

Creación y Administración de Bases de Datos

Contenido

Introducción	1
Creación de bases de datos	2
Creación de grupos de archivos	11
Administración de bases de datos	14
Introducción a las estructuras de datos	25

Notas para el instructor

Este módulo proporciona a los alumnos una descripción de cómo crear una base de datos, configurar opciones de base de datos, crear grupos de archivos y administrar una base de datos y el registro de transacciones. Repasa la asignación de espacio de disco y cómo el registro de transacciones graba las modificaciones a los datos.

Nota Este curso se basa en la base de datos **Northwind**. El esquema de la base de datos **Northwind** se encuentra en el apéndice A. Las prácticas utilizan una versión paralela de la base de datos **Northwind** denominada **ClassNorthwind**.

En la práctica, los alumnos definen la base de datos **ClassNorthwind**, la modifican y, después, establecen una opción para borrar el registro de transacciones.

Después de completar este módulo, los alumnos serán capaces de:

- Crear una base de datos.
- Crear un grupo de archivos.
- Administrar una base de datos.
- Describir estructuras de datos.

Introducción

Objetivo del tema

Proporcionar una introducción a los temas y objetivos del módulo.

Explicación previa

En este módulo aprenderá a crear y administrar bases de datos y grupos de archivos, y cómo SQL Server almacena los datos.

- Creación de bases de datos
- Creación de grupos de archivos
- Administración de bases de datos
- Introducción a las estructuras de datos

Este módulo describe cómo crear una base de datos, establecer las opciones de base de datos, crear grupos de archivos y administrar la base de datos y el registro de transacciones. También describe cómo Microsoft® SQL Server™ 2000 almacena los datos.

Después de completar este módulo, el alumno será capaz de:

- Crear una base de datos.
- Crear un grupo de archivos.
- Administrar una base de datos.
- Describir estructuras de datos.

◆ Creación de bases de datos

Objetivo del tema

Describir cómo crear bases de datos.

Explicación previa

Esta sección describe cómo crear bases de datos.

- Definición de las bases de datos
- Cómo funciona el registro de transacciones
- Configuración de las opciones de base de datos
- Recuperación de información de la base de datos

Esta sección describe cómo crear bases de datos, especificar opciones de base de datos y recuperar información de la base de datos. También describe cómo funciona el registro de transacciones.

Definición de las bases de datos

Objetivo del tema

Describir cómo definir una base de datos.

Explicación previa

Puede definir una base de datos con el Administrador corporativo de SQL Server o mediante instrucciones Transact-SQL.

■ La creación de una base de datos define:

- El nombre de la base de datos
- El tamaño de la base de datos
- Los archivos que componen la base de datos

```
CREATE DATABASE Sample
ON
    PRIMARY ( NAME=SampleData,
    FILENAME='c:\Archivos de programa\...\Data\Sample.mdf',
    SIZE=10MB,
    MAXSIZE=15MB,
    FILEGROWTH=20%)
LOG ON
    ( NAME=SampleLog,
    FILENAME= 'c:\Archivos de programa\...\Data\Sample.ldf',
    SIZE=3MB,
    MAXSIZE=5MB,
    FILEGROWTH=1MB)
COLLATE SQL_Latin1_General_CP1_CI_AS
```

Sugerencia

Muestre las dos maneras de definir una base de datos con el Administrador corporativo de SQL Server.

Puede definir una base de datos mediante el Administrador corporativo de SQL Server o mediante la instrucción CREATE DATABASE en el Analizador de consultas SQL. El proceso de definición de una base de datos crea también un registro de transacciones para esa base de datos.

La información acerca de las bases de datos de SQL Server está almacenada en la tabla **sysdatabases** de la base de datos **master**. Por tanto, debe utilizar la base de datos **master** para definir una base de datos cuando utilice instrucciones Transact-SQL.

La definición de una base de datos consiste en especificar el nombre de la base de datos, y diseñar el tamaño y la ubicación de los archivos de la base de datos. Cuando se crea la nueva base de datos, ésta es un duplicado de la base de datos **model**. Las opciones o configuraciones de la base de datos **model** se copian en la nueva base de datos.

