, TABLESPACE_NAME, STATUS FROM
dba_rollback_segs
2 order by owner, segment_name, status;
```

OWNER	SEGMENT_NAME	TABLESPACE_NAME	STATUS

PUBLIC	_SYSSMU1\$	UNDO_RBS1	OFFLINE
PUBLIC	_SYSSMU2\$	UNDO_RBS1	OFFLINE
PUBLIC	_SYSSMU3\$	UNDO_RBS1	OFFLINE
PUBLIC	_SYSSMU4\$	UNDO_RBS1	OFFLINE
PUBLIC	_SYSSMU5\$	UNDO_RBS1	OFFLINE
PUBLIC	_SYSSMU6\$	UNDO002	OFFLINE
PUBLIC	_SYSSMU7\$	UNDO002	OFFLINE
PUBLIC	_SYSSMU8\$	UNDO002	OFFLINE

```
### Consulta
SQL> SELECT segment_name, tablespace_name, bytes, blocks, extents
2 FROM sys.dba_segments
3 WHERE segment_type = 'ROLLBACK';
```

SEGMENT_NAME	TABLESPACE_NAME	BYTES	BLOCKS	EXTENTS

SYSTEM	SYSTEM	1482752	724	27

25. Averiguar si existe un tablespace llamado RBS, si no crearlo. En adelante se trabajará con este espacio de almacenamiento exclusivo para los segmentos de "rollback".

Creamos el tablespace con las siguientes características

```
### Verificamos si existe el tablespace
SQL> SELECT tablespace_name FROM dba_tablespaces;
TABLESPACE_NAME
SYSTEM
TOOLS
UNDO_RBS1
TEMP
USERS
DATACURS001
INDCURS001
UNDO002
PRUEBA
9 rows selected.
### Dado que no existe lo creamos
SQL> create tablespace rbs
  2 datafile '$ORACLE_BASE/oradata/$ORACLE_SID/rbs_manual01.dbf'
  3 size 1M
  4 default storage (initial 100k next 100k minextents 2 maxextents 5);
Tablespace created.
```

26. Crear un segmento de "rollback" público, en el espacio de almacenamiento anterior, llamado RP01.

Creamos el segmento

```
SQL> CREATE PUBLIC ROLLBACK SEGMENT RP01 TABLESPACE RBS
  2 storage (initial 100k next 100k minextents 2 maxextents 10 optimal
500K);
CREATE PUBLIC ROLLBACK SEGMENT RP01 TABLESPACE RBS
*
ERROR at line 1:
ORA-01552: cannot use system rollback segment for non-system tablespace
'RBS'
```

Según la nota 1005227.6 para la solución a este problema, provocado porque debe haber como mínimo un segmento activado que no sea SYSTEM, deben seguirse los siguientes pasos:

- Crear un segmento de "rollback" en el espacio SYSTEM y activarlo.
- Crear los segmentos oportunos en el espacio dedicado a "rollback".
- Borrar o desactivar el segmento creado en SYSTEM

```
SQL> create public rollback segment r_borrar tablespace system
  2 storage (initial 100k next 100k minextents 2 maxextents 10 optimal
500K);
Rollback segment created.

SQL> ALTER rollback segment r_borrar online;
Rollback segment altered.

SQL> CREATE PUBLIC ROLLBACK SEGMENT RP01 TABLESPACE RBS
  2 storage (initial 100k next 100k minextents 2 maxextents 10 optimal
500K);
Rollback segment created.
```

27. **Observar los parámetros de almacenamiento con que se ha creado el segmento RP01. ¿Cuáles son?, ¿cuáles son los del espacio de almacenamiento RBS?**

Consultamos dba_rollback_segs

```
### Consulta para saber cuales son?
SQL> SELECT substr(segment_name,1,10) seg, substr(tablespace_name,1,10)
2 tables, INITIAL_EXTENT, NEXT_EXTENT, MIN_EXTENTS, MAX_EXTENTS FROM
3 dba_rollback_segs WHERE segment_name = 'RP01';
```

SEG	TABLES	INITIAL_EXTENT	NEXT_EXTENT	MIN_EXTENTS	MAX_EXTENTS
RP01	RBS	102400	102400	2	10

```
### Consulta para saber ¿cuáles son los del espacio de almacenamiento RBS
SQL> SELECT substr(tablespace_name,1,10) tables, INITIAL_EXTENT,
2 NEXT_EXTENT, MIN_EXTENTS, MAX_EXTENTS FROM dba_tablespaces
3 WHERE tablespace_name = 'RBS'
```

TABLES	INITIAL_EXTENT	NEXT_EXTENT	MIN_EXTENTS	MAX_EXTENTS
RBS	102400	102400	2	2147483645

28. **Crear un segmento de "rollback" privado llamado RPR01.**

```
SQL> CREATE ROLLBACK SEGMENT RPR01 TABLESPACE RBS;
Rollback segment created
```

29. **Activar los segmentos de "rollback" creados anteriormente.**

Activamos los segmentos de rollback de los puntos anteriores

```
SQL> ALTER rollback segment rp01 online;
Rollback segment altered.
```

```
SQL> ALTER ROLLBACK SEGMENT RPR01 ONLINE;
Rollback segment altered.
```

30. Averiguar cuáles de los segmentos de "rollback" anteriores son públicos y cuáles privados.

Realizamos la siguiente consulta

```
SQL> SELECT owner, segment_name FROM dba_rollback_segs;
OWNER  SEGMENT_NAME
-----
SYS     SYSTEM
PUBLIC  _SYSSMU1$
PUBLIC  _SYSSMU2$
PUBLIC  _SYSSMU3$
PUBLIC  _SYSSMU4$
PUBLIC  _SYSSMU5$
PUBLIC  _SYSSMU6$
PUBLIC  _SYSSMU7$
PUBLIC  _SYSSMU8$
PUBLIC  R_BORRAR
PUBLIC  RP01
SYS     RPR01

12 rows selected.
```

31. Averiguar en qué espacios de almacenamiento está definido cada uno de los segmentos de

```
SQL> SELECT substr(owner,1,10) prop, substr(segment_name,1,10) nombre,
tablespace_name FROM dba_rollback_segs order by segment_name;
PROP  NOMBRE  TABLESPACE_NAME
-----
PUBLIC RP01      RBS
SYS    RPR01      RBS
PUBLIC R_BORRAR   SYSTEM
SYS    SYSTEM    SYSTEM
PUBLIC _SYSSMU1$  UNDO_RBS1
PUBLIC _SYSSMU2$  UNDO_RBS1
PUBLIC _SYSSMU3$  UNDO_RBS1
PUBLIC _SYSSMU4$  UNDO_RBS1
PUBLIC _SYSSMU5$  UNDO_RBS1
PUBLIC _SYSSMU6$  UNDO002
PUBLIC _SYSSMU7$  UNDO002
PUBLIC _SYSSMU8$  UNDO002

12 rows selected.
```

"rollback" de la BD.

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Identificar cuáles son las vistas para los segmentos de rollback y UNDO*
- 3. Cuáles son los parámetros y valores asignados al final de esta práctica*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 07

Performance y Monitoreo

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la forma de monitorear gráficamente los recursos y el funcionamiento de la base de datos y en caso de observar alguna alarma o recomendación en el servidor, aplicar los cambios correspondientes para mejorar el rendimiento del sistema.
- Guiar al estudiante paso a paso en la forma de monitorear componentes y detectar alguna alarma o recomendación hecha por el sistema.
- Aplicar los cambios recomendados de mantenimiento preventivo o correctivo para el manejador de base de datos.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Performance y Monitoreo

SubTema(s): Mantenimiento Proactivo/Performance

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Preferentemente trabajar bajo Sistema Operativo Unix (Cualquier Arquitectura)
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Preparar archivos de configuración de parámetros de rendimiento
- Generar snapshots de la base de datos
- Aplicar recomendaciones de rendimiento
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el manejador de Base de Datos Oracle.
- Siempre se manejarán comandos de Linux o estatutos de SQL.

Martha Escamilla Zepeda

Desarrollo de la práctica

1. Monitoreo Proactivo. Crear un tablespace de 50MB

Se puede crear un Shell o simplemente ejecutar los queries en SQL*Plus

```
sqlplus / as sysdba << END
set echo on
drop tablespace tbsaddm including contents and datafiles;
CREATE SMALLFILE TABLESPACE "TBSADDM"
DATAFILE 'addm1.dbf' SIZE 50M
LOGGING
EXTENT MANAGEMENT LOCAL
SEGMENT SPACE MANAGEMENT MANUAL;
exit;
END
```

2. Crear un nuevo usuario ADDM, identificados por ADDM. Asignar el tablespace TBSADDM. Asigne el tablespace TEMP como tablespace temporal. Asignar los siguientes privilegios al usuario ADDM: CONNECT, RECURSOS, y DBA.

Se puede crear un Shell o simplemente ejecutar los queries en SQL*Plus

```
sqlplus / as sysdba << END
set echo on
drop user addm cascade;
create user addm identified by addm
default tablespace tbsaddm
temporary tablespace temp;
grant connect, resource, dba to addm;
exit;
END
```

3. Usar el DBMS_ADVISOR de base de datos para establecer el tiempo de actividad a 30 minutos. Como usuario ADDM, borrar y crear la tabla ADDM checar las estadísticas de esta tabla. Crear snapshot in Automatic Workload Repository (AWR)

Hay que generar actividad en la base de datos para ser analizada y se puede sobrecargar con los siguientes shell's

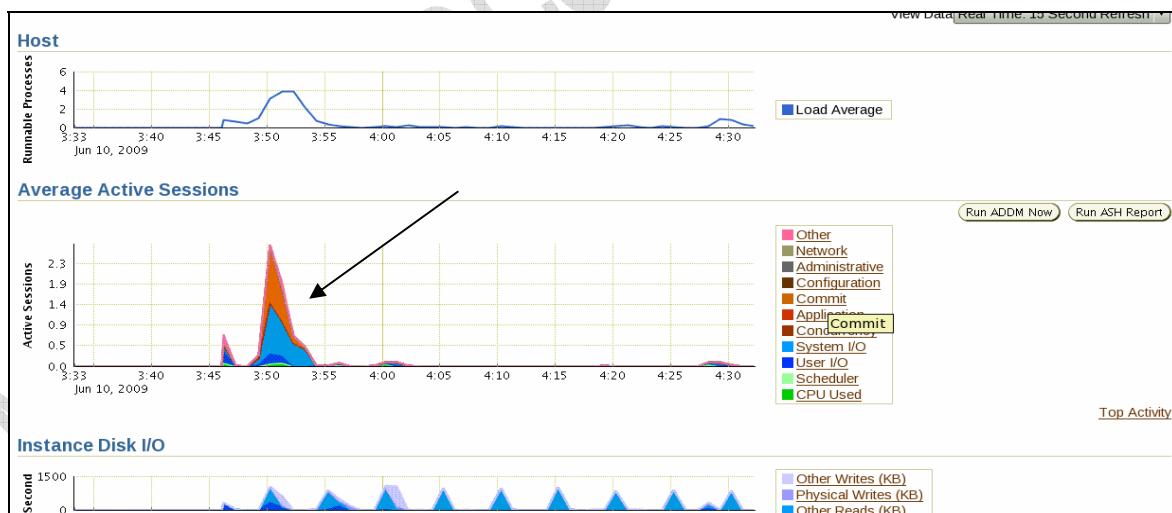
```
sqlplus / as sysdba << EOF
set echo on
exec dbms_advisor.set_default_task_parameter('ADDM','DB_ACTIVITY_MIN',30);
connect addm/addm
drop table addm purge;
```

