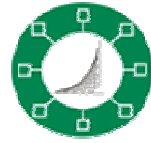


Laboratorio de Informática



MANUAL BÁSICO DE ORACLE

ESCUELA COLOMBIANA DE INGENIERÍA
JULIO GARAVITO
LABORATORIO DE INFORMÁTICA
BOGOTÁ D. C.
2007-2

TABLA DE CONTENIDO

INTRODUCCIÓN.....	3
1. TIPOS DE DATOS.....	4
1.1 TIPOS DE DATOS ALFANUMÉRICOS	4
1.2 TIPOS DE DATOS NUMÉRICOS	4
1.3 TIPOS DE DATOS FECHA.....	5
1.4 TIPO DE DATOS BINARIOS	5
1.5 OTROS TIPOS DE DATOS	5
2. ADMINISTRACIÓN BÁSICA DE ORACLE	6
2.1 CONCEPTO DE USUARIO, PRIVILEGIO Y ROL	6
2.1 CREACIÓN Y ELIMINACIÓN DE USUARIOS.....	6
2.2 CREACION DE ROLES.....	7
2.2.1 TIPOS DE ROLES EN UN USUARIO.....	7
2.3 PRIVILEGIOS DEL SISTEMA Y SOBRE OBJETOS.....	8
3. ALMACENAMIENTO EN ORACLE.....	10
3.1 TABLESPACE	10
3.2 DATAFILE.....	10
3.3 SEGMENT	11
3.4 EXTENT.....	11
3.5 DATA BLOCK	11
4. CATÁLOGO DE ORACLE	12

INTRODUCCIÓN

Una base de datos es un programa residente en memoria, que se encarga de gestionar todo el tratamiento de entrada, salida, protección y elaboración de la información de interés del usuario.

Tipos de bases de datos

Desde el punto de vista de la organización lógica:

- Jerárquicas. (Progress)
- Relacionales. (Oracle, Access, Sybase...)

Desde el punto de vista de número de usuarios:

- Monousuario (dBase, Access, Paradox...)
- Multiusuario cliente/servidor (Oracle, Sybase...)

Oracle es una base de datos relacional para entornos cliente/servidor.

En este manual se encuentra todo lo que es creación y eliminación de roles, así como de privilegios y de usuarios, conceptos de almacenamiento en Oracle, tipos de datos y el manejo del catálogo de Oracle.

Todo lo que es en cuanto a la creación de tablas y el manejo de estas, se realiza mediante comandos DDL y DML del lenguaje SQL.

1. TIPOS DE DATOS

Los tipos de datos soportados por Oracle se agrupan en los siguientes conjuntos.

- **Alfanuméricos**
 - CHAR
 - VARCHAR2
 - VARCHAR
 - NCHAR
 - NVARCHAR2
 - LONG
- **Numéricos**
 - NUMBER
 - FLOAT
- **Fecha**
 - DATE
- **Binarios**
 - RAW
 - LONG RAW
 - BLOB
 - CLOB
 - NLOB
 - BFILE
- **Otros**
 - ROWID

1.1 TIPOS DE DATOS ALFANUMÉRICOS

- **Tipo de dato CHAR(n):** Almacena cadenas de caracteres de longitud fija. Su rango está entre 1 y 2.000 bytes de ocupación.
- **Tipo de dato VARCHAR2(n):** Almacena cadenas de caracteres de longitud variable.
- **Tipo de dato VARCHAR(n):** En **Oracle8** es equivalente a VARCHAR2.
- **Tipo de dato NCHAR(n):** Almacena un valor alfanumérico de longitud fija. Puede almacenar caracteres ASCII, EBCDIC, UNICODE...
- **Tipo de dato NVARCHAR2(n):** Almacena un valor alfanumérico de longitud variable. Puede almacenar caracteres ASCII, EBCDIC, UNICODE...