Importante Debe hacer una copia de seguridad de la base de datos **master** cada vez que cree, modifique o elimine una base de datos.

Sintaxis

```
CREATE DATABASE baseDeDatos
[ON
    { [PRIMARY] (NAME = archivoLógico,
        FILENAME = 'archivoSO'
        [, SIZE = tamaño]
        [, MAXSIZE = {tamañoMáx | UNLIMITED})
        [, FILEGROWTH = incrementoCrecimiento] )
    } [,...n]
]
[LOG ON
    { ( NAME = archivoLógico,
        FILENAME = 'archivoSO'
        [, SIZE = tamaño]
        [, MAXSIZE = {tamañoMáx | UNLIMITED})
        [, FILEGROWTH = incrementoCrecimiento] )
    } [,...n]
]
[COLLATE nombreIntercalación]
```

Al crear una base de datos se pueden establecer los siguientes parámetros:

PRIMARY Este parámetro especifica los archivos del grupo de archivos principal. El grupo de archivos principal contiene todas las tablas del sistema de la base de datos. También contiene todos los objetos no asignados a grupos de archivos del usuario. Todas las bases de datos tienen un archivo de datos principal. El archivo de datos principal es el punto de inicio de la base de datos y señala a los demás archivos de la base de datos. La extensión de nombre de archivo recomendada para los archivos de datos principales es .mdf. Si no se especifica la palabra clave PRIMARY, el primer archivo enumerado en la instrucción es el archivo principal.

FILENAME Este parámetro especifica el nombre del archivo del sistema operativo y la ruta de acceso al archivo. La ruta acceso de *archivoSO* debe especificar una carpeta del servidor en el que está instalado SQL Server.

SIZE Este parámetro especifica el tamaño de los archivos de datos o de registro. Puede especificar los tamaños en megabytes (MB), que es el valor predeterminado, o en kilobytes (KB). El tamaño mínimo es de 512 KB, tanto para los archivos de datos como para los archivos de registro. El tamaño especificado para el archivo de datos principal debe ser, al menos, como el tamaño del archivo principal de la base de datos **model**. Cuando se agrega un archivo de datos o un archivo de registro, el valor predeterminado es 1 MB.

MAXSIZE Este parámetro especifica el tamaño máximo hasta el que puede crecer el archivo. Puede especificar el tamaño en megabytes (valor predeterminado) o en kilobytes. Si no se especifica el tamaño, el archivo crece hasta que el disco esté lleno.

Puede especificar el crecimiento del archivo de tres formas: en megabytes, en kilobytes o como un porcentaje. El porcentaje sólo se aplica al crecimiento del archivo, no al tamaño máximo.

FILEGROWTH Este parámetro especifica el incremento de crecimiento del archivo. El valor de FILEGROWTH de un archivo no puede sobrepasar el valor de MAXSIZE. Un valor de 0 indica que no hay crecimiento. El valor se puede especificar en megabytes (el valor predeterminado), en kilobytes o como porcentaje (%). Si no se especifica FILEGROWTH, el valor predeterminado es el 10 por ciento y el valor mínimo es 64 KB (una extensión). El tamaño especificado se redondea al múltiplo de 64 KB más próximo.

COLLATION

Este parámetro especifica la intercalación predeterminada para la base de datos. La intercalación incluye las reglas que gobiernan el uso de caracteres de un lenguaje o un alfabeto.

Ejemplo

El siguiente ejemplo crea una base de datos denominada **Sample** con un archivo de datos principal de 10 MB y un archivo de registro de 3 MB en una instancia predeterminada de SQL Server.