```
create table addm(id number, name varchar2(2000));
exec DBMS_STATS.GATHER_TABLE_STATS(-
ownname=>'ADDM', tabname=>'ADDM',-
estimate_percent=>DBMS_STATS.AUTO_SAMPLE_SIZE);
exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT();
exit;
EOF
```

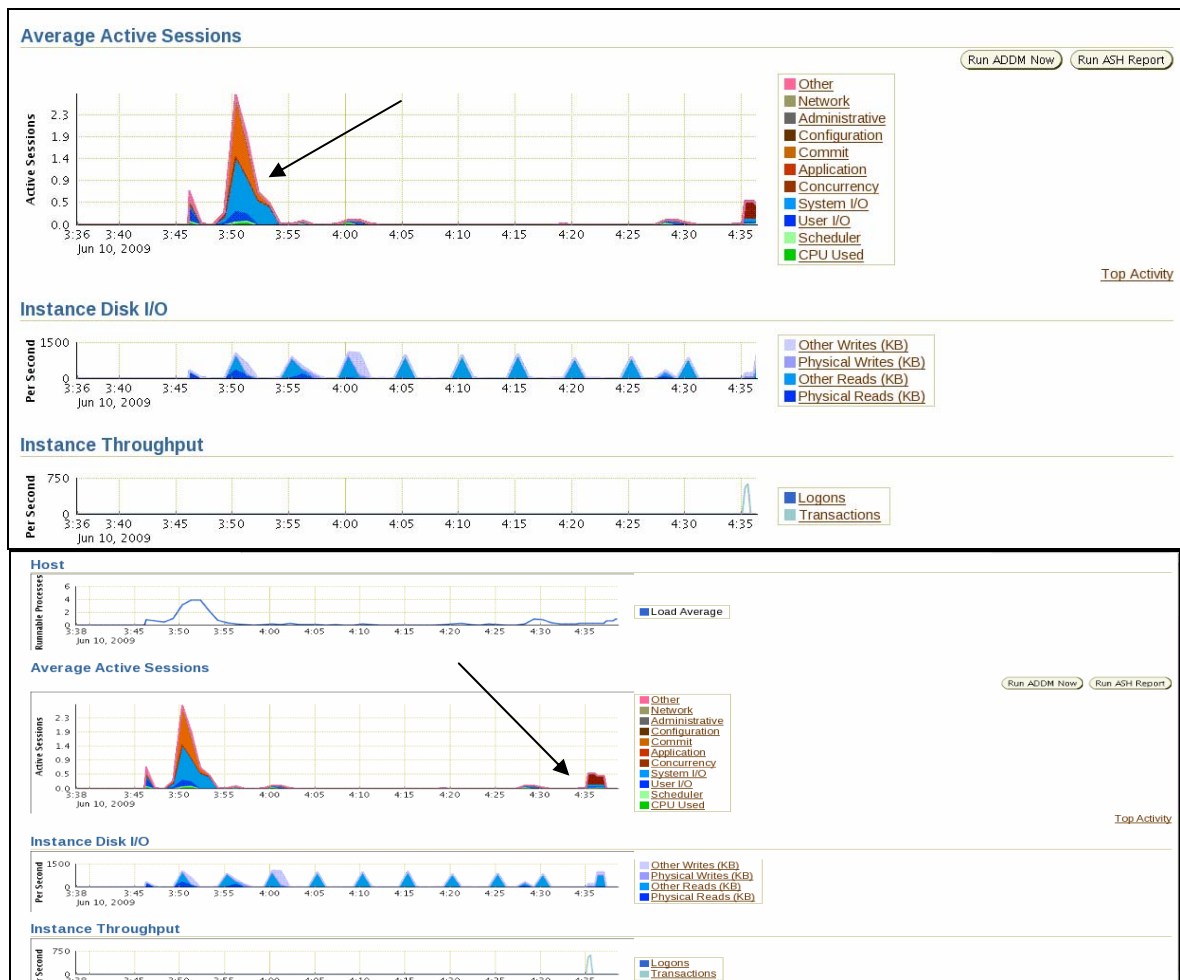
```
users=8 ; x=1;y=$users
UNPW="addm/addm"
while [ $x -le $y ]
do
    sqlplus -s $UNPW @lab_12_04.sql &
    x=`expr $x + 1`
done
```

4. En el Enterprise Manager, consultar los resultados como SYSDBA. Analizar los datos de rendimiento en tiempo real con un ciclo de 15-segundo de actualización en el EM. Después de un tiempo, deberá ver un pico en la gráfica del “Average Active Sessions”.

Esta es la actividad para ser analizada



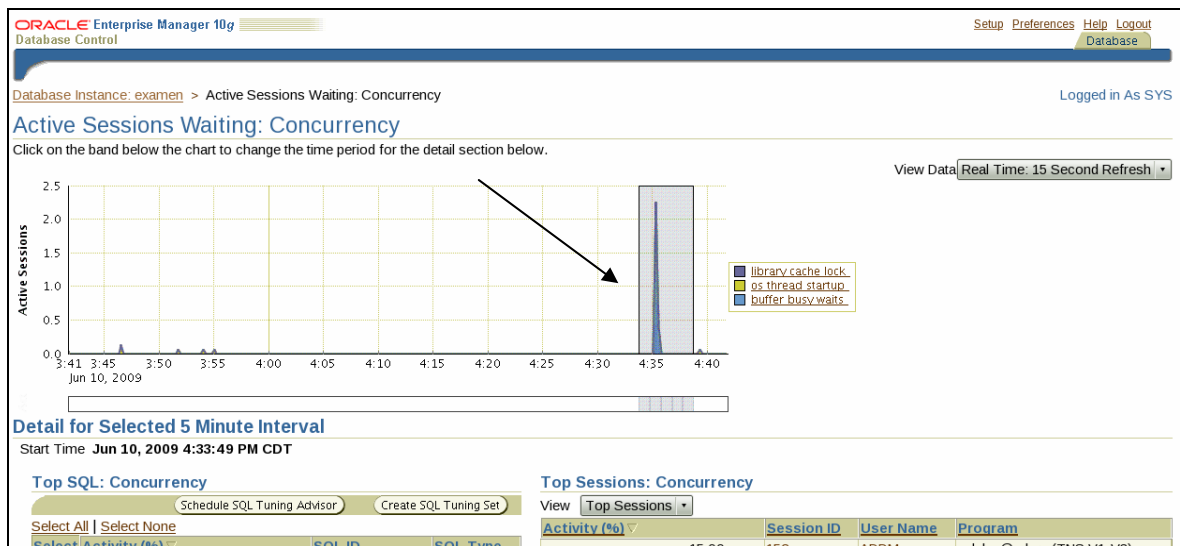
En cuanto a la gráfica, se puede determinar que está sufriendo problemas de concurrencia.



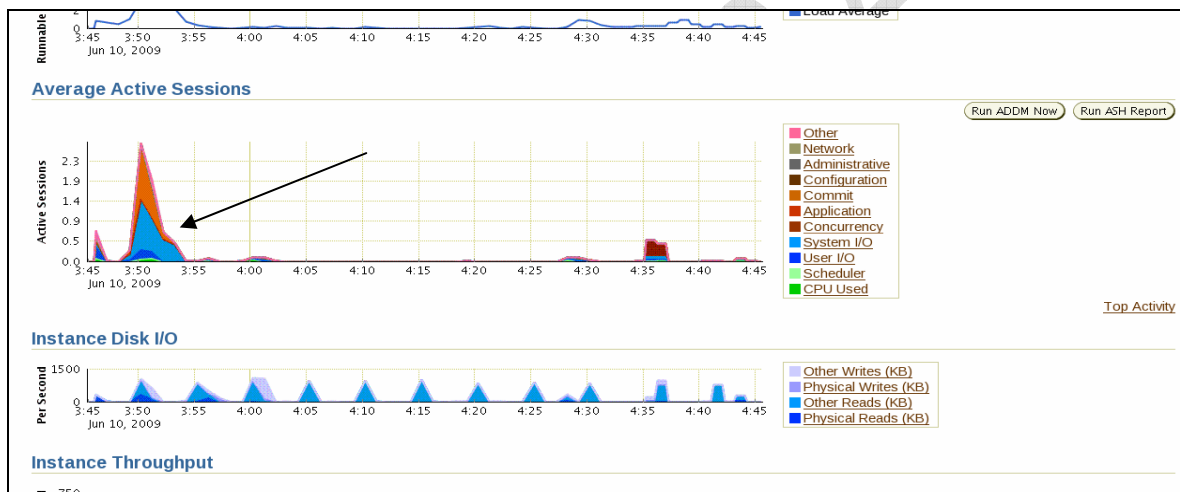
5. Crear una secuencia de comandos para forzar la creación de snapshot y recopilación de estadísticas sobre la tabla ADDM

