

# **Manejo de Usuarios**

**Altas, bajas, limitación de recursos**

# Objetivos

- **Creación de nuevos usuarios**
- **Modificar características**
- **Eliminar usuarios**

# Usuarios y Seguridad



# Control de Seguridad

**Se trata de establecer los límites de actuación de un usuario a través de los siguientes parámetros:**

- **Identificación: S.O. y Base de Datos**
- **Cuotas de tablespace: Limitan el espacio físico**
- **Tablespace por defecto: Localización de sus datos**
- **Tablespace temporal: Ordenación y producto**
- **Bloqueo de cuenta: Evitar la conexión, si es necesario**
- **Limitación de recursos: CPU, num. de I/O, num. sesiones,...**
- **Privilegios directos: Controlar acciones de usuario concreto**
- **Privilegios de role: Indirectos. Grupos de usuarios.**

# Creación de usuarios

1. Elegir username y mecanismo de identificación
2. Establecer tablespace para almacenamiento.
3. Decidir las cuotas por tablespace.
4. Asignar tablespace temporal. System por defecto. Conviene especificar.
5. Crear el usuario.
6. Concederle privilegios y roles.

# Sintaxis para crear usuarios

```
CREATE USER <nombre>  
IDENTIFIED BY {password|EXTERNALLY}  
[DEFAULT TABLESPACE <tb-id>]  
[TEMPORARY TABLESPACE <tb-id>]  
[QUOTA {<int> M|UNLIMITED} ON <tb-id>]  
[PASSWORD EXPIRE] // Fuerza a cambiarla  
[ACCOUNT {LOCK|UNLOCK}]  
[PROFILE {<profile>|DEFAULT}] ;
```

# Ejemplo de Creación de Usuario

```
CREATE USER userp  
IDENTIFIED BY myUserP  
DEFAULT TABLESPACE users  
TEMPORARY TABLESPACE temp  
QUOTA 15M ON users  
PASSWORD EXPIRE;
```

# Identificación desde el S.O.

## Parámetro de Inicialización: OS\_AUTHENT\_PREFIX

OS_AUTHENT_PREFIX	Username	Remote Login Possible
OS_	OS_USERP	No
Cadena vacía o “ ”	USERP	No
OPS\$ (por defecto)	OPS\$USERP (por defecto)	Sí



# Identificación desde S.O. para el usuario “userp”

- **OS\_** : Sistema operativo del servidor.  
Usuario: OS\_userp
- **“ “** : Usuario: userp (no se distinguen)
- **ops\$**: Prefijo por defecto. Permite acceso remoto. Usuario: ops\$userp

**CREATE USER ops\$userp IDENTIFIED BY  
myUserP**

Obliga a identificación sólo desde  
clientes.

# Alteración de Características de Usuario

```
ALTER USER userp  
IDENTIFIED BY myUserP2  
PASSWORD EXPIRE  
ACCOUNT UNLOCK;
```

# Alteración de la cuota de Tablespace

```
ALTER USER userp  
QUOTA 0M ON users;
```

Una cuota de 0M, o inferior a la ya ocupada, impide ocupar más memoria de la actual, pero no elimina ni altera los objetos ya creados.

# Eliminación de Usuarios

```
DROP USER userp;
```

**Para eliminar además todos los objetos de su esquema, se usa:**

```
DROP USER userp CASCADE;
```

# Control de Usuarios

## DBA\_USERS

USERNAME

USER\_ID

CREATED

ACCOUNT\_STATUS

(open/close)

LOCK\_DATE

EXPIRY\_DATE

DEFAULT\_TABLESPACE

TEMPORARY\_TABLESPACE

## DBA\_TS\_QUOTAS

USERNAME

TABLESPACE\_NAME

BYTES

MAX\_BYTES

BLOCKS

MAX\_BLOCKS

-1= Ilimitado

# Ejemplo de consulta al catálogo

```
SELECT tablespace_name,blocks,max_blocks,  
       bytes,max_bytes,account_status  
FROM   dba_ts_quotas  
WHERE  username='PEDRO' ;
```