Sugerencia

Mencione que la ruta de acceso del archivo variará si se utiliza una instancia con nombre de SQL Server en lugar de una instancia predeterminada.

```
CREATE DATABASE Sample
ON
    PRIMARY ( NAME=SampleData,
    FILENAME='c:\Archivos de programa\
        Microsoft SQL Server\MSSQL\Data\Sample.mdf',
    SIZE=10MB,
    MAXSIZE=15MB,
    FILEGROWTH=20%)
LOG ON
    ( NAME=SampleLog,
    FILENAME='c:\Archivos de programa\
        Microsoft SQL Server\MSSQL\Data\Sample.ldf',
    SIZE=3MB,
    MAXSIZE=5MB,
    FILEGROWTH=1MB)
COLLATE SQL_Latin1_General_CP1_CI_AS
```

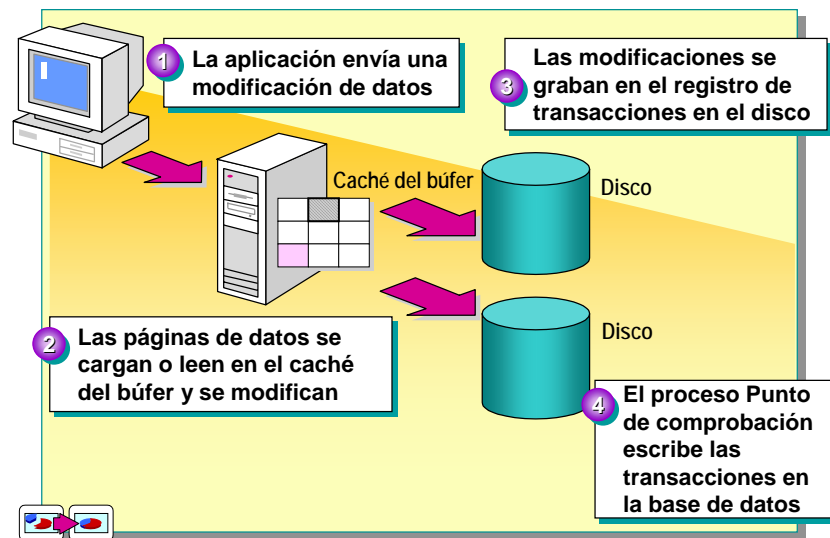
Cómo funciona el registro de transacciones

Objetivo del tema

Describir cómo funciona el registro de transacciones.

Explicación previa

El registro de transacciones graba las modificaciones de los datos a medida que se producen.

**Sugerencia**

Ésta es una diapositiva animada. Consulte las Notas para el instructor si necesita ayuda para desplazarse por esta diapositiva.

SQL Server graba todas las transacciones en un registro de transacciones para mantener la coherencia de la base de datos y facilitar la recuperación. El registro es un área de almacenamiento que efectúa automáticamente el seguimiento de todos los cambios en la base de datos. SQL Server graba las modificaciones al registro en disco cuando se ejecutan, antes de que se escriban en la base de datos.

El proceso de registro es el siguiente:

1. La aplicación envía una modificación de datos.
2. Cuando la modificación se ejecuta, las páginas de datos afectadas se cargan en la *caché del búfer* desde el disco si no están ya cargadas en la caché del búfer por una consulta anterior.
3. Cada instrucción de modificación de datos se graba en el registro mientras se ejecuta. El cambio siempre se graba en el registro y se escribe en disco antes de hacer la modificación en la base de datos. Este tipo de registro se denomina registro de *preescritura*.
4. De forma repetitiva, el proceso de punto de comprobación escribe en disco todas las transacciones completadas en la base de datos.

Si se produce un error del sistema, el proceso automático de recuperación utiliza el registro de transacciones para aplicar todas las transacciones confirmadas y deshacer las transacciones incompletas.

Los marcadores de transacción del registro se utilizan durante la recuperación automática para determinar los puntos de inicio y fin de cada transacción. Una transacción se considera completa cuando el marcador BEGIN TRANSACTION tiene un marcador COMMIT TRANSACTION asociado. Las páginas de datos se escriben en disco cuando existe un punto de comprobación.

Configuración de las opciones de base de datos

Objetivo del tema

Mostrar algunas de las opciones de base de datos que se pueden configurar.

Explicación previa

En la mayor parte de las bases de datos, puede configurar varias opciones.

■ Establezca las opciones de la base de datos mediante:

- El Administrador corporativo de SQL Server
- La instrucción ALTER DATABASE

■ Categorías de la opción de base de datos

- Opciones automáticas
- Opciones de cursor
- Opciones de recuperación
- Opciones de SQL
- Opciones de estado

Después de haber creado una base de datos, puede definir opciones de base de datos con el Administrador corporativo de SQL Server o por medio de la instrucción ALTER DATABASE.

