

Universidad Tecnológica de Panamá

Centro Regional De Veraguas



Facultad de Ingeniería de Sistemas
Computacionales

Curso: Base de Datos II

Profesor: Abdiel Kapell

TAREA 7

Estudiante
Arland Barrera

2024

1	Desarrollo	3
1.1	Limitar las cuotas de espacios de tablas	3
1.2	Administrar los perfiles de límite de recursos	4
1.3	Respaldo y recuperación de base de datos	6
1.3.1	Exportación de objetos y datos	6
1.3.2	Importación de objetos y datos	7

Desarrollo

1.1 Limitar las cuotas de espacios de tablas

En muchas ocasiones resulta necesario limitar el espacio de almacenamiento en un servidor atendiendo a diversos criterios, por ejemplo la limitación del espacio disponible para correo electrónico o para albergue de páginas web.

Para limitar las cuotas de espacios de tablas en una base de datos, se pueden utilizar varias estrategias dependiendo del sistema de gestión de bases de datos (SGBD) utilizado. A continuación, se presentan algunas opciones para MySQL y SQL Server:

MySQL

1. **Asignación de cuotas a una base de datos existente:** Utilizar la sentencia **CREATE TABLESPACE** con la opción **QUOTA** para establecer un límite de espacio para una base de datos específica. Por ejemplo:

```
1 CREATE TABLESPACE mydb QUOTA 100M;
```

Esto establece un límite de 100 MB para la base de datos *mydb*.

2. **Asignación de cuotas a un usuario:** Utilizar la sentencia **CREATE USER** con la opción **QUOTA** para establecer un límite de espacio para un usuario específico. Por ejemplo:

```
1 CREATE USER myuser IDENTIFIED BY 'mypassword' QUOTA 50
   M ON mydb;
```

Esto establece un límite de 50 MB para el usuario *myuser* en la base de datos *mydb*.

SQL SERVER

1. **Asignación de cuotas a un usuario:** Utilizar la sentencia **CREATE USER** con la opción **QUOTA** para establecer un límite de espacio para un usuario específico. Por ejemplo:

```
1 CREATE USER myuser FOR LOGIN = 'mylogin'
   DEFAULT_TABLESPACE = 'mydb' QUOTA 100MB;
```

Esto establece un límite de 100 MB para el usuario *myuser* en la base de datos *mydb*.

2. **Configuración de cuotas en el nivel de instancia:** En la configuración de la instancia de SQL Server, se puede establecer un límite de espacio global para todas las bases de datos. Para hacer esto, se necesita el permiso **ALTER INSTANCE** y se utiliza la sentencia **ALTER INSTANCE** con la opción **SET MAX_DATA_SIZE**. Por ejemplo:

```
1 ALTER INSTANCE SET MAX_DATA_SIZE = 500GB;
```

Esto establece un límite de 500 GB para el tamaño total de todas las bases de datos en la instancia de SQL Server.

1.2 Administrar los perfiles de límite de recursos

Administrar los perfiles de límite de recursos en una base de datos es una buena práctica para asegurar un uso eficiente y controlado de los recursos por parte de los usuarios. Esto es especialmente útil en bases de datos *Oracle*, en las cuales se pueden definir perfiles que restringen ciertos recursos como el uso de CPU, la cantidad de sesiones abiertas, el tiempo de inactividad, y mucho más.

1. **¿Qué es un perfil de límite de recursos?**

Un **perfil de base de datos** permite definir límites sobre el uso de recursos del sistema para los usuarios. Se puede asignar un perfil a un usuario y así controlar:

- Uso de CPU y tiempo de sesión.
- Conexiones concurrentes.
- Políticas de contraseñas y bloqueo de cuentas.