TABLESPACE_NAME	BLOCKS	MAX_BLOCKS	BYTES	MAX_BYTES
DATA01	10	-1	20480	-1

1 row selected

# Ejercicios

1. **Crear el usuario Bob con password ALONG, asegurando que no utilice espacio en SYSTEM y que no sobrepase 1M en el tablespace USERS. Dejar que se conecte.**
2. **Crear el usuario Kay con password Mary asegurando que los objetos y el espacio temporal necesarios no sean de SYSTEM. Asignar cuota ilimitada en el tb de datos.**
3. **Copiar la tabla EMP del usuario SCOTT en la cuenta de Kay.**
4. **Mostar la información sobre Bob y Kay y sobre sus límites de espacio en los tablespaces correspondientes.**

# **Perfiles de Usuario**

**Administración de recursos y limitación  
de los mismos al usuario**



# Objetivos

- **Creación y asignación de perfiles de usuario**
- **Control del uso de recursos a través de perfiles**
- **Modificación y eliminación de perfiles**
- **Control de passwords usando perfiles**
- **Obtención de información sobre perfiles, recursos y passwords**

# Perfiles (Profiles)

- Se trata de dar nombre a una serie de limitaciones en los recursos y uso de passwords.
- La sentencia CREATE/ALTER USER asigna/altera una relación usuario-perfil
- Pueden habilitarse y deshabilitarse
- Existe un perfil por defecto DEFAULT con recursos ilimitados
- Se pueden limitar los recursos a varios niveles

# Recursos Administrados

Tiempo de CPU	Operaciones de I/O
Tiempo muerto	Tiempo de conexión
Memoria	Sesiones concurrentes
Tiempo de validez de la password	Complejidad de la password
Bloqueo de cuenta	

# Creación de un Perfil: Limitando los Recursos

```
CREATE PROFILE developer_prof LIMIT  
SESSIONS_PER_USER 2  
CPU_PER_SESSION 10000  
IDLE_TIME 60  
CONNECT_TIME 480;
```

**No pueden asignarse perfiles  
a otros perfiles**

# Establecimiento de límites a nivel de sesión

Recurso	Descripción
CPU_PER_SESSION	Tiempo total de CPU en décimas de segundos
SESSIONS_PER_USER	Número de sesiones concurrentes permitidas
CONNECT_TIME	Tiempo de conexión medido en minutos
IDLE_TIME	Periodo de tiempo inactivo medido en minutos
LOGICAL_READS_PER_SESSION	Número de bloques de datos físicos leídos en la sesión.
PRIVATE_SGA	Espacio privado en el SGA medido en bytes. Solo en modo distribuido.

# Limitaciones a nivel de llamadas

Recurso	Descripción
CPU_PER_CALL	Tiempo de CPU por llamada en décimas de segundos
LOGICAL_READS_PER_CALL	Número de bloques físicos leídos por llamada

# Asignación de Profiles a un Usuario

```
CREATE USER user3 IDENTIFIED BY user3  
DEFAULT TABLESPACE data01  
TEMPORARY TABLESPACE temp  
QUOTA unlimited ON data01  
PROFILE developer_prof;
```

```
ALTER USER user3  
PROFILE developer_prof;
```

- Un usuario sólo puede tener asignado un perfil
- La asignación de perfiles no afecta a las sesiones en curso

# **Violación de los límites**

## **1) A NIVEL DE SESION:**

- \* Mensaje de error**
- \* Se desconecta al usuario**

## **2) A NIVEL DE LLAMADA:**

- \* Se aborta la ejecución de la sentencia**
- \* Se deshacen los cambios realizados**
- \* El usuario permanece conectado**



# Habilitación de perfiles

Se puede realizar de dos formas:

- 1) Poniendo el parámetro de inicialización **RESOURCE\_LIMIT = TRUE** (FALSE por defecto)
- 2) Habilitarlo desde el cliente con el comando **ALTER SYSTEM.**

```
ALTER SYSTEM SET RESOURCE_LIMIT=TRUE;
```

# Modificación de un Perfil

```
ALTER PROFILE default LIMIT  
SESSIONS_PER_USER 5  
CPU_PER_CALL 3600  
IDLE_TIME 30;
```

# Eliminación de un Perfil

```
DROP PROFILE developer_prof;
```

```
DROP PROFILE developer_prof  
CASCADE;
```

- **CASCADE** quita el perfil a los usuarios que lo tuvieran asignado, que pasan a tener **DEFAULT**
- El perfil por defecto no puede eliminarse.