```
sqlplus addm/addm << END
set echo on
exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT();
exec DBMS_STATS.GATHER_TABLE_STATS(
  ownname=>'ADDM', tabname=>'ADDM',
  estimate_percent=>DBMS_STATS.AUTO_SAMPLE_SIZE);
exit;
END
```

Observar las conclusiones de los análisis de los resultados del fin de su impacto



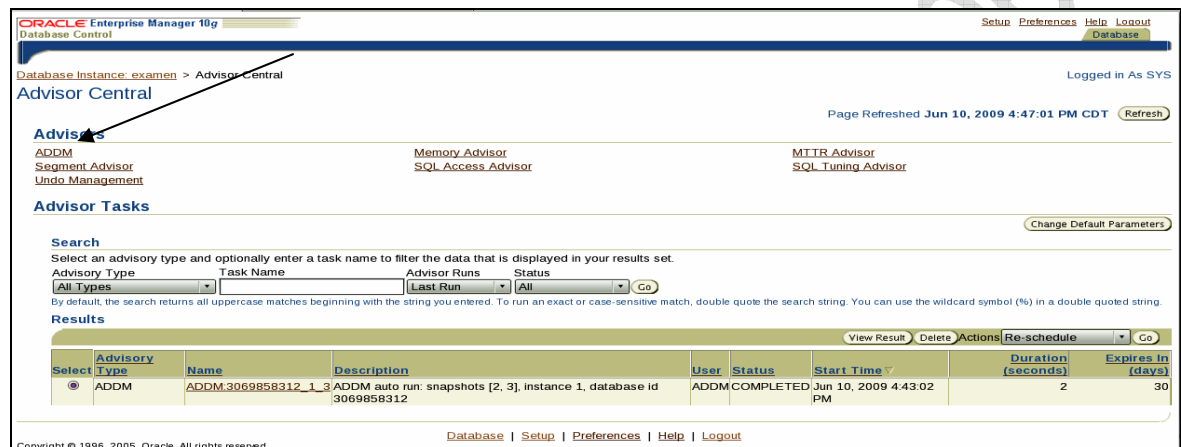
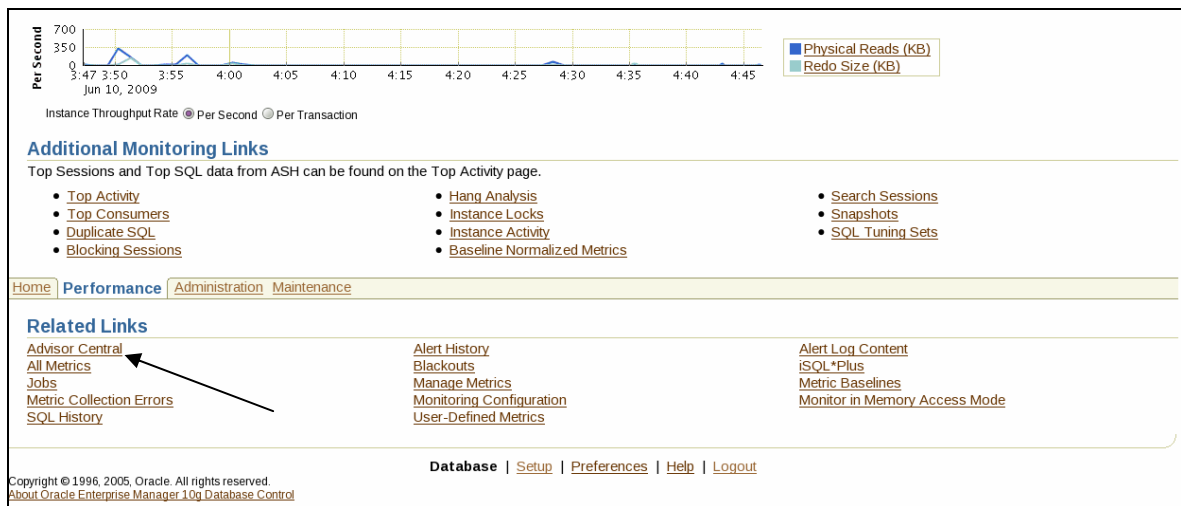
En cuanto a la sección de análisis de los resultados, verá que la primera irregularidad (en el SQL Tuning Recommendations category) tiene un 100% de impacto en el Sistema. Por lo tanto, hay que observar este hallazgo con más detalle.



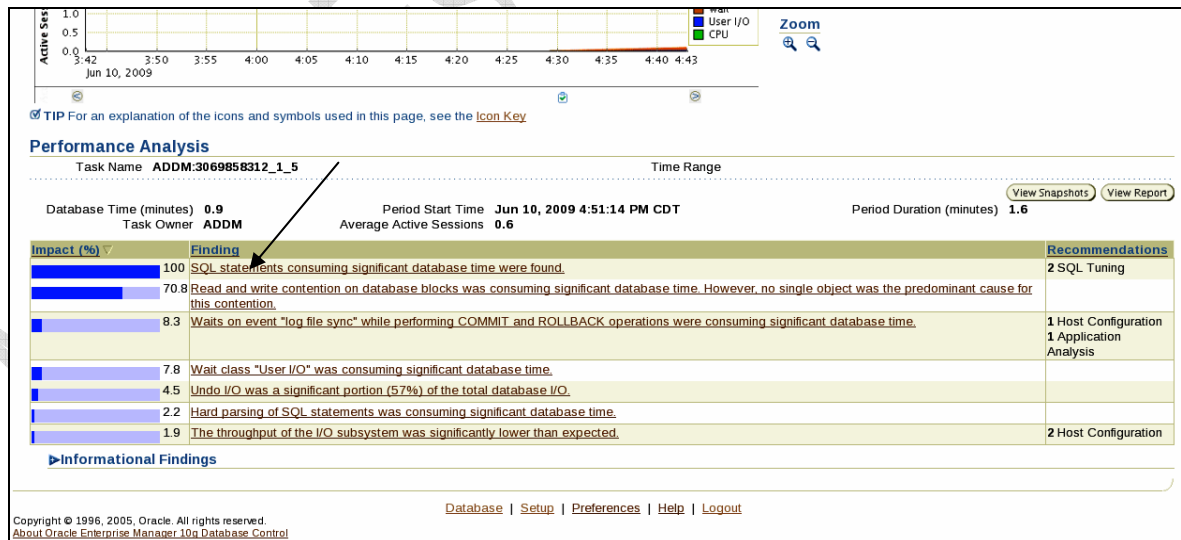
Sin embargo, buscando en esta sentencia SQL todavía no le ayudan a comprender el problema de la concurrencia de su base de datos.

6. Checando recomendaciones y características de la tabla ADDM

En cuanto a la sección de análisis de los resultados, checar que la conclusión tiene un 100% de impacto en el Sistema. Hay que examinar las recomendaciones sobre el rendimiento. Accesando el "AdvisorCentral" y después al ADDM.



Accesando en el "SQL statements consuming significant database time were found"



Examinar las recomendaciones sobre el rendimiento en la página de detalles

Database Time (minutes) 0.9 Task Owner ADDM Period Start Time Jun 10, 2009 4:51:14 PM CDT Period Duration (minutes) 1.6
Task Name ADDM:3069858312_1_5 Average Active Sessions 0.6

Finding SQL statements consuming significant database time were found.
Impact (minutes) 0.9
Impact (%) 100

Recommendations

Show All Details | Hide All Details

Details Category	Benefit (%)
SQL Tuning	86.2

Action Tune the PL/SQL block with SQL_ID "8brpf00mu7ubh". Refer to the "Tuning PL/SQL Applications" chapter of Oracle's "PL/SQL User's Guide and Reference"

SQL Text declare t number; begin for t in 1..2222 loop insert into addm values (Null,'a');
SQL ID 8brpf00mu7ubh

Rationale SQL statement with SQL_ID "8brpf00mu7ubh" was executed 8 times and had an average elapsed time of 5.8 seconds.

SQL Tuning

Action Investigate the SQL statement with SQL_ID "9n78dhpt5sh6f" for possible performance improvements.

SQL Text INSERT INTO ADDM VALUES (NULL,'a')
SQL ID 9n78dhpt5sh6f

Rationale SQL statement with SQL_ID "9n78dhpt5sh6f" was executed 17776 times and had an average elapsed time of 0.0025 seconds.

Findings Path

Expand All | Collapse All

Findings	Impact (%)	Additional Information
SQL statements consuming significant database time were found.	100	

Database | Setup | Preferences | Help | Logout

copyright © 1996, 2005, Oracle. All rights reserved.
About Oracle Enterprise Manager 10g Database Control

Regresando a la pagina "Automatic Database Diagnostic Monitor (ADDM)". Seleccionando "Read and write contention of database blocks was consuming significant database time"

The icon selected below the graph indicates the reason analyzed and period. Click on a different icon to select a different analysis period.

TIP For an explanation of the icons and symbols used in this page, see the [Icon Key](#)

Performance Analysis

Task Name ADDM:3069858312_1_6 (End Time Jun 10, 2009 4:56:46 PM) Time Range Jun 10, 2009 4:45:00 PM to Jun 10, 2009 5:15:00 PM

Database Time (minutes) 1.7 Task Owner ADDM Period Start Time Jun 10, 2009 4:52:51 PM CDT Period Duration (minutes) 3.9
Average Active Sessions 0.4

Impact (%)	Finding	Recommendations
100	SQL statements consuming significant database time were found.	2 SQL Tuning
68.9	A hot data block with concurrent read and write activity was found. The block belongs to segment "ADDM.ADDM" and is block 62 in file 8.	1 Schema 1 Application Analysis
68.9	Read and write contention on database blocks was consuming significant database time.	3 Schema
11.9	Waits on event "log file sync" while performing COMMIT and ROLLBACK operations were consuming significant database time.	1 Host Configuration 1 Application Analysis
5.9	The throughput of the I/O subsystem was significantly lower than expected.	2 Host Configuration
4.7	Undo I/O was a significant portion (47%) of the total database I/O.	
1	The SGA was inadequately sized, causing additional I/O or hard parses.	1 DB Configuration

Informational Findings

Se recomienda el uso del Automatic Segment Space Management feature para la tabla ADDM

ORACLE Enterprise Manager 10g Database Control

Database Instance: examen > Advisor Central > Automatic Database Diagnostic Monitor (ADDM) > Performance Finding Details

Performance Finding Details

Database Time (minutes) 1.7 Task Owner ADDM Period Start Time Jun 10, 2009 4:52:51 PM CDT Period Duration (minutes) 3.9
Task Name ADDM:3069858312_1_6 Average Active Sessions 0.4

Finding Read and write contention on database blocks was consuming significant database time.
Impact (minutes) 1.1
Impact (%) 68.9

Recommendations

Show All Details | Hide All Details

Details Category	Benefit (%)
Schema	68.9

Action Consider using ORACLE's recommended solution of automatic segment space management in a locally managed tablespace for the tablespace "TBSADDM" containing the TABLE "ADDM.ADDM" with object id 52663. Alternatively, you can move this object to a different tablespace that is locally managed with automatic segment space management.

Database Object ADDM.ADDM

Rationale There was significant read and write contention on TABLE "ADDM.ADDM" with object id 52663.

Database Object ADDM.ADDM

Schema

Action Consider partitioning the TABLE "ADDM.ADDM" with object id 52663 in a manner that will evenly distribute concurrent DML across multiple partitions.

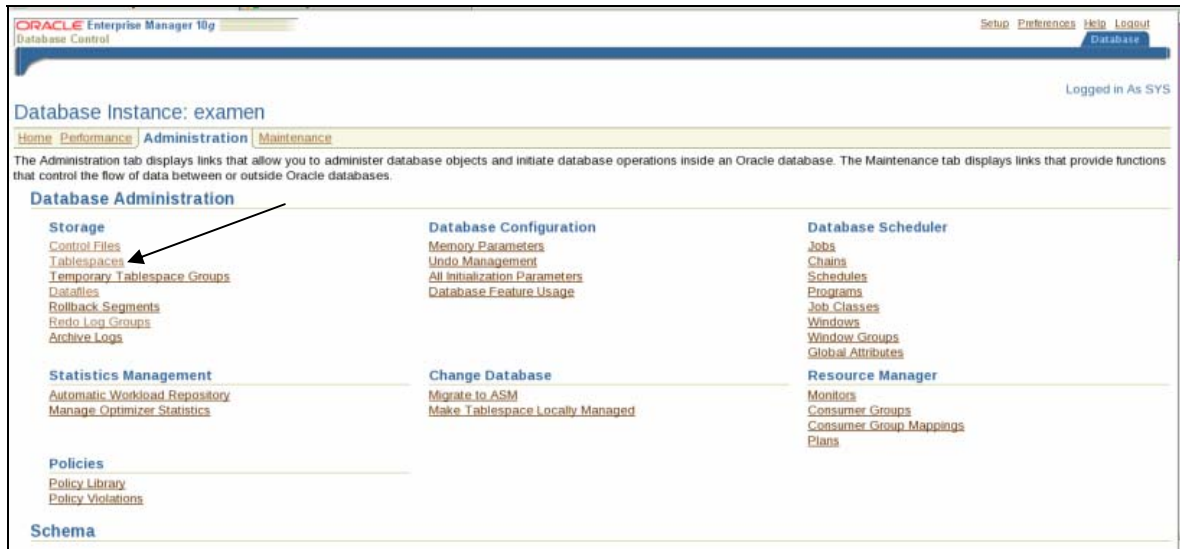
Database Object ADDM.ADDM

Rationale The INSERT statement with SQL_ID "9n78dhpt5sh6f" was significantly affected by "buffer busy" waits.

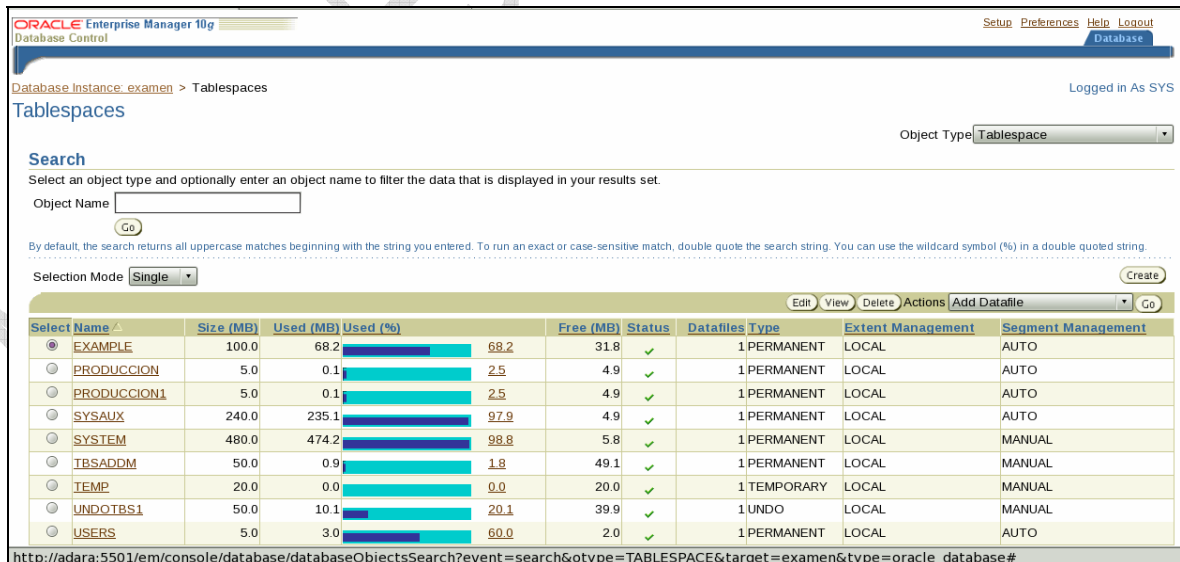
SQL Text INSERT INTO ADDM VALUES (NULL,'a')
SQL ID 9n78dhpt5sh6f

7. Para aplicar la recomendación, debe volver a recrear los objetos. Crear un nuevo tablespace llamado TBSADDM2 con un datafile de 50 MB, llamado addm2_1.dbf. Asegúrese de que el uso del tablespace TBSADDM2 sea automático en el segmento de la Gestión de espacio.

Creando el tablespace TBSADDM2 ubicándose en el área de administration del EM > seleccionando Storage < tablespace



Creando el tablespace TBSADDM2 dando click en crear



Completando los requisitos para la práctica

General **Storage** Show SQL Cancel OK

* Name

Extent Management

☒ Locally Managed
☐ Dictionary Managed

Type

☒ Permanent
☐ Set as default permanent tablespace
☐ Temporary
☐ Set as default temporary tablespace
☐ Undo
Undo Retention Guarantee ☐ Yes ☒ No

Status

☒ Read Write
☐ Read Only
☐ Offline

Datafiles

☐ Use bigfile tablespace
Tablespace can have only one datafile with no practical size limit.

Select Name	Directory	Size (MB)
No items found		

Add

General **Storage** Show SQL Cancel OK

Copyright © 1996, 2005, Oracle. All rights reserved.
[About Oracle Enterprise Manager 10g Database Control](#)

<http://adara:5501/em/console/database/storage/tablespace?event=create&cancelURL=/em/console/database/databaseObjectsSearch?event=redisplay&lastEvent=..>

ORACLE Enterprise Manager 10g
Database Control Setup Preferences Help Logout

Database Instance: examen > Tablespaces > Create Tablespace Logged in As SYS

Create Tablespace Show SQL Cancel OK

General **Storage**

Extent Allocation

☒ Automatic
☐ Uniform
Size KB

Segment Space Management

☒ Automatic
Objects in the tablespace automatically manage their free space. It offers high performance for free space management.
☐ Manual
Objects in the tablespace will manage their free space using free lists. It is provided for backward compatibility.

Enable logging

☒ Yes
Generate redo logs for creation of tables, indexes and partitions, and for subsequent inserts. Recoverable
☐ No
Redo log entries are smaller, the above operations are not logged and not recoverable.

También puede crear la tabla de manera manual con el siguiente SQL.

