

Unidad 4. Operación y Mantenimiento

4.1 Archivo Log del SGBD

El archivo de log en un Sistema de Gestión de Bases de Datos (SGBD) es un componente crítico que registra todas las transacciones y cambios realizados en la base de datos. A continuación, se presenta una descripción detallada de su función y su relación con SQL Server.

Funciones del Archivo Log

1. **Registro de Transacciones:**
 - Cada transacción que se realiza (inserciones, actualizaciones, eliminaciones) se registra en el archivo de log antes de ser confirmada. Esto asegura que las operaciones puedan ser revertidas en caso de error.
2. **Recuperación ante Fallos:**
 - En caso de una caída del sistema o un fallo, el archivo log permite la recuperación de la base de datos al estado más reciente antes del fallo. SQL Server utiliza este archivo para restaurar datos y asegurar la integridad.
3. **Soporte para la Replicación:**
 - Los logs también son utilizados en procesos de replicación, donde los cambios en una base de datos se reflejan en otra.
4. **Auditoría y Monitoreo:**
 - Proporciona un rastro de auditoría que puede ser utilizado para analizar cambios y actividades en la base de datos.

En SQL Server, el archivo de log se encuentra en el formato de archivo .ldf (Log Data File). A continuación, se detallan aspectos específicos de cómo SQL Server maneja estos archivos:

1. Estructura del Archivo Log:

- SQL Server organiza el archivo log en grupos de registros de transacciones. Cada registro incluye información sobre la operación realizada, el objeto afectado y el estado de la transacción.

2. Modelo de Recuperación:

- SQL Server permite elegir entre diferentes modelos de recuperación (Completo, Moderado y Simple) que afectan cómo se gestiona el archivo log:

- ✓ **Completo:** Registra toda transacción y permite la recuperación a un punto en el tiempo.
- ✓ **Moderado:** Registra transacciones, pero permite la truncación del log.
- ✓ **Simple:** Reduce el tamaño del log eliminando registros automáticamente, pero no permite la recuperación a un punto específico.

3. Mantenimiento del Archivo Log:

- Es fundamental realizar un mantenimiento adecuado del archivo log, incluyendo su truncación y respaldo, para evitar que crezca sin control y consuma espacio en disco.

Comandos Relacionados en SQL Server

1. Visualizar el tamaño del log:

DBCC SQLPERF(logspace);

2. Hacer un Respaldo del Log (**en modelo de recuperación completo**):

BACKUP LOG [nombre_base_datos] TO DISK = 'ruta_del_respaldo.bak';

Ejemplo: respaldar el log de la BD prueba:

`BACKUP LOG prueba TO DISK = 'C:\Backups\log_base_prueba.bak';`

3. Truncar el Log (**en modelo de recuperación simple**):

CHECKPOINT;

CHECKPOINT fuerza a SQL Server a escribir todos los cambios de la memoria (buffer cache) en el disco. Esto incluye las modificaciones realizadas en las páginas de datos desde la última vez que se ejecutó un checkpoint.

RESTAURAR RESPALDOS DE LA BASE DE DATOS

Asegúrate de que la carpeta **C:\respaldosql** exista antes de realizar el respaldo.

Proceso en Modo Gráfico

1. Crear la base de datos `ferreteria2`

- Abre SQL Server Management Studio (SSMS).
- Conéctate a tu instancia de SQL Server.
- Haz clic derecho en "Databases" y selecciona "New Database...".
- En la ventana que aparece, ingresa `ferreteria2` como el nombre de la base de datos.
- Haz clic en "OK" para crear la base de datos.

2. Generar un respaldo

- Haz clic derecho en la base de datos `ferreteria2`.
- Selecciona "Tasks" > "Back Up...".
- En el campo "Destination", selecciona "Disk" y haz clic en "Add...".
- Escribe la ruta `C:\respaldosql\ferreteria2.bak` y haz clic en "OK".
- Haz clic en "OK" para realizar el respaldo.

3. Eliminar la base de datos `ferreteria2`

- Haz clic derecho en la base de datos `ferreteria2`.
- Selecciona "Delete".
- Asegúrate de marcar la opción "Close existing connections" si es necesario.
- Haz clic en "OK" para eliminar la base de datos.

4. Recuperar el respaldo

- Haz clic derecho en "Databases" y selecciona "Restore Database...".
- En la ventana, selecciona "Device" y luego haz clic en "..." para buscar el archivo de respaldo.
- Agrega el archivo `C:\respaldosql\ferreteria2.bak`.
- Asegúrate de que la base de datos se llame `ferreteria2` en el campo "Database name".
- Haz clic en "OK" para restaurar la base de datos.