Puede configurar varias opciones de base de datos, pero sólo se puede hacer en una sola base de datos al mismo tiempo. Para que las opciones afecten a todas las bases de datos nuevas, cambie la base de datos **model**.

La tabla siguiente enumera algunas de las opciones de uso más frecuente.

Categoría de la opción de base de datos

	Opción de base de datos	Descripción
Opciones automáticas	AUTO_CREATE_STATISTICS	Crea automáticamente las estadísticas que faltan que son necesarias para la optimización de consultas. El valor predeterminado es ON.
	AUTO_UPDATE_STATISTICS	Actualiza automáticamente las estadísticas desfasadas necesarias para la optimización de consultas. El valor predeterminado es ON.
Opciones de cursor	CURSOR_CLOSE_ON_COMMIT	Cierra automáticamente los cursores abiertos cuando se confirma una transacción. El valor predeterminado es OFF y los cursores permanecen abiertos.
	CURSOR_DEFAULT LOCAL GLOBAL	CURSOR_DEFAULT_LOCAL limita el ámbito del cursor. Es local al lote, procedimiento almacenado o desencadenador en el que se creó el cursor. CURSOR_DEFAULT_GLOBAL es el valor predeterminado; el ámbito del cursor es global respecto a la conexión.

*(continuación)***Categoría de la opción de base de datos**

	Opción de base de datos	Descripción
Opciones de recuperación	RECOVERY FULL BULK_LOGGED SIMPLE	FULL proporciona recuperabilidad completa ante errores del medio; es el valor predeterminado. BULK_LOGGED utiliza menos espacio de registro porque el registro es mínimo, pero tiene un riesgo mayor de exposición. SIMPLE recupera la base de datos sólo hasta la última copia de seguridad completa de la base de datos o hasta la última copia de seguridad diferencial.
	TORN_PAGE_DETECTION	Permite a SQL Server detectar operaciones de E/S incompletas causadas por cortes de energía u otros apagones del sistema. El valor predeterminado es ON.
Opciones de SQL	ANSI_NULL_DEFAULT	Permite al usuario controlar la capacidad de asignación predeterminada de valores NULL de la base de datos. SQL Server 2000 toma como valor predeterminado NOT NULL.
	ANSI_NULLS	Si está activada, todas las comparaciones con un valor nulo se evalúan como NULL (desconocido). Si está desactivada, todas las comparaciones de valores no Unicode con un valor nulo se evalúan como verdaderas (TRUE) si ambos valores son NULL. De manera predeterminada, la opción de base de datos ANSI_NULLS es OFF.
Opciones de estado	READ_ONLY READ_WRITE	Define la base de datos como de sólo lectura (se utiliza para establecer la seguridad en las bases de datos de ayuda a la toma de decisiones) o devuelve la base de datos a operaciones de lectura y escritura.
	SINGLE_USER RESTRICTED_USER MULTI_USER	SINGLE_USER permite que los usuarios se conecten a la base de datos de uno en uno. Todas las demás conexiones se interrumpen. RESTRICTED_USER permite que sólo los miembros de la función fija de base de datos db_owner y de las funciones fijas de servidor dbcreator y sysadmin se conecten a la base de datos. MULTI_USER permite que todos los usuarios con los permisos adecuados se conecten a la base de datos. MULTI_USER es la opción predeterminada.

Recuperación de información de la base de datos

Objetivo del tema
Enumerar las diversas formas de recuperar información de la base de datos.

Explicación previa
Existen varias formas de recuperar información de la base de datos.

- Determine las propiedades de la base de datos mediante la función DATABASEPROPERTYEX
- Utilice los procedimientos almacenados del sistema para mostrar información acerca de las bases de datos y los parámetros de la base de datos
 - sp_helpdb
 - sp_helpdb *baseDeDatos*
 - sp_spaceused [*nombreObjeto*]

Puede determinar las propiedades de la base de datos con la función DATABASEPROPERTYEX.