```
sqlplus addm/addm << END
set echo on
drop table addm purge;
create table addm(id number, name varchar2(2000)) tablespace TBSADDM2;
exec DBMS_STATS.GATHER_TABLE_STATS(-
ownname=>'ADDM', tabname=>'ADDM',-
estimate_percent=>DBMS_STATS.AUTO_SAMPLE_SIZE);
exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT();
exit;
END
```

```
users=8
x=1
```

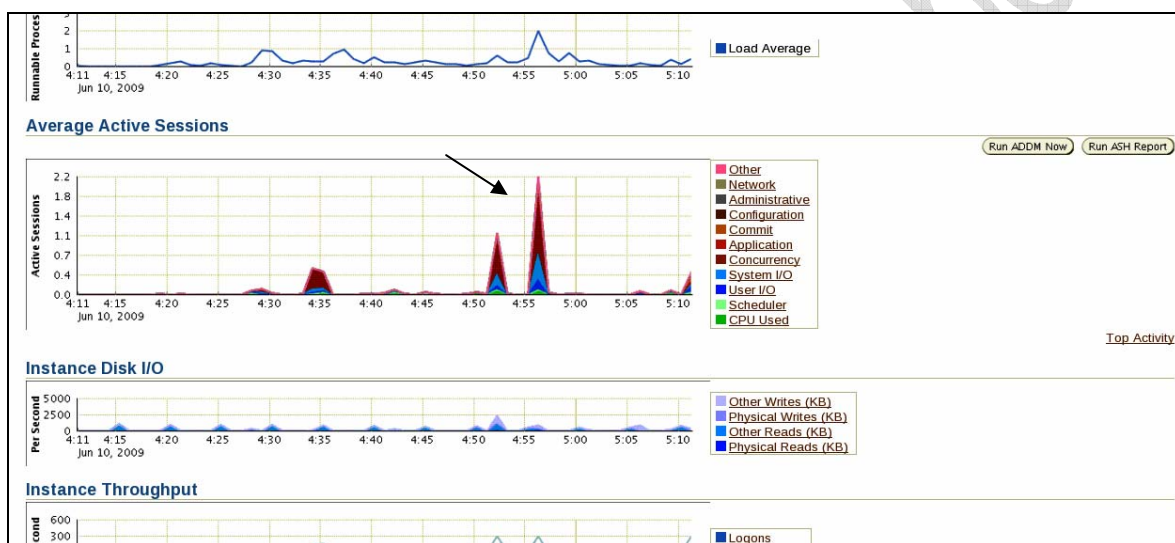
```

y=$users
UNPW="addm/addm"
while [ $x -le $y ]
do
    sqlplus -s $UNPW @lab_12_04.sql &
    x=`expr $x + 1`
done

```

8. En Enterprise Manager, consulte la página de resultados como un usuario conectado como SYSDBA. Ver datos de rendimiento en tiempo real con un periodo de 15-segundos.

Se presenta un pico con el cual se observa la concurrencia sobre la base de datos



Ejecutando estas queries para equilibrar el acceso de los datos al disco.

