

Tema

1

Planificación de una Bases Datos en MS SQL Server



Objetivos y resultados de aprendizaje

- Una parte importante del trabajo de un administrador de bases de datos consiste en planear y crear bases de datos.
- Entender cómo se almacena los datos, procesan las transacciones y se usa los archivos y grupos de archivos ayuda a tomar decisiones sobre la mejor ubicación de los archivos de base de datos:
 - ✓ Permite obtener el máximo rendimiento y optimizar la recuperación de desastres.

Objetivos y resultados de aprendizaje

- Los objetivos específicos consisten en:
 - ✓ Describir cómo se almacenan los datos en el SGBD
 - ✓ Describir el funcionamiento del registro de transacciones.
 - ✓ Describir las consideraciones para la ubicación de archivos
 - ✓ Describir el uso de archivos y grupos de archivos.
 - ✓ Describir cómo planear la capacidad

Evaluación del tema

- Los resultados de aprendizaje correspondiente a este tema se evaluarán con los siguientes tipos de pruebas:
 - ✓ Pruebas escritas de carácter teórico-práctico
 - ✓ Entrega de Prácticas
 - ✓ Participación en clase

Bibliografía

- Para obtener más información puedes consultar:
 - ✓ Libros en pantalla de SQL Server.
 - ✓ STANEK, WILLIAM R. SQL Server 2012 Guía del Administrador. Editorial Anaya. Madrid 2013. ISBN 84-4153221-2.

Índice de contenidos

- ✓ Almacenamiento de Datos
- ✓ Transacciones
- ✓ Registro de Transacciones
- ✓ Tipos de Archivos
- ✓ Ubicación de Archivos
- ✓ Grupo de Archivos
- ✓ Planeamiento de la Capacidad
- ✓ Práctica

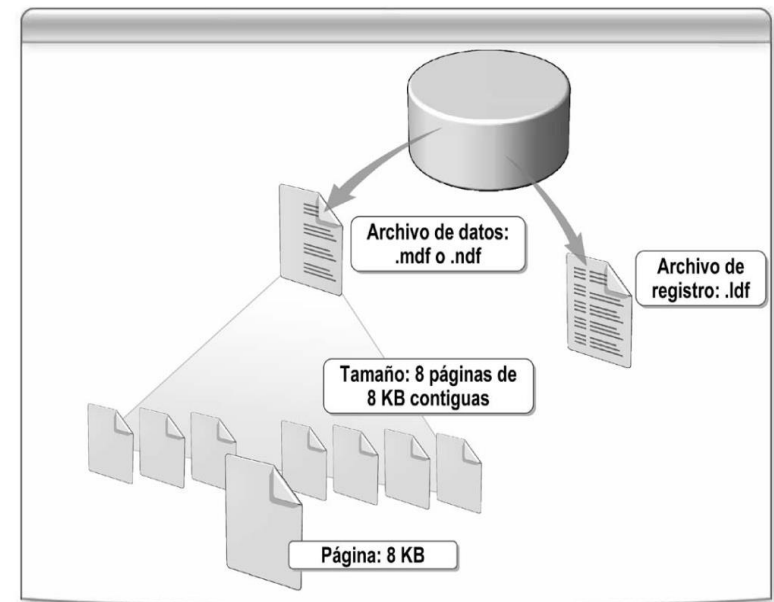
- Todas las bases de datos tienen un archivo de datos principal y uno o más archivos de registro de transacciones. Una base de datos también puede tener archivos secundarios de datos.
- Estos archivos físicos tienen los nombres de los archivos del sistema operativo y los nombres de los archivos lógicos que se pueden usar en las instrucciones Transact-SQL.

- En el caso concreto de SQL Server son
 - ✓ Archivo de datos principal (.mdf)
 - ✓ Archivos secundarios de datos (.ndf)
 - ✓ Archivos de registro de transacciones (.ldf)
- Y su ubicación es de forma predeterminada:
 - ✓ C:\Archivos de Programa\Microsoft SQL\Server\MSSQL.n(*)\MSSQL\Data

(*) n es un número único que identifica una instancia instalada.