Sintaxis

SELECT DATABASEPROPERTYEX (*base de datos, propiedad*)

Sugerencia
Demuestre cómo se examina la información de la base de datos en el Administrador corporativo de SQL Server.

La siguiente tabla enumera algunas de las propiedades de la base de datos.

Intercalación	IsFulltextEnabled
IsAnsiNullDefault	IsInStandBy
IsAnsiNullsEnabled	IsNullConcat
IsAnsiPaddingEnabled	IsQuotedIdentifiersEnabled
IsAnsiWarningsEnabled	IsRecursiveTriggersEnabled
IsArithmeticAbortEnabled	Recuperación
IsAutoCreateStatistics	Estado
IsAutoShrink	Capacidad de actualización
IsAutoUpdateStatistics	Acceso de usuario
IsCloseCursorsOnCommitEnabled	Versión

La siguiente tabla enumera los procedimientos almacenados del sistema de uso común que presentan información acerca de las bases de datos y sus parámetros.

Procedimiento almacenado de sistema	Descripción
sp_helpdb	Informa acerca de todas las bases de datos de un servidor. Proporciona el nombre, tamaño, propietario, Id., fecha de creación y opciones de la base de datos.
sp_helpdb <i>baseDeDatos</i>	Informa sólo acerca de la base de datos especificada. Proporciona el nombre, tamaño, propietario, Id., fecha de creación y opciones de la base de datos. También enumera los archivos de datos y de registro.
sp_spaceused [<i>nombreObjeto</i>]	Resume el espacio de almacenamiento que utiliza una base de datos o un objeto de base de datos.

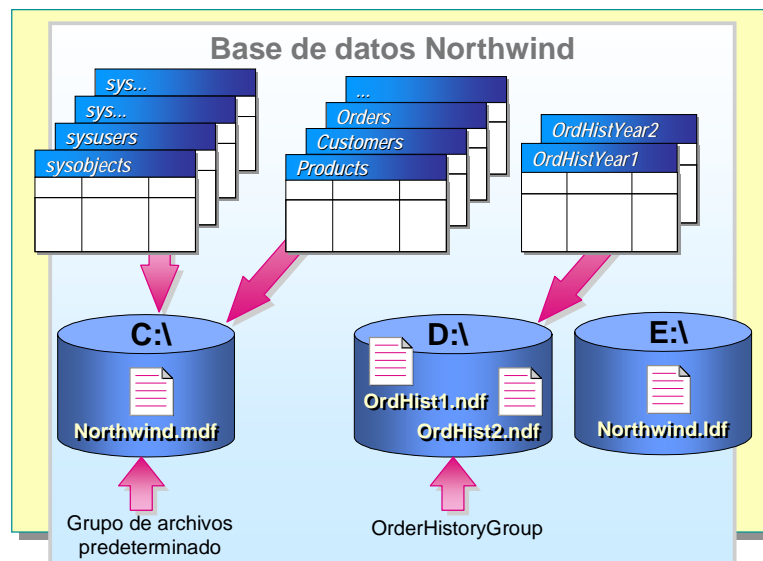
◆ Creación de grupos de archivos

Objetivo del tema

Presentar el concepto de grupos de archivos.

Explicación previa

Los grupos de archivos son una colección con nombre de uno o más archivos que forman una unidad de asignación y administración.



Si la configuración de hardware incluye varias unidades de disco, puede colocar objetos y archivos específicos en discos individuales y agrupar los archivos de la base de datos en grupos de archivos. Los *grupos de archivos* son colecciones de archivos con nombre. SQL Server incluye un grupo de archivos como el predeterminado. Puede crear grupos de archivos adicionales con la instrucción CREATE DATABASE o ALTER DATABASE.

Con los grupos de archivos, puede colocar objetos específicos en un archivo específico. En la ilustración, los archivos OrdHist1.ndf y OrdHist2.ndf se encuentran en un disco diferente para separar los archivos que son muy consultados de aquéllos que son continuamente actualizados y así reducir los conflictos en las unidades de disco.