```

sqlplus addm/addm << END
|set echo on
exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT();
exec DBMS_STATS.GATHER_TABLE_STATS(
ownname=>'ADDM', tabname=>'ADDM',
estimate_percent=>DBMS_STATS.AUTO_SAMPLE_SIZE);
exit;
END

```

Se genera un snapshot y se observan resultados.

Database Control

Database Instance: examen > Advisor Central

Logged in As SYS

Advisor Central

Page Refreshed Jun 10, 2009 5:16:48 PM CDT [Refresh](#)

Advisors

[ADDM](#) [Memory Advisor](#) [MTTR Advisor](#)
[Segment Advisor](#) [SQL Access Advisor](#) [SQL Tuning Advisor](#)
[Undo Management](#)

Advisor Tasks [Change Default Parameters](#)

Search

Select an advisory type and optionally enter a task name to filter the data that is displayed in your results set.

Advisory Type: Task Name: Advisor Runs: Status: [Go](#)

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Results

Select	Advisory Type	Name	Description	User	Status	Start Time	Duration (seconds)	Expires In (days)
☒	ADDM	ADDM3069858312_1_7	ADDM auto run: snapshots [6, 7], instance 1, database id 3069858312	ADDM	COMPLETED	Jun 10, 2009 5:09:46 PM	0	30

Copyright © 1996, 2005, Oracle. All rights reserved.
 About Oracle Enterprise Manager 10g Database Control

Se observa claramente que el rendimiento mejoró, equilibrando la carga y evitando picos.

Automatic Database Diagnostic Monitor (ADDM)

Page Refreshed Jun 10, 2009 5:17:13 PM CDT [Refresh](#)

Database Activity

The icon selected below the graph identifies the ADDM analysis period. Click on a different icon to select a different analysis period.

☒ TIP For an explanation of the icons and symbols used in this page, see the [Icon Key](#).

Performance Analysis

Task Name: Time Range:

Database Time (minutes): 0.4 Period Start Time: Jun 10, 2009 4:56:46 PM CDT Period Duration (minutes): 13
 Task Owner: ADDM Average Active Sessions: 0

[View Snapshots](#) [View Report](#)

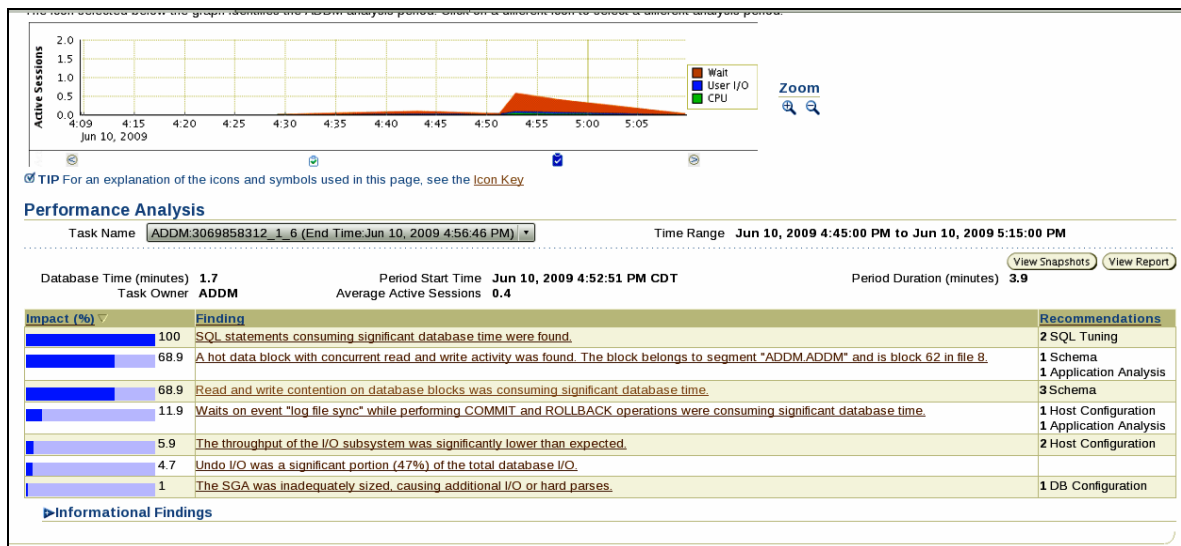
Impact (%)	Finding	Recommendations
ADDM did not find any problems during this period		

Informational Findings

There was no significant database activity to run the ADDM.

Copyright © 1996, 2005, Oracle. All rights reserved.

Y se observa que no hay ningún esquema de las recomendaciones. Moviendo la tabla ADDM a la administrada localmente con el tablespace TBSADDM2, que utiliza la automática Automatic Autoextend Segment feature, acción que fija la causa de este problema



9. Se limpia toda acción con el siguiente query

```
sqlplus / as sysdba << END
drop user addm cascade;
drop tablespace tbsaddm including contents and datafiles;
drop tablespace tbsaddm2 including contents and datafiles;
exit;
END
```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Monitorear y verificar el rendimiento de los recursos de la base de datos*
- 3. Para cada alerta anotar la recomendación y cómo se resolvió*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 08

Respaldo y Recuperación

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes a la administración y generación de los diferentes tipos de respaldos de la base de datos, así como la forma de recuperar la información en caso de falla o daño en los datos, con la finalidad de mantener la consistencia y confiabilidad de la base de datos.
- Aprender a definir y configurar los espacios de disco donde serán alojados los respaldos, así como los permisos y privilegios necesarios para poder realizar el respaldo o recuperación de datos.
- Guiar al estudiante paso a paso en la configuración y generación de los diferentes métodos de respaldo y recuperación.
- Conocer cuáles son las vistas y tablas donde la información de los respaldos quedan almacenadas y poder consultar la información correspondiente.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Respaldo y Recuperación

Subtema(s): Conceptos y consideraciones generales/Respaldando Base de Datos/Recuperando de Base de Datos/FlashBack

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Identificar y generar las diferentes formas de respaldos y recuperación
- Definir y crear las áreas de respaldos con privilegios y permisos necesarios.
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el

manejador de Base de Datos Oracle.

- *Siempre se manejarán comandos de Linux o estatutos de SQL.*

Martha Escamilla Zepeda

Desarrollo de la práctica

1. Comprobar que usuarios y roles tienen asignado el rol "exp_full_database".

Conectados como SYSDBA realizamos la siguiente consulta

```
SQL> SELECT * FROM dba_role_privs WHERE GRANTED_role='EXP_FULL_DATABASE';
```

GRANTEE	GRANTED_ROLE	ADM	DEF
DBA	EXP_FULL_DATABASE	NO	YES
SYS	EXP_FULL_DATABASE	YES	YES

2. Asignar el rol "exp_full_database" al rol "admin".

Para ello debemos tener un rol llamado admin creado en la práctica de usuarios, en caso contrario, lo creamos como

Se muestra

```
SQL> CREATE ROLE admin;
```

Role created

Asignando algunos privilegios

```
SQL> GRANT create session, create user, create role TO admin;
```

Asignamos el rol "exp_full_database" al rol "admin" y verificamos

```
SQL> GRANT exp_full_database to admin;  
GRANT succeeded.
```

```
SQL> SELECT * FROM dba_role_privs WHERE GRANTED_role='EXP_FULL_DATABASE';
```

GRANTEE	GRANTED_ROLE	ADM	DEF
DBA	EXP_FULL_DATABASE	NO	YES
SYS	EXP_FULL_DATABASE	YES	YES
ADMIN	EXP_FULL_DATABASE	NO	YES

2. En el Sistema de archivos \$ORACLE_BASE/export/<nombre_bd>, crear un archivo de parámetros llamado "par_admin01" que permita realizar una exportación total de la base de datos, incluyendo sus datos y lo permisos sobre todos los objetos, a un archivo llamado "total_admin01.dmp" (opciones "file", "full", "log"). Realizar la exportación como usuario "administrador". El registro de la exportación se guardará en el archivo "total_admin01.log".

Desde una sesión conectados como el usuario oracle, creamos los directorios "export"/"<nombre_bd>" bajo /opt/oracle, y dentro de esta estructura creamos un archivo llamado

```
$ mkdir $ORACLE_BASE/export
$ mkdir $ORACLE_BASE/export/$ORACLE_SID
$ vi $ORACLE_BASE/export/ORACLE_SID/par_admin01

### Agregamos las siguientes lineas:
buffer=102400
file=$ORACLE_BASE/export/$ORACLE_SID/total_admin01.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/total_admin01.log
full=y
```

par_Admin01.

Nos cambiamos al directorio antes creado y realizamos la exportación, utilizando como usuario al user administrador" creado en la práctica anterior.

```
$ cd $ORACLE_BASE/export/$ORACLE_SID/
$ exp parfile=par_admin01

Export: Release 9.2.0.1.0 - Production on MiÚ Nov 5 22:37:01 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Username: administrador
Password: admin

Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production
Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set

About to export the entire database ...
. exporting tablespace definitions
. exporting profiles
. exporting user definitions
. exporting roles
. exporting resource costs
. exporting rollback segment definitions
. exporting database links
. exporting sequence numbers
```

Revisamos el archivo "total_admin01.log" para comprobar si se ha producido algún error

```
$ vi total_admin01.log

### Nos vamos al final del archivo y verificamos
...
. exporting refresh groups and children
. exporting dimensions
. exporting post-schema procedural objects and actions
. exporting user history table
. exporting default and system auditing options
. exporting statistics
Export terminated successfully without warnings.
```

4. Realizar de nuevo el ejercicio 3 usando el modo directo de exportación. Defina el valor del parámetro **RECORDLENGTH**. El nombre de los archivos usados es:

- Archivo de parámetros: `part_admin01`
- Archivo de exportación: `Totald_admin01.dmp`
- Archivo Log: `Totald_admin01.log`

El archivo de parámetros queda de la siguiente manera

```
$ vi $ORACLE_BASE/export/$ORACLE_SID/part_admin01
### con los siguientes parámetros
file=$ORACLE_BASE/export/$ORACLE_SID/Totald_admin01.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/Totald_admin01.log
owner=(administrador)
compress=y
direct=y
RECORDLENGTH=65535
```

Realizamos la exportación como se muestra

```
$ exp parfile=Pard_admin01
Export: Release 9.2.0.1.0 - Production on MiÚ Nov 5 23:11:44 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Username: administrador
Password:

Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production
Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set

About to export specified users ...
. exporting pre-schema procedural objects and actions
. exporting foreign function library names for user ADMINISTRADOR
. exporting PUBLIC type synonyms
. exporting private type synonyms
...
```

Para los siguientes puntos de la práctica se puede trabajar sobre el usuario de muestra "scott" y su esquema, o se puede crear un usuario llamado "usuario01" con su esquema propuesto en el punto 5.

5. Crear un usuario llamado "usuario01" autenticado por "usuario01". Indicar como "tablespace" por default **USERS** y como "tablespace" temporal **TEMP**; asignar una cuota de 500K en el "tablespace" **USERS**. Asignar los roles **CONNECT** y **RESOURCE**. Conectado como usuario01 crear las tablas "Bonus", "Salario", "Emp" y "Depto". Con mínimo 2 campos por tabla.

Conectados como SYSTEM creamos el usuario usuario01, y le asignamos los privilegios mostrados

```
SQL> CREATE USER usuario01 IDENTIFIED BY usuario01
2 DEFAULT TABLESPACE USERS
3 TEMPORARY TABLESPACE TEMP
4 QUOTA 500K ON USERS;

SQL> GRANT CONNECT, RESOURCE TO usuario01;

GRANT succeeded.
```

Nos conectamos como usuario01 y creamos las tablas, (Insertar datos en las tablas)

```
SQL> conn usuario01/usuario01
Connected.

SQL> CREATE TABLE Bonus
2 (ID varchar2(3),
3 sal_bonus number);

Table created.

SQL> CREATE TABLE Salario
2 (ID varchar2(3),
3 SAL varchar2(30));

Table created.

SQL> CREATE TABLE emp
2 (ID varchar2(3),
3 NAME varchar2(30));

Table created.