# Información sobre Perfiles

## DBA\_USERS

- profile
- username

## DBA\_PROFILES

- profile
- resource\_name
- resource\_type  
(KERNEL)
- limit

```
SELECT p.profile, p.resource_name, p.limit
FROM dba_users u, dba_profiles p
WHERE p.profile=u.profile AND username='user3'
      AND p.resource_type='KERNEL' ;
```

# Control de Passwords







# Control de Passwords

- Se realiza aunque `RESOURCE_LIMIT=FALSE`
- Bloqueo de cuenta cuando falla la password un número de veces.
- Tiempo de vida de la password.
- Historia de la password: Comprueba que no se está reutilizando después de haber transcurrido su tiempo de vida.
- Verificación de complejidad, para evitar intrusos.

# Creación de un Perfil para Control de Passwords




```
CREATE PROFILE grace_5 LIMIT  
  FAILED_LOGIN_ATTEMPTS 3  
  PASSWORD_LIFE_TIME 30  
  PASSWORD_REUSE_TIME 30  
  PASSWORD_VERIFY_FUNCTION verify_function  
  PASSWORD_GRACE_TIME 5;
```

# Control de Passwords

	Parametro	Descripción
	FAILED_LOGIN_ATTEMPTS	Número de intentos fallidos antes de ser bloqueada la cuenta
	PASSWORD_LOCK_TIME	Número de días que se bloquea la cuenta tras expirar la password. Debe obtenerse otra.
	PASSWORD_LIFE_TIME	Vida de la password en días.
	PASSWORD_GRACE_TIME	Días que se conceden para cambiar la password después de que ésta finalice su periodo de vida máx.



# Control de Passwords

	Parametro	Descripción
	PASSWORD_REUSE_TIME	Días que deben pasar antes de reutilizar la password.
	PASSWORD_REUSE_MAX	Máximo número de veces que puede usarse una password.
	PASSWORD_VERIFY_FUNCTION	Función PL/SQL que verifica la complejidad de una password antes de admitirla como buena.

# Función de Verificación de Passwords: VERIFY\_FUNCTION



- Longitud mínima 4 caracteres
- La password debe ser diferente al nombre de usuario
- La password debe tener, al menos, una letra, un número y un carácter especial
- Una password debe diferenciarse de la anterior en, al menos, tres caracteres
- SYS puede programar la función a su gusto.

# Información sobre Passwords

## DBA\_USERS

- profile
- username
- password
- account\_status
- lock\_date
- expiry\_date

## DBA\_PROFILES

- profile
- resource\_name
- resource\_type  
(PASSWORD)
- limit

```
SELECT username, password, account_status,  
expiry_date FROM dba_users;
```

# Ejercicios

1. Crear un perfil “nuevo” que permita dos sesiones concurrentes por usuario y un máx. de un minuto de inactividad. Asignárselo a Bob.
2. Conectarse como Bob más de dos veces.
3. Asignar los siguientes límites al perfil default
  - a) Bloquear la cuenta tras dos intentos fallidos
  - b) La password expira a los 30 días
  - c) La password tiene un periodo de gracia de 5 días para ser cambiada.

Comprobar resultados.

4. Alterar el perfil por defecto para que la password no expire nunca

# **Administración de Privilegios**

**Control de la actuación del usuario:  
limitación de su operatividad**

# Administración de Privilegios

Existen dos tipos de privilegios:

- **Del SISTEMA:** Permiten al usuario realizar acciones particulares sobre la BD
- **De OBJETO:** Permiten al usuario acceder/manipular un objeto específico (tabla, vista, función,...)

# Privilegios de Sistema

- Existen alrededor de 80 privilegios de este tipo.
- La palabra ANY en el nombre de un privilegio, significa que se tiene dicho privilegio en cualquier esquema.
- La sentencia GRANT concede privilegios a usuarios o grupos.
- La sentencia REVOKE los retira.