Los administradores de sistemas pueden hacer copias de seguridad y restaurar archivos o grupos de archivos individuales en lugar de hacer la copia de seguridad o la restauración de toda la base de datos. Realizar copias de seguridad de los archivos o grupos de archivos es necesario en bases de datos grandes para disponer de una estrategia eficaz de copia de seguridad y restauración.

Consideraciones acerca del uso de grupos de archivos

El uso de grupos de archivos es una técnica avanzada de diseño de bases de datos. Debe comprender en detalle la estructura de la base de datos, los datos, las transacciones y las consultas para determinar la mejor forma de colocar las tablas y los índices en grupos de archivos específicos. En muchos casos, el uso de la capacidad de creación de conjuntos de bandas de los sistemas RAID proporciona casi el mismo rendimiento que aquél que se puede obtener con los grupos de archivos, sin la carga administrativa que supone definirlos y administrarlos.

Sugerencia

Destaque que se puede crear una estructura de base de datos muy compleja. Los programas pueden verse más beneficiados por RAID que por los grupos de archivos.

Nota Los archivos de registro no forman parte de un grupo de archivos. El espacio de registro se administra de forma independiente del espacio de datos.

Tipos de grupos de archivos

SQL Server ofrece los dos tipos siguientes de grupos de archivos:

- El grupo de archivos principal, que contiene las tablas del sistema en el archivo de datos principal.
- Los grupos de archivos definidos por el usuario, que son los grupos de archivos que se especifican mediante la palabra clave FILEGROUP.

Designación del grupo de archivos predeterminado

Al crear una base de datos, el grupo de archivos principal se convierte automáticamente en el predeterminado. El grupo de archivos predeterminado recibe todas las nuevas tablas, índices y archivos que no tienen especificado un grupo de archivos. Si la base de datos contiene más de un grupo de archivos, se recomienda que el predeterminado sea uno de los grupos de archivos definidos por el usuario. Esto impide que el grupo de archivos principal, que contiene las tablas del sistema, se llene inesperadamente con una tabla de usuario.

Tamaño del grupo de archivos principal predeterminado

Si el grupo de archivos predeterminado se mantiene como grupo de archivos principal, es importante definir correctamente su tamaño. Si el grupo de archivos se queda sin espacio, no podrá agregar información nueva a las tablas del sistema. Si un grupo de archivos definido por el usuario se queda sin espacio, sólo se verán afectados los archivos de usuario que estén asignados específicamente a ese grupo de archivos.

Ejemplo

En el siguiente ejemplo se crea un grupo de archivos definido por el usuario en la base de datos **Northwind** y se agrega un archivo de datos secundario a ese grupo de archivos.

```
ALTER DATABASE Northwind
ADD FILEGROUP OrderHistoryGroup
GO
```

```
ALTER DATABASE Northwind
ADD FILE
( NAME = 'OrdHistYear1',
  FILENAME = 'c:\Archivos de programa\
    Microsoft SQL Server\MSSQL\Data\OrdHist1.ndf',
  SIZE = 5MB),
TO FILEGROUP OrderHistoryGroup
GO
```

Presentación de la información de los grupos de archivos

Puede encontrar información acerca de los grupos de archivos mediante funciones como FILE_NAME, FILE_ID, FILE_PROPERTY, FILEGROUP_NAME, FILEGROUP_ID y FILEGROUP_PROPERTY.

Los procedimientos almacenados del sistema de la tabla siguiente presentan también información acerca de los grupos de archivos.

Procedimiento almacenado de sistema	Descripción
-------------------------------------	-------------

sp_helpfile [[@filename =] 'nombre']	Devuelve los nombres físicos y los atributos de los archivos asociados con la base de datos actual. Utilice este procedimiento almacenado de sistema para determinar los nombres de los archivos que va a agregar o quitar del servidor.
--	--

sp_helpfilegroup [grupoDeArchivos]	Devuelve los nombres y los atributos de los grupos de archivos asociados con la base de datos actual.
--	---

◆ Administración de bases de datos

Objetivo del tema

Proporcionar un resumen de este tema.

Explicación previa

En esta sección aprenderá a administrar bases de datos y registros de transacciones.