SQL> CREATE TABLE depto
2 (ID varchar2(3),
3 NAME_DEPTO varchar2(15));

Table created.
```

6. Como usuario "usuario01", realizar una exportación de su esquema. Los parámetros se almacenarán en un archivo llamado "par_usuario01_total", la exportación en el archivo "usuario01_total.dmp" y el registro en el archivo "usuario01_total.log".

El archivo de parámetros tendrá el siguiente contenido

```
buffer=102400
file=$ORACLE_BASE/export/$ORACLE_SID/usuario01_total.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/usuario01_total.log
owner=(usuario01)
```

Realizamos la exportación.

```
$ exp parfile=par_usuario01_total
Export: Release 9.2.0.1.0 - Production on Jue Nov 6 01:12:17 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
Username: usuario01/usuario01
Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production
Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set
. exporting pre-schema procedural objects and actions
. exporting foreign function library names for user USUARIO01
...
...
. exporting post-schema procedural objects and actions
. exporting statistics
Export terminated successfully without warnings.
```

- 7. Conectado como usuario "usuario01", realizar una exportación de sus tablas "emp" y "depto". Los parámetros se almacenarán en un archivo llamado "par_usuario01", la exportación en el archivo "tablas_usuario01.dmp" y el registro en el archivo "tablas_usuario01.log".**

El contenido del archivo de parámetros será

```
$ vi $ORACLE_BASE/export/$ORACLE_SID/par_usuario01
buffer=102400
file=$ORACLE_BASE/export/$ORACLE_SID/tablas_usuario01.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/tablas_usuario01.log
tables=(EMP, DEPTO)
compress=v
```

Realizamos la exportación

```
$ exp parfile=par_usuario01
Export: Release 9.2.0.1.0 - Production on Jue Nov 6 00:09:31 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
Username: usuario01/usuario01
Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production
Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set

About to export specified tables via Conventional Path ...
. . exporting table EMP 0 rows exported
. . exporting table DEPTO 0 rows exported
Export terminated successfully without warnings.
```

8. Conectarse como usuario "usuario01" y eliminar la tabla "emp".

Conectados como usuario01, eliminamos la tabla "emp"

```
SQL> conn usuario01/usuario01
Connected.

SQL> DROP TABLE emp;

Table dropped.

### Verificamos cuales son las tablas "usuario01" con:
SQL> SELECT tname FROM TAB;
TNAME
-----
BONUS
DEPTO
SALARIO
```

9. Comprobar que usuarios y roles tienen asignado el rol "imp_full_database". Y asignar el rol "imp_full_database" al usuario "administrador".

Conectados como SYSTEM realizamos la consulta

```
SQL> SELECT * FROM dba_role_privs WHERE GRANTED_role='IMP_FULL_DATABASE';
```

GRANTEE	GRANTED_ROLE	ADM	DEF
DBA	IMP_FULL_DATABASE	NO	YES
SYS	IMP_FULL_DATABASE	YES	YES

Asignamos el rol "mp_full_database" en caso de que el usuario "administrador" no lo tenga

```
SQL> GRANT imp_full_database TO administrador;
Grant succeeded.
```

asignado

10. Como usuario "administrador", importar la tabla "usuario01.emp" desde el archivo "usuario01_total.dmp". Crear un archivo de parámetros llamado "par_imp_admin01" (parámetros file, fromuser y tables), el registro se guardar á en el archivo "imp_tabla_emp".

El contenido del archivo de parámetros de importación es

```
buffer=102400
file=$ORACLE_BASE/export/$ORACLE_SID/usuario01_total.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/imp_tabla_emp.log
tables=(EMP)
fromuser=usuario01
touser=usuario01
```

Realizamos la importación

```
$ imp parfile=par_imp_admin01
Import: Release 9.2.0.1.0 - Production on Jue Nov 6 01:32:25 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Username: administrador
Password:

Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 -
Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production

Export file created by EXPORT:V09.02.00 via conventional path

Warning: the objects were exported by USUARIO01, not by you

import done in WE8MSWIN1252 character set and AL16UTF16 NCHAR
character set
. importing USUARIO01's objects into USUARIO01
. . importing table "EMP" 0 rows imported
Import terminated successfully without warnings.
```

11. Verificar las tablas existentes en la BD para el usuario "usuario01"

Realizamos la consulta conectados como "usuario01".

```
SQL> SELECT tname FROM TAB;
TNAME
-----
BONUS
DEPTO
EMP
SALARIO
```

12. Como usuario "administrador", realizar una importación a archivo usando el archivo "usuario01_total.dmp" (parámetro SHOW=Y). Crear un archivo de parámetros llamado "par_imparchivo_admin04"; el registro se guardará en el archivo "imp_usuario01_archivo".

El contenido del archivo de parámetros de importación es

```
buffer=102400
file=$ORACLE_BASE/export/$ORACLE_SID/usuario01_total.dmp
log=$ORACLE_BASE/export/$ORACLE_SID/imp_usuario01_archivo.log
full=y
show=y
```

Realizamos la importación El resultado de la importación, las sentencias de creación, se almacenará en el archivo de log.

```
$ imp parfile=par_imparchivo_admin04
Import: Release 9.2.0.1.0 - Production on Jue Nov 6 01:51:40 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Username: administrador/admin

Connected to: Oracle10g Enterprise Edition Release 9.2.0.1.0 -
Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production

Export file created by EXPORT:V09.02.00 via conventional path

Warning: the objects were exported by USUARIO01, not by you

import done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character
set
. importing USUARIO01's objects into ADMINISTRADOR
"BEGIN "
"sys.dbms_logrep_imp.instantiate_schema(schema_name=>'USUARIO01',
export_db_"
"name=>'DBUNIX', inst_scn=>'166031');"
"COMMIT; END;"
"CREATE TABLE "BONUS" ("ID" VARCHAR2(3), "SAL_BONUS" NUMBER) PCTFREE 10
PCT"
"USED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1
FREELIST "
"GROUPS 1) TABLESPACE "USERS" LOGGING NOCOMPRESS"
. . skipping table "BONUS"

"CREATE TABLE "DEPTO" ("ID" VARCHAR2(3), "NAME_DEPTO" VARCHAR2(30))
PCTFREE"
" 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS
1 FR"
"EELIST GROUPS 1) TABLESPACE "USERS" LOGGING NOCOMPRESS"
. . skipping table "DEPTO"

"CREATE TABLE "EMP" ("ID" VARCHAR2(3), "NAME" VARCHAR2(30)) PCTFREE 10
PCTU"
"SED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1
FREELIST G"
"ROUPS 1) TABLESPACE "USERS" LOGGING NOCOMPRESS"
. . skipping table "EMP"

"CREATE TABLE "SALARIO" ("ID" VARCHAR2(3), "SAL" VARCHAR2(30)) PCTFREE
10 P"
"CTUSED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1
FREELIS"
"TG GROUPS 1) TABLESPACE "USERS" LOGGING NOCOMPRESS"
. . skipping table "SALARIO"

Import terminated successfully without warnings.
```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Cómo verificas que cada respaldo se realice correctamente.*
- 3. Cuáles son los archivos de parámetros, explicando claramente el porqué de cada parámetro definido.*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

Práctica 09

Moviendo Datos

Objetivos

- El objetivo de esta práctica es que el estudiante ponga en práctica los conocimientos teóricos adquiridos referentes los diferentes métodos de mover la información de una base de datos, que puede ser requerida para migración, actualización, depuración o mantener la consistencia de la misma.
- Aprender la diferencia entre los métodos de carga de datos de SQL*Loader, Export-Import y Datapump y en qué caso es recomendable la aplicación de cada técnica.
- Aprender a definir y crear los espacios de disco donde se moverán los datos, así como la creación de permisos, privilegios y archivos de configuración necesarios para poder realizar estas tareas.
- Guiar al estudiante paso a paso en la configuración y generación de los diferentes métodos de carga o movimiento de datos.
- Permitir con el reporte del estudiante, el alumno afirme aún más dicho conocimiento.

Tema: Moviendo Datos

Subtema(s): Carga Conceptos/SQL Loader/Export/ Import

Material y equipo necesario

- Material teórico visto en clase sobre el tema y subtema correspondiente.
- Equipos de Laboratorio o computadora personal.
- Sistema Operativo Linux ya instalado, cualquier distribución y versión.
- Software del Manejador de Base de Datos, en este caso Oracle10g.

Metodología

- Creación de un usuario
- Creación de tablas
- Edición del archivo de control, de parámetros y de datos.
- Siempre las pantallas descriptivas de color verde se refieren a comandos Unix.
- Siempre las pantallas descriptivas de color azul se refieren a estatutos de SQL para el

manejador de Base de Datos Oracle.

- *Siempre se manejarán comandos de Linux o estatutos de SQL.*

Martha Escamilla Zepeda

Desarrollo de la práctica

Ejercicio 1. MOVIENDO DATOS CON SQLoader

1. Crear al usuario user1 y darle permisos de crear sesión y de resource.

Dentro del sql*plus.

```
SQL> create user user1 identified by user1;
User created.
SQL> grant create session, resource to user1;
Grant succeeded.
```

Conectarse como el usuario user1 y crear la tabla t1 con campos c1,c2, ambos tipo int

```
SQL> conn user1/user1;
Connected.
SQL> create table t1(
    2 c1 int,
    3 c2 int
    4 );
Table created.
```

2. Desde una sesión conectado como oracle acceder al directorio \$ORACLE_HOME y crear un directorio llamado cargas

```
oracle@host:/u01/app/oracle$ cd $ORACLE_HOME
oracle@host:/u01/app/oracle/product/10.2.0/db_1$ mkdir -v cargas
mkdir: created directory `cargas'
```

3. Dentro del directorio cargas, crear un archivo llamado data.datl con el siguiente contenido

```
$ vi data.datl
100,3
200,4
300,5
400,6
500,7
```

4. Dentro del directorio cargas, crear un archivo llamado control.ctl con el siguiente contenido

```
$ vi control.ctl
load data
infile 'data.dat'
into table t1
fields terminated by ","
( c1, c2 )
```

5. Dentro del directorio cargas, crear un archivo llamado parametros.par con el siguiente contenido, que serán los parámetros de ejecución de la carga de datos.