Almacenamiento de Datos

- Al crear una base de datos, una copia de la base de datos model, donde se incluyen las tablas del sistema, se copia en la base de datos y el resto de la base de datos se rellena con páginas vacías.
- Los datos se almacenan en bloques de 8 kilobytes (KB) de espacio en disco contiguo llamado páginas. Esto significa que una base de datos puede almacenar 128 páginas por megabyte (MB).



- Las filas no pueden abarcar varias páginas. Por consiguiente, la cantidad máxima de datos en una fila única menos el espacio requerido para la carga de fila es de 8.060 bytes. Hay dos excepciones a esta regla:
 - ✓ La característica de desbordamiento de fila en SQL Server permite que las filas contengan columnas definidas como varchar, nvarchar, varbinary y sql_variant o tipos CLR definidos por el usuario que pueden exceder el tamaño de la página siempre y cuando cada columna se encuentre dentro del límite de 8.000 bytes.
 - ✓ Las columnas definidas como texto, imagen o binario, así como las columnas de tipo varchar, nvarchar y varbinary definidas con máximo, se almacenan como un puntero en la página de datos que hace referencia a un conjunto de páginas de desbordamiento donde se almacena el valor de columna real.

- Las tablas y los índices se almacenan en extensiones. Una extensión es una colección de ocho páginas físicamente contiguas, o bien 64 KB. Por consiguiente, una base de datos tiene 16 extensiones por megabyte. Las tablas pequeñas pueden compartir las extensiones con otros objetos de la base de datos.
- Los archivos de registro de transacciones contienen la información necesaria para la recuperación de la base de datos en caso de un error del sistema, pero no contienen las páginas de datos.

- Una transacción es un conjunto de una o más instrucciones Transact-SQL tratadas como una única unidad de trabajo y recuperación. Dentro de una transacción, las instrucciones Transact-SQL se deben realizar por completo o no realizarse.
- De forma predeterminada, SQL Server funciona en modo de confirmación automática. Esto significa que una transacción implícita se confirma después de la ejecución sin una instrucción COMMIT TRANSACTION que finalice la transacción.

- SQL Server realiza una transacción implícita cuando cualquiera de las siguientes instrucciones Transact-SQL se ejecuta como una transacción:

✓ ALTER TABLE

✓ CREATE

✓ DELETE

✓ DROP

✓ FETCH

✓ GRANT

✓ INSERT

✓ OPEN

✓ REVOKE

✓ SELECT

✓ TRUNCATE

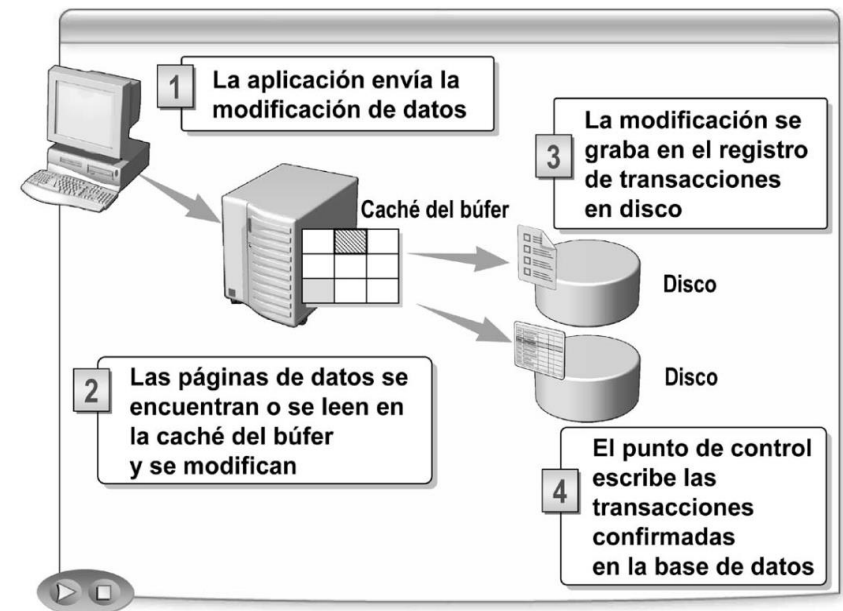
TABLE

✓ UPDATE

- SQL Server también puede funcionar en modo de transacción explícita. Esto significa que cuando cualquiera de las instrucciones Transact-SQL mencionadas inicia una transacción, la transacción debe finalizar con la instrucción COMMIT TRANSACTION.
- SQL Server 2012 realiza una transacción explícita cuando se definen explícitamente el principio y el fin de la transacción. Puede definir el principio y fin de la transacción en Transact-SQL usando las instrucciones BEGIN TRANSACTION y COMMIT TRANSACTION.