# Algunos Privilegios de Sistema

Categoría	Ejemplos
INDICES	<ul style="list-style-type: none"><li>• CREATE ANY INDEX</li><li>• ALTER ANY INDEX</li><li>• DROP ANY INDEX</li></ul>
TABLAS	<ul style="list-style-type: none"><li>• CREATE/DROP TABLE</li><li>• CREATE ANY TABLE</li><li>• ALTER ANY TABLE</li><li>• DROP ANY TABLE</li><li>• SELECT ANY TABLE</li><li>• UPDATE ANY TABLE</li><li>• DELETE ANY TABLE</li></ul>
SESIONES	<ul style="list-style-type: none"><li>• CREATE SESSION</li><li>• ALTER SESSION</li></ul>
TABLESPACES	<ul style="list-style-type: none"><li>• CREATE TABLESPACE</li><li>• ALTER TABLESPACE</li><li>• DROP TABLESPACE</li></ul>



# Concesión de Privilegios de Sistema

```
GRANT {system_priv|system_role,...}  
TO {user|role|public,...}  
[WITH ADMIN OPTION];
```

```
GRANT CREATE TABLE, SELECT ANY  
TABLE TO user1;
```

```
GRANT CREATE ANY INDEX TO scott  
WITH ADMIN OPTION;
```

# Privilegios Especiales: SYSDBA y SYSOPER

Categoría	Contenido
SYSOPER	<ul style="list-style-type: none"><li>• STARTUP</li><li>• SHUTDOWN</li><li>• ALTER DATABASE OPEN   MOUNT</li><li>• ALTER DATABASE BACKUP CONTROLFILE</li><li>• ALTER TABLESPACE BEGIN/END BACKUP</li><li>• RECOVER DATABASE,</li><li>• ALTER DATABASE ARCHIVELOG</li></ul>
SYSDBA	<ul style="list-style-type: none"><li>• SYSOPER privileges WITH ADMIN OPTION</li><li>• CREATE DATABASE</li><li>• RECOVER DATABASE</li></ul>

# Información sobre Privilegios del Sistema

## DBA\_SYS\_PRIVS

- GRANTEE
- PRIVILEGE
- ADMIN OPTION

```
SELECT * FROM DBA_SYS_PRIVS;
```

GRANTEE	PRIVILEGE	ADM
-----		
PETER	CREATE SESSION	NO
USER1	CREATE SESSION	NO
USER2	CREATE SESSION	NO
USER1	CREATE TABLE	YES
USER2	CREATE PROCEDURE	NO

# Derogación de Privilegios de Sistema

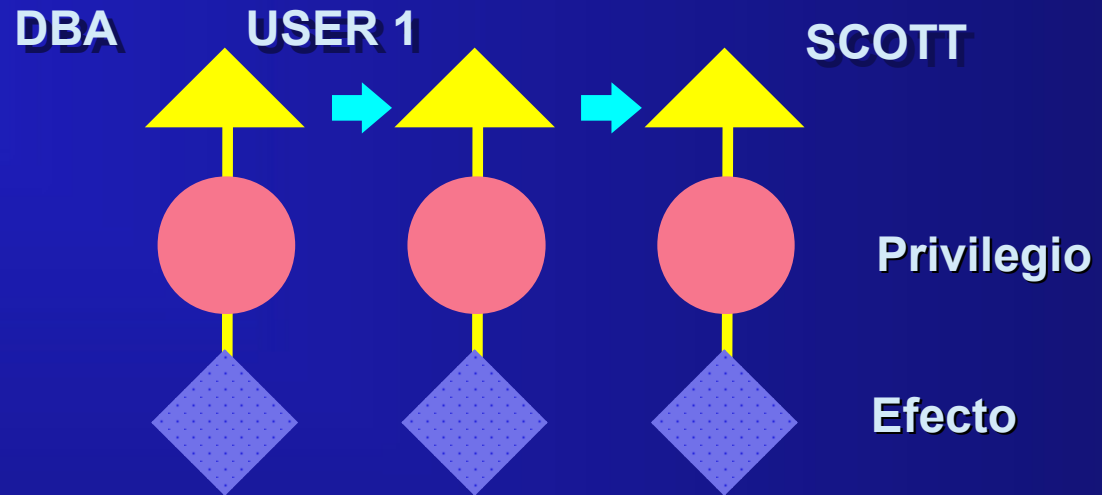
```
REVOKE {system_priv|system_role,...}  
FROM {user|role|public,...};
```

```
REVOKE CREATE TABLE FROM user1;
```