```
$ vi parametros.par
USERID=user1/user1
CONTROL=control.ctl
LOG=logfile.log
```

6. Listar el contenido del directorio cargas para comprobar la creación de los tres archivos

```
oracle@host:/u01/app/oracle/product/10.2.0/db_1/cargas$ ls
control.ctl data.dat parametros.par
```

7. En el sql*plus como user1 hacer un select del contenido de la tabla t1

```
SQL> select * from t1;
no rows selected
```

8. En una sesión dentro del directorio cargas ejecutar el SQL*Loader utilizando el archivo de parámetros que

creamos con el comando "sqlldr parfile=parametros.par"

```
oracle@host:/u01/app/oracle/product/10.2.0/db_1/cargas$ sqlldr parfile=parametros.par
SQL*Loader: Release 10.2.0.1.0 - Production on Fri Jun 12 04:26:00 2009
Copyright (c) 1982, 2005, Oracle. All rights reserved.
Commit point reached - logical record count 5
```

En esta parte se realizó la carga de los registros a la tabla T1

9. En el SQL*Plus como user1 hacer una consulta del contenido de la tabla t1 para comprobar la carga

```
SQL> select * from t1;
```

C1	C2
100	3
200	4
300	5
400	6
500	7

Observar el contenido del archivo logfile.log

```
$ vi logfile.log
```

```
SQL*Loader: Release 10.2.0.1.0 - Production on Fri Jun 12 04:26:00 2009
```

```
Copyright (c) 1982, 2005, Oracle. All rights reserved.
```

```
Control File: control.ctl
```

```
Data File: data.dat
```

```
Bad File: data.bad
```

```
Discard File: none specified
```

```
(Allow all discards)
```

```
Number to load: ALL
```

```
Number to skip: 0
```

```
Errors allowed: 50
```

```
Bind array: 64 rows, maximum of 256000 bytes
```

```
Continuation: none specified
```

```
Path used: Conventional
```

```
Table T1, loaded from every logical record.
```

```
Insert option in effect for this table: INSERT
```

Column Name	Position	Len	Term	Encl	Datatype
-------------	----------	-----	------	------	----------

C1	FIRST	*	,		CHARACTER
----	-------	---	---	--	-----------

C2	NEXT	*	,		CHARACTER
----	------	---	---	--	-----------

```
Table T1:
```

```
5 Rows successfully loaded.
```

```
0 Rows not loaded due to data errors.
```

```

0 Rows not loaded because all WHEN clauses were failed.
0 Rows not loaded because all fields were null.
Space allocated for bind array:          33024 bytes(64 rows)
Read  buffer bytes: 1048576
Total logical records skipped:      0
Total logical records read:         5
Total logical records rejected:     0
Total logical records discarded:    0
Run began on Fri Jun 12 04:26:00 2009
Run ended on Fri Jun 12 04:26:00 2009
Elapsed time was:   00:00:00.08
CPU time was:      00:00:00.03

```

10. Borrar el contenido de la tabla t1 y hacer commit

```

SQL> delete from t1;
5 rows deleted.
SQL> commit;
Commit complete.

```

11. En una sesión dentro del directorio cargas ejecutar el SQL*Loader utilizando pasando como parámetro el userid y contraseña, e indicando cual es el archivo de control con el comando "sqlldr userid=user1/user1 control=control.ctl "

```

oracle@host:/u01/app/oracle/product/10.2.0/db_1/cargas$ sqlldr userid=user1/user1
control=control.ctl
SQL*Loader: Release 10.2.0.1.0 - Production on Fri Jun 12 04:35:08 2009
Copyright (c) 1982, 2005, Oracle. All rights reserved.
Commit point reached - logical record count 5

```

12. En el SQL*Plus como user1 hacer un select del contenido de la tabla t1

```

SQL> select * from t1;

   C1      C2
-----
   100      3
   200      4
   300      5
   400      6
   500      7

```

De nuevo se pueden observar los 5 registros que subió debido a que los primeros se eliminaron anteriormente.

- 13. Como el usuario user1, crear la tabla poblacion, con un registro num tipo number, nombre tipo varchar, hombres int, mujeres tipo int**

```
SQL> create table poblacion(  
2 num number,  
3 nombre varchar(30),  
4 hombres int,  
5 mujeres int  
6);  
Table created.
```

- 14. Dentro del directorio cargas crear el archivo de control poblacion.ctl, que contenga los datos a insertar, que introduzca automáticamente el campo num de forma incremental y que maneje un tamaño fijo para los registros, con el siguiente contenido**

```
$ vi población.ctl  
LOAD DATA  
infile *  
APPEND  
INTO TABLE poblacion(  
num SEQUENCE(COUNT,1),  
nombre POSITION(07:35) CHAR,  
hombres POSITION(44:49) INTEGER EXTERNAL,  
mujeres POSITION(51:56) INTEGER EXTERNAL)  
begindata  
Oaxaca 21484 10422 11062  
Guerrero 726 377 349  
Veracruz 10412 5167 5245  
Durango 6770 3400 3370  
Hidalgo 4163 2098 2065  
Guadalajara 697487 333469 364018  
Mexico 8660 4326 4334  
Monterrey 16980 8393 8587
```

Esta es una forma implícita de subir datos.

15. Hacer una consulta del contenido de la tabla poblacion

```
SQL> select * from poblacion;  
no rows selected
```

16. Cargar los datos a la tabla poblacion utilizando el archivo de control y el identificador del usuarios

```
oracle@host:/u01/app/oracle/product/10.2.0/db_1/cargas$ sqlldr userid=user1/user1  
control=poblacion.ctl
```

```
SQL*Loader: Release 10.2.0.1.0 - Production on Fri Jun 12 04:59:35 2009
```

```
Copyright (c) 1982, 2005, Oracle. All rights reserved.
```

```
Commit point reached - logical record count 8
```

17. Hacer un select del contenido de la tabla poblacion

```
SQL> select * from poblacion;  
  
      NUM      NOMBRE      HOMBRES  MUJERES  
-----  
      1 Oaxaca      10422    11062  
      2 Guerrero      377      349  
      3 Veracruz      5167    5245  
      4 Durango      3400    3370  
      5 Hidalgo      2098    2065  
      6 Guadalajara  333469  364018  
      7 Mexico      4326    4334  
      8 Monterrey    8393    8587
```

```
8 rows selected.
```

18. Observar el contenido del archivo pobalcion.log

```
SQL*Loader: Release 10.2.0.1.0 - Production on Fri Jun 12 05:06:31 2009
```

```
Copyright (c) 1982, 2005, Oracle. All rights reserved.
```

```
Control File: poblacion.ctl
```

```
Data File: poblacion.ctl
```

```
Bad File: poblacion.bad
```

```
Discard File: none specified
```

```
(Allow all discards)
```

```

Number to load: ALL
Number to skip: 0
Errors allowed: 50
Bind array: 64 rows, maximum of 256000 bytes
Continuation: none specified
Path used: Conventional
Table POBLACION, loaded from every logical record.
Insert option in effect for this table: APPEND
  Column Name          Position  Len  Term Encl Datatype
-----
NUM                     SEQUENCE (COUNT, 1)
NOMBRE                  7:35  29    CHARACTER
HOMBRES                 44:49  6     CHARACTER
MUJERES                 51:56  6     CHARACTER
Table POBLACION:
  8 Rows successfully loaded.
  0 Rows not loaded due to data errors.
  0 Rows not loaded because all WHEN clauses were failed.
  0 Rows not loaded because all fields were null.
Space allocated for bind array:          4544 bytes(64 rows)
Read  buffer bytes: 1048576
Total logical records skipped:          0
Total logical records read:             8
Total logical records rejected:          0
Total logical records discarded:         0
Run began on Fri Jun 12 05:06:31 2009
Run ended on Fri Jun 12 05:06:31 2009
Elapsed time was: 00:00:00.12
CPU time was:    00:00:00.01

```

Ejercicio 2. MOVIENDO DATOS CON Export/Import

Mediante unos ejemplos vamos a explicar cómo realizar el export, por supuesto existirán más formas de hacerlo según los parámetros definidos. El **comando** para realizar export es **exp**. Para ver todos los argumentos y significado de ellos de un export basta con realizar un **exp help=yes**

1. Export. Copia completa de la base de datos

```
$exp file=/oracle/export_orasite.dmp full=yes log=/oracle/log/export_orasite.log  
buffer=1000000
```

file=/oracle/export_orasite.dmp Nombre y ubicación del archivo del export (el usuario oracle tiene que tener permisos para escribir ahí. La extensión de este archivo es dmp.

full=yes Con esto indicamos que el export es completo, todos los esquemas de la base de datos y sus datos permisos, privilegios.

log=/oracle/log/export_orasite.log Archivo para tener el log del export, el usuario oracle también tiene que tener permisos en ese directorio para escribir.

buffer=1000000 Reservamos buffer para la operación

6. Export. Copia de tablas específicas de un usuario

```
$exp scott/tiger file=orasitescott.dmp tables=(emp,dept) buffer=1000000
```

En este caso realizamos sólo un backup de las tablas especificadas en el argumento tables del usuario scott

7. Export. Copiar tablas de un usuario con una condición específica

Exportamos la tabla emp del usuario scott y en el parámetro **query**, especificamos una condición para realizar el export de esa tabla.

```
$exp scott/tiger file=c:\orasitempleados.dmp tables=emp query=\"where deptno=10\"
```

8. **Import.** Partiendo de un archivo realizado con la utilizad export podemos recuperar datos de toda la base de datos, de ciertas tablas. El **comando** para recuperar es **imp**. Para ver todos los argumentos y significado de ellos de un export basta con realizar un **imp help=yes**.

Importar todo el archivo exportado

```
$imp system/manager file=c:\orasitefull.dmp full=yes ignore=yes  
log=c:\orasite\log\import_log.log buffer=1000000
```

Importamos con el usuario system que tiene permisos para importar el archivo orasitefull.dmp dejando un log de dicha importación en import_log.log reservando un buffer de 100000

5.Import. Importar una tabla de un usuario específico

```
$imp scott/tiger file=orasitempleados.dmp fromuser=scott touser=scott tables=dept
```

Importamos del archivo orasitempleados.dmp sólo del usuario scott de ese archivo al esquema del usuario scott la tabla departamento

Ejercicio 3. Moviendo datos con DataPump

1. La utilidad DataPump es la que ORACLE10g utiliza para realizar los exports. (expdp). Para poder hacer uso de esta utilidad hay que especificar el directorio que va a ser utilizado por ORACLE10g Export/Import (impdp) para poder exportar e importar los datos.

Existen diversas formas de especificar este directorio, una de las formas que se puede utilizar es la siguiente

```
SQL > CONNECT system/manager
SQL > CREATE OR REPLACE DIRECTORY export as '/backup/database/BBDD/orasite';
-- creamos el mapeo del directorio export a /backup/database/BBDD/orasite
SQL > CREATE OR REPLACE DIRECTORY log as '/backup/database/BBDD/orasite/log';
-- creamos el mapeo del directorio export a /backup/database/BBDD/orasite/log
SQL > GRANT read,write ON DIRECTORY export TO system
SQL > GRANT read, write ON DIRECTORY log TO system
```

Otorgamos permisos lectura, escritura a esos directorios. En este caso es system el usuario encargado de hacer el export, por eso se le otorgan los privilegios. **Directory:** Este parámetro especifica la localización en la cual Export DataPump o Import DataPump pueden escribir el “dump file”, “log file”, “SQL file”

Para revisar los directorios disponibles podemos utilizar la siguiente consulta. Los permisos de lectura y escritura en un directorio sólo significan que la base de datos ORACLE deberá leer o escribir el archivo en esa ubicación.