Registro de Transacciones

- SQL Server registra todas las transacciones en un registro de transacciones para mantener la coherencia de la base de datos y contribuir en la recuperación. El registro es un área de almacenamiento que realiza automáticamente un seguimiento de los cambios efectuados en una base de datos.
- SQL Server registra las modificaciones en el registro del disco a medida que dichas modificaciones se ejecutan y antes de que se escriban en la base de datos.



- El proceso de registro sigue estos pasos:
 - ✓ Una aplicación envía una modificación de datos.
 - ✓ Cuando se ejecuta la modificación, SQL Server carga las páginas de datos afectadas del disco en la memoria, llamada caché del búfer, siempre y cuando una consulta anterior no las haya puesto ya en la caché del búfer.
 - ✓ SQL Server graba todas las instrucciones de modificación de datos en el registro a medida que se producen. El cambio siempre se graba en el registro y se escribe en el disco antes de aplicarse a la base de datos. Este tipo de registro se denomina registro de escritura anticipada.
 - ✓ Periódicamente, el proceso de punto de comprobación escribe en el disco todas las transacciones completadas en la base de datos.

- Si se produce un error del sistema, el proceso de recuperación automática usa el registro de transacciones para poner al día todas las transacciones confirmadas y revertir las transacciones incompletas.
- El registro usa marcadores de transacción durante la recuperación automática para determinar los puntos inicial y final de una transacción.
- Se considera que una transacción está completa cuando el marcador `BEGIN TRANSACTION` tiene asociado un marcador `COMMIT TRANSACTION`.
- Las páginas de datos se escriben en el disco cuando se produce un punto de comprobación.

- El archivo principal contiene la información de inicio de la base de datos y apunta al resto de archivos de la base de datos. Los datos de usuario y los objetos se pueden almacenar en este archivo principal o en archivos secundarios de datos.
- Puede crear archivos secundarios para almacenar datos de usuario. Algunas bases de datos pueden ser lo bastante grandes para que requieran varios archivos secundarios de datos, o también pueden usar los archivos secundarios en unidades de disco independientes para repartir los datos entre varios discos.

- Cada base de datos debe tener un registro de transacciones. A menos que se especifique lo contrario, un archivo de registro de transacciones se crea automáticamente con un nombre generado por el sistema.
- Normalmente, el archivo de registro de transacciones debe ser aproximadamente el 25% del tamaño de la base de datos.
- Es importante la gestión del mismo a partir de los procesos de:
 - ✓ Backup
 - ✓ Truncado
 - ✓ Reducción

- Se puede mejorar el rendimiento e implementar la tolerancia a errores si se administra la ubicación de archivos de datos y registros de transacciones en discos.
 - ✓ Para bases de datos más grandes, debe distribuir tantos datos por tantas unidades físicas como sea posible. De esa manera, se produce una mejora en el rendimiento puesto que se tiene acceso a los datos en paralelo usando varios archivos.
 - ✓ En general, cree un archivo para cada disco físico y agrupe los archivos en uno o más grupos de archivos.

- SQL Server puede realizar los recorridos siguientes:
 - ✓ Recorridos de los datos en paralelo si el equipo tiene varios procesadores y varios discos.
 - ✓ Varios recorridos en paralelo para una tabla única si el grupo de archivos de la tabla contiene varios archivos.

- Para distribuir de forma uniforme los datos en todos los discos, use las tecnologías de matriz redundante de discos independientes RAID y, si es necesario, use a continuación grupos de archivos definidos por el usuario para distribuir los datos por varios conjuntos de secciones de hardware.
 - ✓ Una técnica avanzada, es separar las tablas de los índices no agrupados. Un índice no agrupado es un índice cuyo orden lógico es diferente al orden físico almacenado de las filas en disco.