1.2 TIPOS DE DATOS NUMÉRICOS

- **Tipo de dato NUMBER(p, s):** Almacena valores numéricos en punto flotante que pueden estar entre 1.0×10^{-130} y $9.9... (38 \text{ nueves}) \dots 9 \times 10^{125}$.

- **Tipo de dato FLOAT(N):** Almacena un número en punto decimal sin restricción de dígitos decimales, donde n indica la precisión binaria máxima que puede moverse en el rango 1 a 126.

1.3 TIPOS DE DATOS FECHA

- **Tipo de dato DATE:** Almacena un valor de fecha y hora. Para un tipo de dato DATE, Oracle almacena internamente los siguiente datos:
 - Siglo
 - Año
 - Mes
 - Día
 - Hora
 - Minuto
 - Segundo

El formato por defecto de las fechas es: 'DD-MON-YYYY'

1.4 TIPO DE DATOS BINARIOS

- **Tipos de datos binarios:** Permiten almacenar información en formato "crudo", valores binarios tal y como se almacenan en el disco duro o como residen en memoria.
- **Tipo de dato LONG:** Almacena caracteres de longitud variable hasta 2 Gb. En **Oracle8** y siguientes versiones se deben usar los tipos de datos CLOB y NLOB para almacenar grandes cantidades de datos alfanuméricos.

1.5 OTROS TIPOS DE DATOS

- **Tipo de dato ROWID:** Representa una dirección de la base de datos, ocupada por una única fila. Este tipo de dato sirve para guardar punteros a filas concretas.

2. ADMINISTRACIÓN BÁSICA DE ORACLE

2.1 CONCEPTO DE USUARIO, PRIVILEGIO Y ROL

Un usuario no es más que un conjunto de permisos que se aplican a una conexión de base de datos.

De igual manera, el usuario también tiene otras funciones:

- Ser el propietario de ciertos objetos.
- Definición del tablespace por defecto para los objetos de un usuario.
- Copias de seguridad.
- Cuotas de almacenamiento.

El usuario que posee privilegios está en la posibilidad de realizar dos operaciones:

- **Operación de sistema:** necesita el permiso de sistema correspondiente.
- **Operación sobre objeto:** necesita el permiso sobre el objeto en cuestión.

El rol en una base de datos es una agrupación de permisos de sistema y de objeto.

2.1 CREACIÓN Y ELIMINACIÓN DE USUARIOS

Antes de aprender como es la creación de usuarios es importante tener claro el concepto de **tablespace**.

Una base de datos se divide en unidades lógicas denominadas **TABLESPACES**.

Un **tablespace** es el nombre que tiene un conjunto de propiedades de almacenamiento que se aplican a los objetos (tablas, secuencias...) que se creen en la base de datos bajo el tablespace indicado (tablas, secuencias...).

Ahora si, la creación de usuarios se hace a través de la sentencia SQL CREATE USER, donde su sintaxis es:

```
CREATE USER nombre_usuario  
IDENTIFIED [ BY clave | EXTERNALLY ]  
{ DEFAULT TABLESPACE tablespace_por_defecto }
```

```
{ TEMPORARY TABLESPACE tablespace_temporal }  
{ DEFAULT ROLE [ roles, ALL [EXCEPT roles], NONE ] };
```

- **IDENTIFIED BY:** indicaa que tipo de autenticación se utilizará:
 - **Interna de Oracle:** una clave para cada usuario de base de datos.
 - **Interna del SO:** utilizando la seguridad del SO.
- **DEFAULT TABLESPACE** será el tablespace por defecto en la creación de objetos del usuario que estamos creando. Si se omite se utilizará el tablespace SYSTEM.
- **TEMPORARY TABLESPACE:** indica el tablespace que se utilizará para la creación de objetos temporales en la operaciones internas de Oracle. Si se omite se utilizará el tablespace SYSTEM.
- **DEFAULT ROLE:** asigna roles de permisos durante la creación del usuario.

La eliminación de usuarios se hace a través de la instrucción DROP USER, donde su sintaxis es:

```
DROP USER usuario {CASCADE};
```

- **CASCADE:** borra el usuario y todos los objetos que posee.