2. **Activar la administración de recursos en Oracle**

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

3. Crear un perfil personalizado

El siguiente ejemplo crea un perfil llamado *mi_perfil* que establece límites en el uso de recursos:

```
1 CREATE PROFILE mi_perfil LIMIT
2 SESSIONS_PER_USER          2                -- Maximo
   2 sesiones por usuario
3 CPU_PER_SESSION            10000             -- Maximo
   10,000 unidades de CPU por sesion
4 CPU_PER_CALL                1000             -- Maximo
   1,000 unidades de CPU por llamada
5 CONNECT_TIME                60               -- Maximo
   60 minutos de conexion
6 IDLE_TIME                   10               -- Tiempo
   inactivo de 10 minutos antes de desconectar
7 FAILED_LOGIN_ATTEMPTS       5               --
   Bloqueo despurs de 5 intentos fallidos
8 PASSWORD_LIFE_TIME          30               -- La
   contraseña expira en 30 ídas
9 PASSWORD_REUSE_TIME         365              --
   Reutilizacion de contraseña despues de 1 ano
10 PASSWORD_REUSE_MAX          5               --
   Reutilizar contraseña despues de 5 cambios;
```

4. Asignar el perfil a un usuario

```
1 ALTER USER usuario1 PROFILE mi_perfil;
```

Esto significa que el *usuario1* estará sujeto a las restricciones definidas en *mi_perfil*.

5. Modificar un perfil existente

```
1 ALTER PROFILE mi_perfil LIMIT
2 IDLE_TIME 15;    -- Aumentar el tiempo inactivo a 15
   minutos
```

6. Eliminar un perfil

```
1 DROP PROFILE mi_perfil CASCADE;
```

El uso de **CASCADE** asegura que el perfil se elimine incluso si está asignado a usuarios.

7. Consultar información sobre perfiles

```
1 SELECT PROFILE , RESOURCE_NAME , LIMIT
2 FROM DBA_PROFILES
3 WHERE PROFILE = 'MI_PERFIL';
```

1.3 Respaldo y recuperación de base de datos

1.3.1 Exportación de objetos y datos

La exportación es el proceso de crear una copia de seguridad de los datos y objetos de tu base de datos. En un sistema como **Oracle Database** esto se realiza utilizando la utilidad *expdp*.

1. Exportar un esquema completo

Este comando exportará todos los objetos (tablas, vistas, índices, procedimientos, etc.) del esquema *mi_usuario* a un archivo *.dmp*.

```
1 expdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_esquema.dmp SCHEMAS=mi_usuario
   LOGFILE=backup_esquema.log
```

- **DIRECTORY:** El directorio en el servidor donde se guardará el archivo.
- **DUMPFILE:** Nombre del archivo de volcado.
- **SCHEMAS:** Esquema a exportar.
- **LOGFILE:** Archivo donde se registrarán los detalles de la exportación.

2. Exportar solo tablas específicas

Si solo se necesitan exportar algunas tablas, se puede usar el parámetro **TABLES**:

```
1 expdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_tablas.dmp TABLES=tabla1,tabla2
   LOGFILE=backup_tablas.log
```

3. Exportar con filtros (ej. filas específicas)

Para exportar solo ciertas filas de una tabla:

```
1 expdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_ventas.dmp TABLES=ventas QUERY=
   ventas:"WHERE fecha > '01-JAN-2024'" LOGFILE=
   backup_ventas.log
```

1.3.2 Importación de objetos y datos

La importación es el proceso de restaurar los datos y objetos desde un archivo de exportación previamente creado, en un sistema como **Oracle Database** esto se puede lograr utilizando *textbfimpdp*.

1. Importar un esquema completo

Este comando importa todos los objetos desde un archivo *.dmp* en el esquema *mi_usuario*.

```
1 impdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_esquema.dmp SCHEMAS=mi_usuario
   LOGFILE=import_esquema.log
```

2. Importar tablas específicas

Restaurar algunas tablas:

```
1 impdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_tablas.dmp TABLES=tabla1,tabla2
   LOGFILE=import_tablas.log
```

3. Importar con remapeo de esquema

Para importar datos a un esquema diferente, puedes usar

REMAP_SCHEMA:

```
1 impdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio
   DUMPFILE=backup_esquema.dmp REMAP_SCHEMA=
   antiguo_usuario:nuevo_usuario LOGFILE=
   import_remapeo.log
```

4. Importar solo datos (sin índices, constraints, etc.)

Para importar únicamente los datos sin los objetos adicionales:

```
1  impdp mi_usuario/ñmi_contrasea DIRECTORY=mi_directorio  
    DUMPFILE=backup_tablas.dmp CONTENT=DATA_ONLY  
    LOGFILE=import_datos.log
```