- Debe crear el registro de transacciones en un disco independiente o usar RAID.
 - ✓ Dado que el archivo de registro de transacciones se escribe secuencialmente, usar un disco dedicado independiente permite que los cabezales de disco permanezcan en posición correcta para la operación de escritura siguiente. RAID también proporciona tolerancia a errores.
 - ✓ **Ejemplo:** si su entorno de producción tiene varias bases de datos en un servidor, podría usar discos independientes para cada registro de transacciones. Esta estrategia contribuye a obtener un rendimiento óptimo.

- Respecto a la base de datos de sistemas Tempdb:
 - ✓ Es conveniente ubicarla en un subsistema de E/S rápido independiente de las bases de datos de usuario para garantizar un rendimiento óptimo.
 - ✓ Puede usar RAID para distribuir la base de datos Tempdb por varios discos para obtener un mayor rendimiento.

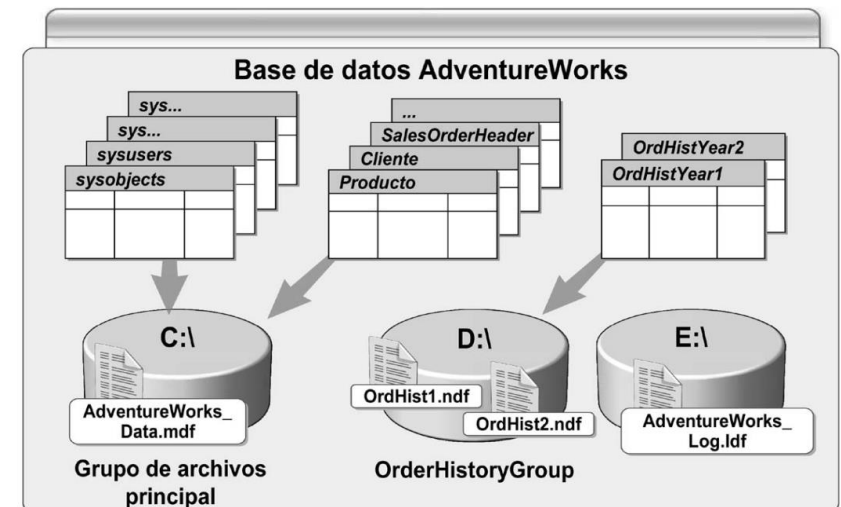
(*)Para obtener más información sobre cómo administrar el espacio en disco usado por la base de datos tempdb, consulta “Solucionar problemas de espacio en disco insuficiente en tempdb” y “Optimizar el rendimiento de tempdb” en los Libros en pantalla de SQL Server.

- Los grupos de archivos son colecciones de archivos con nombre y se usan para simplificar la ubicación de los datos.
- Si su configuración de hardware incluye varias unidades de disco, puede ubicar objetos y archivos específicos en discos individuales mediante la agrupación de los archivos de base de datos en uno o más grupos de archivos. Los grupos de archivos mejoran el rendimiento al distribuir los datos por varios discos y usar subprocesos paralelos para el procesamiento de consultas. Los grupos de archivos también pueden facilitar el mantenimiento de la base de datos.

- El grupo de archivos principal contiene el archivo principal de datos con las tablas del sistema. También contiene cualquier archivo secundario definido por el usuario que no esté asignado a otros grupos de archivos.
- Un grupo de archivos definido por el usuario consta de archivos de datos agrupados con fines de ubicación y administración.

Grupos de Archivos

- Los grupos de archivos usan una estrategia de relleno proporcional en todos los archivos del grupo de archivos. Cuando los datos se escriben en el grupo de archivos, cada archivo se rellena en paralelo.
 - ✓ Cada archivo se coloca físicamente en un disco o conjunto de discos.
 - ✓ SQL Server 2012 mantiene una asignación de archivos que asocia cada objeto de base de datos a su ubicación en el disco.
- Los archivos de registro de transacciones no forman parte de un grupo de archivos.
 - ✓ El espacio del registro de transacciones se administra por separado del espacio de datos.



- Grupo de archivos predeterminado:
 - ✓ contiene las páginas para todas las tablas e índices que no tienen un grupo de archivos específico cuando se crean.
 - ✓ Cambie el grupo de archivos predeterminado si usa grupos de archivos definidos por el usuario. Si su base de datos tiene varios grupos de archivos, debería asignar como predeterminado uno de los grupos de archivos definidos por el usuario.
 - ✓ Esto evitará que el crecimiento inesperado de la tabla provoque una restricción en las tablas del sistema del grupo de archivos principal.