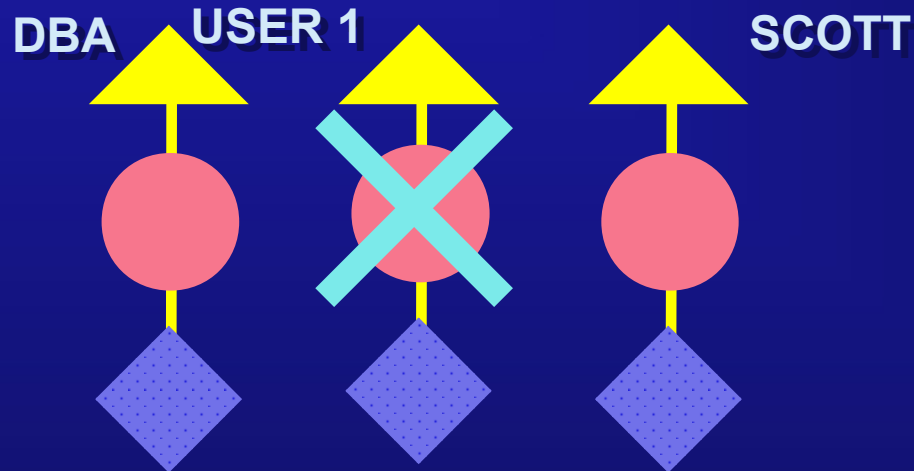
```
REVOKE CREATE SESSION FROM scott;
```

# Derogación de Privilegios con ADMIN OPTION

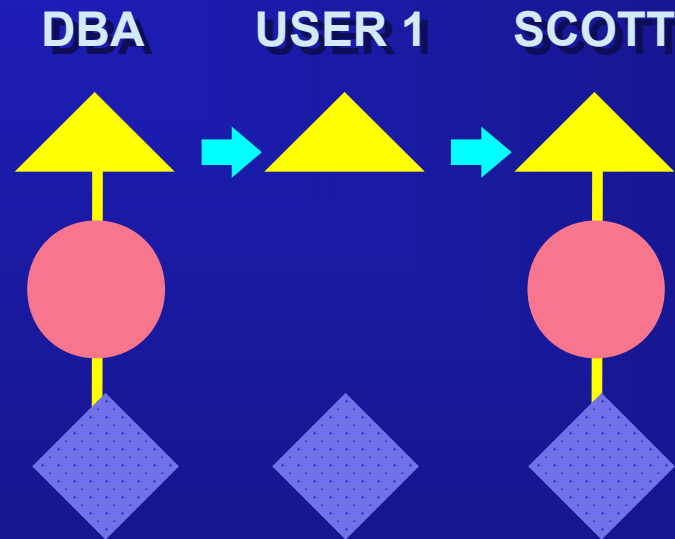
**GRANT**



**REVOKE**



# Derogación de Privilegios con ADMIN OPTION



**RESULTADO: No hay efecto en cascada**

# Privilegios de Objeto

Privilegio	Tabla	Vista	Secuencia	Procedure
ALTER	✓		✓	
DELETE	✓	✓		
EXECUTE				✓
INDEX	✓			
INSERT	✓	✓		
REFERENCES	✓			
SELECT	✓	✓	✓	
UPDATE	✓	✓		

# Concesión de Privilegios de Objeto

```
GRANT {obj_priv [(column_list)],...  
      |ALL PRIVILEGES}  
ON <objeto>  
TO {user|role|public,...}  
[WITH GRANT OPTION];
```

```
GRANT UPDATE(ename,sal) ON emp  
TO user1 WITH GRANT OPTION;
```