Proceso en Modo Programador

1. Crear la base de datos `ferreteria2`

CREATE DATABASE ferreteria2;

2. Generar un respaldo

BACKUP DATABASE ferreteria2

TO DISK = 'C:\respaldosql\ferreteria2.bak';

3. Eliminar la base de datos `ferreteria2`

DROP DATABASE ferreteria2;

4. Recuperar respaldo.

RESTORE DATABASE ferreteria2

FROM DISK = 'C:\respaldosql\ferreteria2.bak';

4.2 Definición de los modos de operación de un SGBD (alta, baja, recovery) y comandos de activación en SQL server.

Modos de Operación de un SGBD en SQL Server

Los sistemas de gestión de bases de datos (SGBD) operan en diferentes modos que se refieren a cómo manejan las transacciones y la recuperación de datos. En SQL Server, los modos de operación se pueden clasificar en tres categorías principales: alta, baja y recuperación (recovery). A continuación, se desarrolla cada uno de estos modos y sus respectivos comandos de activación.

1. Modo de Alta (High Availability)

Definición:

El modo de alta disponibilidad se refiere a las configuraciones y técnicas que garantizan que una base de datos esté siempre disponible y accesible, minimizando el tiempo de inactividad. Esto es crucial para aplicaciones críticas que requieren un acceso constante a los datos.

Características:

- **Failover Clustering:** Utiliza múltiples servidores físicos para proporcionar redundancia. Si un servidor falla, otro toma su lugar sin interrupciones.
- **Always On Availability Groups:** Permite que múltiples bases de datos sean replicadas y disponibles en varios nodos, asegurando la alta disponibilidad y recuperación ante desastres.

Comandos de Activación:

- Para configurar un grupo de disponibilidad siempre activo:

```
CREATE AVAILABILITY GROUP [NombreGrupo]
```

```
FOR DATABASE [NombreBaseDatos]
```

```
REPLICA ON 'ServidorPrimario' WITH (ENDPOINT_URL =  
'TCP://ServidorPrimario:sepuerto')
```

2. Modo de Baja (Low Availability)

El modo de baja disponibilidad se refiere a configuraciones donde la base de datos puede no estar disponible durante ciertos períodos. Esto puede ser aceptable en entornos de desarrollo o en bases de datos donde la pérdida de datos es tolerable.

Características:

- **Backup y Restore:** Las bases de datos pueden ser respaldadas y restauradas, pero no se garantiza un acceso constante.
- **Mantenimiento Programado:** Las bases de datos pueden estar fuera de línea durante tareas de mantenimiento.

Comandos de Activación:

- Para poner una base de datos en modo de solo lectura (que puede ser considerado un modo de baja disponibilidad):

ALTER DATABASE [NombreBaseDatos] SET READ_ONLY;

3. Modo de Recuperación (Recovery Mode)

El modo de recuperación se refiere a cómo SQL Server maneja el registro de transacciones y la recuperación de datos. Hay tres modelos de recuperación: completo, simple y bulk-logged.

Modelos de Recuperación

1. FULL (Completo):

- ❖ **Descripción:** Registra todas las transacciones y permite la recuperación a un punto en el tiempo.
- ❖ **Uso:** Ideal para bases de datos críticas donde la pérdida de datos es inaceptable.
- ❖ **Número:** 1

2. SIMPLE (Sencillo):

- ❖ **Descripción:** No registra información de log de transacciones, lo que significa que el log se trunca automáticamente después de que las transacciones se completan.
- ❖ **Uso:** Adecuado para bases de datos donde la pérdida de datos es aceptable y donde no se requiere recuperación a un punto en el tiempo.
- ❖ **Número:** 2

3. BULK_LOGGED:

- ❖ **Descripción:** Registra operaciones de log mínimas para operaciones de carga masiva, lo que reduce el tamaño del log durante estas operaciones.
- ❖ **Uso:** Se utiliza en escenarios de carga masiva donde el rendimiento es más crítico que la recuperación a un punto específico.
- ❖ **Número:** 3

Para obtener el cual es el modelo de recuperación de una base de datos y su valor numérico, puedes ejecutar la siguiente consulta:

```
SELECT name, recovery_model_desc, database_id  
  
FROM sys.databases;
```

Podemos cambiar el modelo mediante los siguientes comandos:

1. Modelo de Recuperación Completo (FULL)

Este modelo permite la recuperación a un punto en el tiempo y requiere respaldos de log de transacciones.

```
ALTER DATABASE [NombreBaseDatos]  
SET RECOVERY FULL;
```

2. Modelo de Recuperación Sencillo (SIMPLE)

Este modelo no mantiene un registro de transacciones extensivo, permitiendo que el log de transacciones se trunque automáticamente.

```
ALTER DATABASE [NombreBaseDatos]  
SET RECOVERY SIMPLE;
```

3. Modelo de Recuperación Bulk-Logged

Este modelo optimiza el registro de log para operaciones de carga masiva, pero no permite la recuperación a un punto específico en el tiempo durante esas operaciones.

```
ALTER DATABASE [NombreBaseDatos]  
SET RECOVERY BULK_LOGGED;
```

Investigar : como recuperar una base de datos mediante el respaldo de un log (.ldf) subir su investigación al padlet.