- Grupo de archivos Principal:
 - ✓ El grupo de archivos principal debe ser lo bastante grande para contener todas las tablas del sistema y cualquier tabla e índice que no se haya asignado a un grupo de archivos definido por el usuario.
 - ✓ Si el grupo de archivos principal se queda sin espacio, no podrá agregar información a las tablas del sistema.
 - ✓ Si un grupo de archivos definido por el usuario se queda sin espacio, sólo se ven afectados los archivos de usuario que están específicamente asignados a ese grupo de archivos.

- En general:
 - ✓ Cualquier grupo de archivos, a excepción del grupo de archivos principal, se puede marcar como de sólo lectura. Un grupo de archivos marcado como de sólo lectura no se puede modificar de ningún modo.
 - ✓ Para evitar alteraciones accidentales en los datos, coloque las tablas pertinentes en un grupo de archivos y marque el grupo de archivos como de sólo lectura.
 - ✓ SQL Server admite la compresión NTFS de grupos de archivos definidos por el usuario como de sólo lectura y bases de datos de sólo lectura.
 - ✓ Debe considerar comprimir los datos de sólo lectura si ha limitado el espacio en disco y tiene un volumen grande de datos estáticos que deben estar disponibles para un acceso de sólo lectura.

- El tamaño del registro de transacciones, que se ve afectado por la cantidad y frecuencia de la actividad de modificación, el tamaño de cada transacción y con qué frecuencia se realizan copias de seguridad o se vuelca el registro.
 - ✓ Como punto de partida, debería asignar el 25% del tamaño de la base de datos al registro de transacciones para entornos de procesamiento de transacciones en línea (OLTP, online transaction processing).

- El tamaño de las tablas del sistema, como el número de usuarios, objetos, etc., que no suele representar un porcentaje grande del tamaño de la base de datos.
 - ✓ Puede asignar un porcentaje menor para bases de datos que se usan principalmente para consultas.

(*)Para obtener una descripción detallada del planeamiento de la capacidad, consulta “Estimar el tamaño de una base de datos” en los Libros en pantalla de SQL Server.

Práctica del tema

- Al realizar la práctica una vez finalizado el tema debes tener en cuenta:
 - ✓ Después de crear una base de datos, puede establecer las opciones de base de datos mediante SQL Server Management Studio o la instrucción ALTER DATABASE de Transact-SQL.
 - ✓ Puede configurar varias opciones de base de datos, pero puede establecerlas sólo para una base de datos cada vez. Para que las opciones afecten a todas las bases de datos nuevas, modifique la base de datos model.

Establecer las opciones de la base de datos usando:

- SQL Server Management Studio

- Instrucción ALTER DATABASE

Categoría de opción	Propósito
Automático	Controla los comportamientos automáticos como las estadísticas, el cierre de la base de datos y la reducción
Disponibilidad	Controla si la base de datos está en línea, quién puede conectarse a ella y si es de sólo lectura
Cursor	Controla el comportamiento y el ámbito del cursor
Recuperación	Controla el modelo de recuperación de la base de datos
SQL	Controla las opciones de compatibilidad con ANSI como valores nulos ANSI y desencadenadores recursivos

Práctica del tema

- ✓ Después de crear una base de datos, puede establecer las opciones de base de datos mediante SQL Server Management Studio o la instrucción ALTER DATABASE de Transact-SQL.
- ✓ Puede configurar varias opciones de base de datos, pero puede establecerlas sólo para una base de datos cada vez. Para que las opciones afecten a todas las bases de datos nuevas, modifique la base de datos model.

Establecer las opciones de la base de datos usando:

- SQL Server Management Studio
- Instrucción ALTER DATABASE

Categoría de opción	Propósito
Automático	Controla los comportamientos automáticos como las estadísticas, el cierre de la base de datos y la reducción
Disponibilidad	Controla si la base de datos está en línea, quién puede conectarse a ella y si es de sólo lectura
Cursor	Controla el comportamiento y el ámbito del cursor
Recuperación	Controla el modelo de recuperación de la base de datos
SQL	Controla las opciones de compatibilidad con ANSI como valores nulos ANSI y desencadenadores recursivos