# Información sobre Privilegios de Objeto

## **DBA\_TAB\_PRIVS**

**GRANTEE  
OWNER  
TABLE\_NAME  
GRANTOR  
PRIVILEGE  
GRANTABLE**

## **DBA\_COL\_PRIVS**

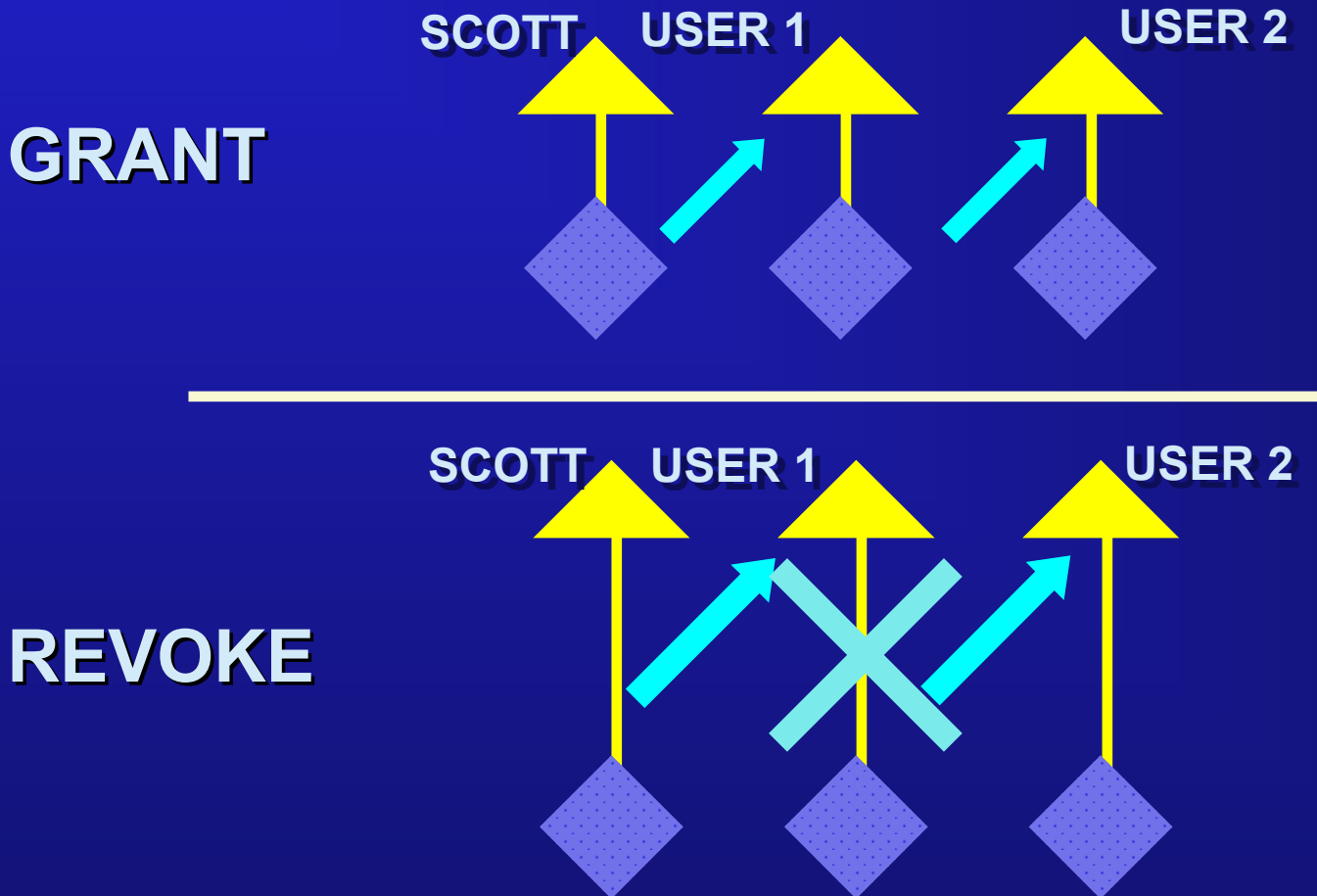
**GRANTEE  
OWNER  
TABLE\_NAME  
COLUMN\_NAME  
GRANTOR  
PRIVILEGE  
GRANTABLE**

# Derogación de Privilegios de Objeto

```
REVOKE {obj_priv,..|ALL PRIVILEGES}  
ON <objeto>  
FROM {user|role|public,...}  
[CASCADE CONSTRAINTS];
```