```
SQL > SELECT privilege, directory_name FROM user_tab_privs t, all_directories d WHERE
t.table_name(+) = d.directory_name ORDER BY 2,1;
```

2. Datapump. Dados los permisos anteriormente, podemos realizar un export completo de la base de datos.

```
$ expdp system/manager DUMPFILE=export:orasite.dmp FULL=Y LOGFILE=log:orasite.log
```

Realizamos el export con el usuario system (system/manager). Ponemos el nombre del archivo mediante DUMPFILE indicándole el directorio que hemos creado y dado permisos para el efecto. Indicamos el nombre del archivo log mediante LOGFILE indicándole el directorio creado para ubicar estos archivos (log)

3. Para los ejemplos de de oracle, primero desbloquear la cuenta SCOTT y otorgar permiso de crear un objeto de directorio.

```
CONN /AS SYSDBA

SQL>ALTER USER scott IDENTIFIED BY tiger ACCOUNT UNLOCK;
GRANT CREATE ANY DIRECTORY TO scott;

SQL>CREATE OR REPLACE DIRECTORY test_dir AS '/u01/app/oracle/oradata/';
GRANT READ, WRITE ON DIRECTORY test_dir TO scott;

SQL> connect /as sysdba
Connected to an idle instance.

SQL> alter user scott identified by tiger account unlock;
User altered.

SQL> grant create any directory to scott;
Grant succeeded.

SQL> create or replace directory test_dir as '/u01/app/oracle/oradata/';
Directory created.

SQL> grant read, write on directory test_dir to scott; Grant succeeded.
```

4. Datapump. Este es un ejemplo de la exportación e importación de tablas.

Exportar las tablas emp y dept

```
SQL> host expdp scott/tiger tables=EMP,DEPT directory=TEST_DIR dumpfile=EMP_DEPT.dmp
logfile=expdpEMP_DEPT.log

Export: Release 10.2.0.1.0 - Production on Thursday, 11 June, 2009 17:00:18
Copyright (c) 2003, 2005, Oracle. All rights reserved.
Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options
Starting "SCOTT"."SYS_EXPORT_TABLE_01": scott/***** tables=EMP,DEPT directory=TEST_DIR
dumpfile=EMP_DEPT.dmp logfile=expdpEMP_DEPT.log
Estimate in progress using BLOCKS method...
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
Total estimation using BLOCKS method: 128 KB
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/INDEX/INDEX
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
.. exported "SCOTT"."DEPT"                5.656 KB    4 rows
.. exported "SCOTT"."EMP"                  7.820 KB   14 rows
Master table "SCOTT"."SYS_EXPORT_TABLE_01" successfully loaded/unloaded

Dump file set for SCOTT.SYS_EXPORT_TABLE_01 is:
/u01/app/oracle/oradata/EMP_DEPT.dmp
Job "SCOTT"."SYS_EXPORT_TABLE_01" successfully completed at 17:00:47
```

Importando las tablas de emp y dept. El parámetro TABLE_EXISTS_ACTION=APPEND permite que se puedan importar las tablas existentes.

```
SQL> host impdp scott/tiger tables=EMP,DEPT directory=TEST_DIR dumpfile=EMP_DEPT.dmp
logfile=impdpEMP_DEPT.log
```

```
Import: Release 10.2.0.1.0 - Production on Thursday, 11 June, 2009 17:02:18 Copyright (c)
2003, 2005, Oracle. All rights reserved. Connected to: Oracle Database 10g Enterprise
Edition Release 10.2.0.1.0 - Production With the Partitioning, OLAP and Data Mining
options Master table "SCOTT"."SYS_IMPORT_TABLE_01" successfully
loaded/unloaded Starting "SCOTT"."SYS_IMPORT_TABLE_01": scott/***** tables=EMP,DEPT
directory=TEST_DIR dumpfile=EMP_DEPT.dmp logfile=impdpEMP_DEPT.log Processing
object type TABLE_EXPORT/TABLE/TABLE ORA-39151: Table "SCOTT"."DEPT" exists. All
dependent metadata and data will be skipped due to table_exists_action of skip ORA-
39151: Table "SCOTT"."EMP" exists. All dependent metadata and data will be skipped due to
table_exists_action of skip Processing object type
TABLE_EXPORT/TABLE/TABLE_DATA Processing object type
TABLE_EXPORT/TABLE/INDEX/INDEX Processing object type
TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT Processing object type
TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS Processing object type
TABLE_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT Processing object type
TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS Job "SCOTT"."SYS_IMPORT_TABLE_01"
completed with 2 error(s) at 17:02:23
```

5. **Datapump. Exportaciones / Importaciones de esquemas.** El parámetro OWNER de la exportación ha sido sustituido por el parámetro SCHEMAS el cual se utiliza para especificar los esquemas que se van a exportar. El siguiente es un ejemplo de la exportación e importación del esquema.

Exportación

```
SQL> host expdp scott/tiger schemas=SCOTT directory=TEST_DIR dumpfile=SCOTT.dmp
logfile=expdpSCOTT.log
```

```
Export: Release 10.2.0.1.0 - Production on Thursday, 11 June, 2009 17:03:55
Copyright (c) 2003, 2005, Oracle. All rights reserved.
Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options
Starting "SCOTT"."SYS_EXPORT_SCHEMA_01": scott/***** schemas=SCOTT
directory=TEST_DIR dumpfile=SCOTT.dmp logfile=expdpSCOTT.log
Estimate in progress using BLOCKS method...
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
Total estimation using BLOCKS method: 192 KB
Processing object type SCHEMA_EXPORT/PRE_SCHEMA/PROACT_SCHEMA
Processing object type SCHEMA_EXPORT/TABLE/TABLE
Processing object type SCHEMA_EXPORT/TABLE/INDEX/INDEX
Processing object type SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type SCHEMA_EXPORT/TABLE/COMMENT
Processing object type SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
Processing object type SCHEMA_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
```

```

.. exported "SCOTT"."DEPT"                5.656 KB    4 rows
.. exported "SCOTT"."EMP"                  7.820 KB   14 rows
.. exported "SCOTT"."SALGRADE"             5.585 KB    5 rows
.. exported "SCOTT"."BONUS"                 0 KB      0 rows
Master table "SCOTT"."SYS_EXPORT_SCHEMA_01" successfully loaded/unloaded

*****

Dump file set for SCOTT.SYS_EXPORT_SCHEMA_01 is:
/u01/app/oracle/oradata/SCOTT.dmp  Job "SCOTT"."SYS_EXPORT_SCHEMA_01" successfully
completed at 17:04:16

```

Importación

```

SQL> host impdp scott/tiger schemas=SCOTT directory=TEST_DIR dumpfile=SCOTT.dmp
logfile=impdpSCOTT.log

Import: Release 10.2.0.1.0 - Production on Thursday, 11 June, 2009 17:05:54
Copyright (c) 2003, 2005, Oracle. All rights reserved.
Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options
Master table "SCOTT"."SYS_IMPORT_SCHEMA_01" successfully loaded/unloaded
Starting "SCOTT"."SYS_IMPORT_SCHEMA_01": scott/***** schemas=SCOTT
directory=TEST_DIR dumpfile=SCOTT.dmp logfile=impdpSCOTT.log
Processing object type SCHEMA_EXPORT/PRE_SCHEMA/PROCACT_SCHEMA
Processing object type SCHEMA_EXPORT/TABLE/TABLE
ORA-39151: Table "SCOTT"."DEPT" exists. All dependent metadata and data will be skipped
due to table_exists_action of skip
ORA-39151: Table "SCOTT"."EMP" exists. All dependent metadata and data will be skipped
due to table_exists_action of skip
ORA-39151: Table "SCOTT"."BONUS" exists. All dependent metadata and data will be
skipped due to table_exists_action of skip
ORA-39151: Table "SCOTT"."SALGRADE" exists. All dependent metadata and data will be
skipped due to table_exists_action of skip
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
Processing object type SCHEMA_EXPORT/TABLE/INDEX/INDEX
Processing object type SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
Processing object type SCHEMA_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Job "SCOTT"."SYS_IMPORT_SCHEMA_01" completed with 4 error(s) at 17:05:58

```

Sugerencias Didácticas

Esta práctica puede realizarse en equipos para apoyarse y resolver dudas y posteriormente cada uno de los estudiantes debe realizarla de manera individual. No se recomienda aprenderse los pasos de memoria sino que se entienda que se hace en cada uno. El tiempo máximo para resolver esta práctica debe ser de una hora, por lo que se sugiere practicar continuamente. Trabajar directamente con sqlplus de Oracle, creando directamente sin ayuda de menús los estatutos correspondientes. Se recomienda generar los queries en editor de textos y ejecutarlos en sql, así como el respaldo de los mismos. Consultar siempre los manuales de clase para cualquier duda teórica o de sintaxis. Finalmente realizar el reporte del estudiante de manera individual y completo.

Reporte del estudiante

El estudiante entregará el reporte completo de la práctica desarrollada además de la información que se piden como complemento:

- 1. Realizar la práctica completa*
- 2. Definir los métodos para realizar los movimientos de datos de una base de datos*
- 3. Explicar cuáles son las diferencias entre dichos métodos*
- 4. Cuáles son los archivos que deben configurarse y comandos por ejecutar para cada método*

Bibliografía o Consultas sugeridas

Oracle Database 10g: Administration Workshop I

Oracle Database 10g: Administration Workshop II

<http://www.oracle.com/technology/software/products/database/index.html>

<http://www.oracle.com/technology/documentation/database10gr2.html>

OBSERVACIONES DEL DOCUMENTO!

1.OBJETIVO GENERAL DEL MANUAL

2. APARTADO DEFINA NO. NOMBRE Y OBJETIVOS ESPECIFICOS DE CADA PRACTICA

3. PRACT 1 . NO INTRODUCCION cambiar de nombre

4. CORRELACION CON TEMAS Y SUBTEMAS DEL PROGRAMA DE ESTUDIOS VIGENTE DE LA ASIGNATURA

5. AMPLIAR OBJETIVOS QUE CON QUE Y PARA QUE?

6. DESCRIBIR EL EQUIPO A UTILIZAR

7. DENTRO DE LA METODOLOGIA DONDE INICIA EL DESARROLLO DE LA PRACTICA AGREGAR SUBTITULO CORRESPONDIENTE

8. DENTRO DEL DESARROLLO PANTALLAS DESCRIPTIVAS, COMO EDITA LA DIRECCION ETC.

9. INCLUIR EN TODAS SUGERENCIAS DIDACTICAS EL REPORTE DEL ALUMNO

10. Y BIBLIOGRAFIAS

Sugerencias

Reporte Final

(Manual Completo)

Periodo Sabático

M.C. Martha Escamilla Zepeda

Febrero 2010

Martha Escamilla Zepeda