2.2 CREACION DE ROLES

La creación de roles permite asignar un grupo de permisos a un usuario, y poder modificar este grupo de permisos sin tener que ir modificando todos los usuarios.

Su sintaxis es:

```
CREATE ROLE nombre_rol  
{ [NOT IDENTIFIED | IDENTIFIED [BY clave | EXTERNALLY]] };
```

Una vez que el rol ha sido creado será necesario añadirle permisos a través de instrucción GRANT (se explica en privilegios del sistema).

2.2.1 TIPOS DE ROLES EN UN USUARIO

- **CONNECT:** Todos los permisos necesarios para iniciar sesión en Oracle.

- **RESOURCE:** Todos los permisos necesarios para tener recursos para la creación de objetos.
- **DBA:** Todos los permisos para un administrador de bases de datos (DBA).
- **EXP_FULL_DATABASE:** Permisos para poder exportar toda la base de datos.
- **IMP_FULL_DATABASE:** Permisos para poder importar toda la base de datos.

Un usuario básico debe tener por lo menos dos permisos:

- **CONNECT**
- **RESOURCE**

2.3 PRIVILEGIOS DEL SISTEMA Y SOBRE OBJETOS

Los privilegios de sistema son permisos para realizar ciertas operaciones en la base de datos, donde para poder asignarlos se usa la instrucción GRANT y para cancelarlos REVOKE.

Instrucción GRANT:

```
GRANT [privilegios_de_sistema | roles] TO [usuarios | roles |PUBLIC]
{ WITH ADMIN OPTION };
```

- **WITH ADMIN OPTION** permite que el privilegio/rol que se ha concedido, pueda ser concedido a otros usuarios por el usuario al que se le está asignando.

Instrucción REVOKE:

```
REVOKE [privilegios_de_sistema | roles] FROM [usuarios | roles
|PUBLIC];
```

Los privilegios sobre objetos consienten que un objeto (creado por un usuario) pueda ser accedido por otros usuarios. Y el nivel de acceso depende del permiso que puede ser de SELECT, de UPDATE, de DELETE, de INSERT o de todos ellos.

La sintaxis es:

```
GRANT [ALL {PRIVILEGES} | SELECT | INSERT | UPDATE | DELETE]
```



```
ON objeto  
TO [usuario | rol | PUBLIC]  
{WITH ADMIN OPTION};
```

De igual manera para eliminar privilegios sobre objetos se tiene la instrucción REVOKE:

Si sintaxis es:

```
REVOKE [ALL {PRIVILEGES} | SELECT | INSERT | UPDATE | DELETE]  
ON objeto  
FROM [usuario | rol | PUBLIC]  
{WITH ADMIN OPTION};
```

3. ALMACENAMIENTO EN ORACLE

En el almacenamiento en Oracle se manejan cinco conceptos básicos:

- TABLESPACE
- DATAFILE
- SEGMENT
- EXTEND
- DATA BLOCK

3.1 TABLESPACE

El concepto básico de Tablespace fue explicado brevemente en la sección de creación y eliminación de usuarios. Pero cave añadir que:

Las propiedades que se asocian a un tablespace son y cuando un objeto es creado dentro de un tablespace, hereda todas estas propiedades:

- Localización de los ficheros de datos.
- Especificación de máximas cuotas de consumo de disco.
- Control de la disponibilidad de los datos (en línea o fuera de línea).
- Backup de datos.

3.2 DATAFILE

Un datafile son los "ficheros de datos" donde se almacena la información físicamente, este puede tener cualquier nombre y extensión (siempre dentro de las limitaciones del SO), y puede estar localizado en cualquier directorio del disco duro, además está asociado a un solo tablespace y un tablespace está asociado a uno o varios datafiles

Los datafiles tienen una propiedad llamada **AUTOEXTEND**, que en dado caso de que está activa, esta se encarga de que el datafile crezca automáticamente) cada vez que se necesite espacio y no exista.