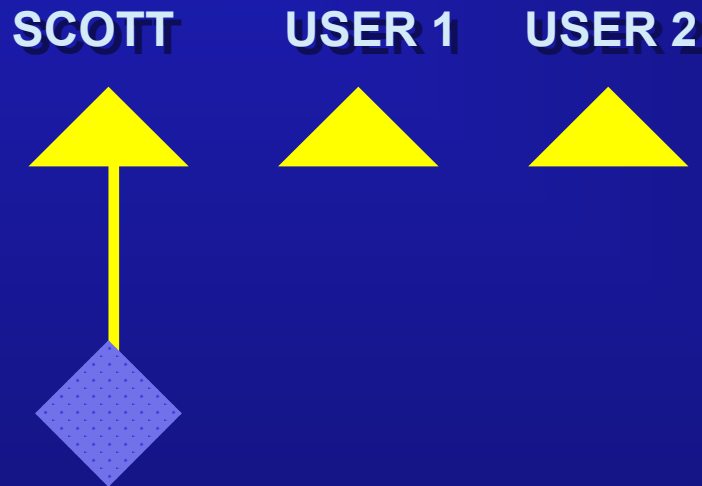
- **CASCADE CONSTRAINTS** elimina la integridad referencial definida mediante los privilegios REFERENCES o ALL.

# Derogación de Privilegios con GRANT OPTION



# Derogación de Privilegios con GRANT OPTION

**RESULTADO:**



# Ejercicios

1. Permitir a kay conectarse a la BD y crear tablas propias
2. Conectar como kay y crear la tabla DEPT (ejecutar script ulcase1.sql)
3. Conectar como sys y rellenar las tablas de kay con las de scott.EMP y scott.DEPT
4. Conceder a Bob (como sys) el privilegio de consultar la tabla EMP de Kay. Hacerlo como Kay y conceder grant option.
5. Consultar los cambios en el catálogo

## Ejercicios (continuación)

6. Crear el usuario Todd con capacidad de conexión.
7. Conectar como Bob y permitir a Todd acceder a la tabla EMP de Kay.
8. Conectar como Kay y quitarle el privilegio a Bob de consultar su tabla EMP.
9. Conectar como Todd y consultar la tabla EMP de Kay...

# **Manipulación de Roles**

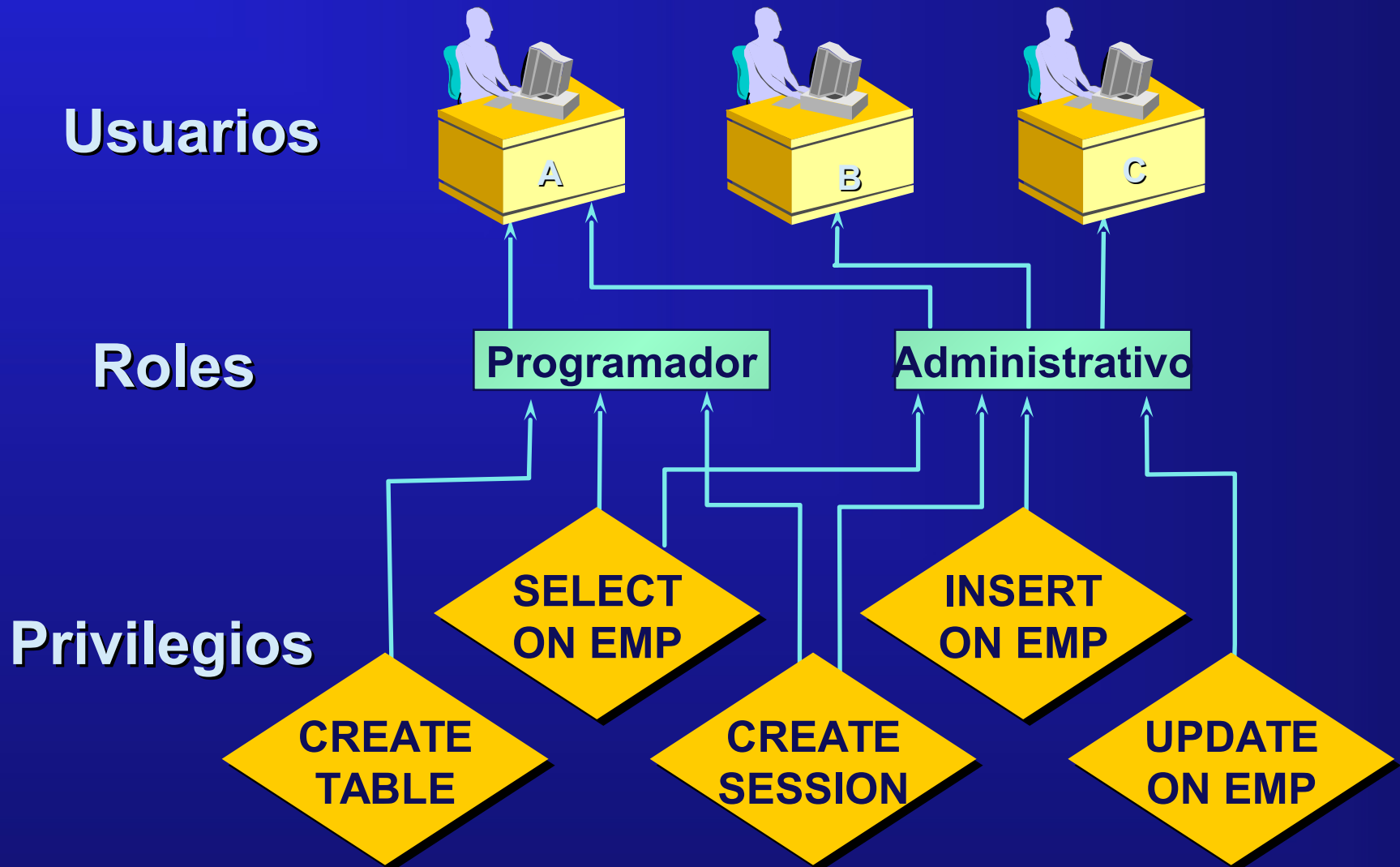
**Agrupación de privilegios bajo un nombre**