Aunque el datafile este vacío, este tiene el tamaño en disco que ha sido indicado en su creación o el que el sistema le ha dado por defecto. Oracle hace esto para direccionar espacio continuo en disco y evitar así la fragmentación. Conforme se vayan creando objetos en ese tablespace, se irá ocupando el espacio direccionado.

3.3 SEGMENT

Un segment es aquel espacio direccionado por la base de datos dentro de un datafile para ser utilizado por un solo objeto. Así una tabla (o cualquier otro objeto) está dentro de su segment, y nunca podrá salir de él, ya que si la tabla crece, el segment también crece.

En pocas palabras, el segment es la representación física del objeto en base de datos

Existen tres tipos de segments (principalmente):

- **Segmentos de tipo TABLE:** son aquellos que contienen tablas
- **Segmentos de tipo INDEX:** son aquellos que contienen índices
- **Segmentos de tipo ROLLBACK:** son aquellos se usan para almacenar información de la transacción activa.

3.4 EXTENT

Extent es un espacio de disco que se direcciona de una sola vez, es decir un segment que se direcciona en un momento determinado de tiempo.

El concepto de extent es un concepto físico, unos extent están separados de otros dentro del disco.

Hay dos tipos de extent:

- **INITIAL** (extensiones iniciales): estas son las extensiones que se direccionan en el momento de la creación del objeto. Una vez que un objeto está creado, no se puede modificar su extensión inicial.
- **NEXT** (siguientes o subsiguientes extensiones): toda extensión direccionada después de la creación del objeto.

3.5 DATA BLOCK

El concepto de Data block es un concepto físico, ya que representa la mínima unidad de almacenamiento que es capaz de manejar Oracle, además es el último eslabón en la cadena de almacenamiento

En un disco duro no es posible que un fichero pequeño ocupe menos de lo que indique la unidad de asignación, así si la unidad de asignación es de 4 Kb, un fichero que ocupe 1 Kb, en realidad ocupa 4 Kb.

4. CATÁLOGO DE ORACLE

Oracle cuenta con una serie de tablas y vistas que conforman una estructura denominada catálogo.

La principal función del catálogo de Oracle es almacenar toda la información de la estructura lógica y física de la base de datos, desde los objetos existentes, la situación de los datafiles, la configuración de los usuarios, etc.

El catálogo sigue un estándar de nomenclatura para que su memorización sea más fácil y son una serie de prefijos, que son:

- **DBA_** Objetos con información de administrador. Sólo accesibles por usuarios DBA
- **USER_** Objetos con información propia del usuario al que se está conectando. Accesible desde todos los usuarios. Proporcionan menos información que los objetos DBA_
- **ALL_** Objetos con información de todos los objetos en base de datos.
- **V_ \$ ó V\$** Tablas virtuales

Para acceder a los elementos del catálogo, se utiliza el respectivo prefijo que se necesite seguido del nombre del objeto en plural, por ejemplo: **DBA_TABLES** es la información para administradores de las tablas en base de datos.

Ciertos datos del catálogo de Oracle están continuamente actualizados, como por ejemplo las columnas de una tabla. Sin embargo hay otros datos que no pueden actualizarse en tiempo real ya que como consecuencia penalizaría demasiado el rendimiento general de la base de datos.

Para llevar acabo esa actualización del catálogo con ese tipo de datos es necesario ejecutar una sentencia especial que se encargue de volcar la información recopilada al catálogo:

Esta sentencia es:

```
ANALYZE [TABLE|INDEX] nombre  
[COMPUTE|ESTIMATE|DELETE] STATISTICS;
```

- **COMPUTE** hace un cálculo exacto de la estadísticas (tarda más en realizarse en ANALYZE).
- **ESTIMATE** hace una estimación partiendo del anterior valor calculado y de un posible factor de variación y la cláusula
- **DELETE** borra las anteriores estadísticas.