# **Puntos a tratar:**

- **Creación y modificación de roles**
- **Accesibilidad a los roles**
- **Eliminación de roles**
- **Uso de roles predefinidos**
- **Obtención de información sobre roles a través del diccionario de datos**



# Roles



# Beneficios de los Roles

- Facilitan la concesión de grupos de privilegios
- Manipulación dinámica de privilegios
- Se pueden (in)habilitar temporalmente
- Reduce la cantidad de información a almacenar en el catálogo

# Creación de Roles

```
CREATE ROLE administrativo;
```

```
CREATE ROLE programador;
```

# Uso de Roles Predefinidos

Nombre de Rol	Descripción
CONNECT	Consulta a tablas públicas
RESOURCE	Crear tablas e índices
DBA	Todos los privilegios
EXP_FULL_DATABASE	Exportar la DB
IMP_FULL_DATABASE	Importar la DB
DELETE_CATALOG_ROLE	Privilegio de borrado sobre el catálogo
EXECUTE_CATALOG_ROLE	Privilegio de ejecutar paquetes del catálogo
SELECT_CATALOG_ROLE	Privilegio de SELECT sobre todo el catálogo

# Asignación de Roles

```
GRANT administrativo TO scott;
```

```
GRANT programador TO opc;
```

```
GRANT RESOURCE TO alumno1  
WITH ADMIN OPTION;
```

**WITH ADMIN OPTION** permite al usuario que recibe el role, concederlo a otros usuarios

# Asignación de Roles por Defecto

```
ALTER USER scott  
DEFAULT ROLE programador;
```

```
ALTER USER opc DEFAULT ROLE ALL;
```

```
ALTER USER alumno1 DEFAULT ROLE ALL  
EXCEPT DBA;
```

```
ALTER USER scott DEFAULT ROLE NONE;
```

# Derogación de Roles

```
REVOKE programador FROM scott;
```

```
REVOKE administrativo FROM  
PUBLIC;
```

# Eliminación de Roles

```
DROP ROLE jefe_ventas;
```



# Información sobre Roles

Vista	Descripción
DBA_ROLES	Todos los roles que existen
DBA_ROLE_PRIVS	Roles asignados a usuarios y a roles
ROLE_ROLE_PRIVS	Roles asignados a roles
DBA_SYS_PRIVS	Privilegios del sistema asignados usuarios y roles
ROLE_SYS_PRIVS	Privilegios del sistema asignados a roles
ROLE_TAB_PRIVS	Privilegios de tablas asignados a roles

# Ejercicios

1. Listar todos los privilegios que tiene el role RESOURCE
2. Crear el role DEV para crear tablas, crear vistas y consultar la tabla EMP de Kay.
3. Conceder a Bob los roles DEV y RESOURCE, pero habilitarle sólo RESOURCE cuando se conecte.
4. Conceder a Bob el role que le permite consultar todo el catálogo. Comprobar alcance.