- Administración del crecimiento de los archivos de datos y de registro
- Supervisión y ampliación de un registro de transacciones
- Reducción de una base de datos o un archivo
- Eliminación de una base de datos

Sugerencia

Pregunte a los alumnos si están familiarizados con los temas de esta sección. En caso afirmativo, puede impartir esta sección a un nivel alto u omitirla por completo.

A medida que la base de datos crece o cambia, usted puede ampliar o reducir su tamaño automáticamente o manualmente. Cuando ya no necesite una base de datos, puede eliminarla, junto con todos sus archivos asociados.

Administración del crecimiento de los archivos de datos y de registro

Objetivo del tema

Explicar cómo administrar el crecimiento de los archivos de datos y de registro.

Explicación previa

Hay varias formas de controlar el tamaño de una base de datos.

- Uso del crecimiento automático de los archivos
- Ampliación de los archivos de la base de datos
- Agregar archivos secundarios a la base de datos

```
ALTER DATABASE Sample
    MODIFY FILE ( NAME = 'SampleLog',
                  SIZE = 15MB)
GO

ALTER DATABASE Sample
    ADD FILE
        (NAME = SampleData2,
         FILENAME='c:\Archivos de programa\...\Data\Sample2.ndf',
         SIZE=15MB,
         MAXSIZE=20MB)
GO
```

Cuando los archivos de datos crecen o cuando aumenta la actividad de modificación de datos, puede que necesite ampliar el tamaño de los archivos de datos o de registro. Puede administrar el crecimiento de la base de datos mediante el Administrador corporativo de SQL Server o con la instrucción ALTER DATABASE. Para utilizar la instrucción ALTER DATABASE, debe encontrarse en la base de datos **master**.

Puede controlar el tamaño de la base de datos mediante las tareas siguientes:

- Configure los archivos de base de datos y de registro para que crezcan automáticamente.
- Aumente o disminuya manualmente el tamaño actual o el tamaño máximo de los archivos de base de datos y de registro existentes.
- Agregue manualmente archivos secundarios de base de datos y de registro.

Uso del crecimiento automático de los archivos

Puede configurar la opción de crecimiento automático de los archivos mediante la instrucción ALTER DATABASE o el Administrador corporativo de SQL Server para especificar que los archivos de la base de datos se expandan automáticamente en una cantidad especificada cuando sea necesario. El uso del crecimiento automático de los archivos reduce las tareas administrativas relacionadas con la ampliación manual del tamaño de la base de datos.

Puede especificar el tamaño inicial, el tamaño máximo y el incremento de crecimiento de cada archivo. Aunque es posible especificar el crecimiento de los archivos en megabytes o kilobytes, debe especificarlo como un porcentaje. Si no especifica un tamaño máximo, un archivo puede seguir creciendo hasta ocupar todo el espacio disponible en disco.

Si se emplea el crecimiento automático de archivos con varios archivos, SQL Server utiliza una estrategia de llenado proporcional en todos los archivos de cada grupo de archivos. A medida que los datos se escriben en el grupo de archivos, SQL Server escribe una cantidad proporcional al espacio libre en el archivo en cada archivo del grupo de archivos, en lugar de escribir todos los datos en el primer archivo hasta que está lleno y, luego, escribir en el siguiente archivo.

Para obtener un rendimiento óptimo:

- Asigne un tamaño inicial suficiente a la base de datos y al registro para evitar que se active con frecuencia la característica de crecimiento automático.
- Establezca un tamaño máximo para los archivos de datos en caso de que utilice varias bases de datos.
- Establezca los incrementos de crecimiento del archivo de datos y de registro con unos tamaños suficientes como para evitar que se active con frecuencia la característica de crecimiento automático.

Por ejemplo, si el registro crece 40 MB cada día, establezca el incremento de crecimiento automático en 50 MB o 100 MB, en lugar de en 1 MB.

Ampliación de los archivos de la base de datos

Incluso si no configura un archivo existente para que crezca automáticamente, aún puede aumentar su tamaño. Un valor de cero como incremento indica que no crece automáticamente.

Agregar archivos secundarios a la base de datos

Puede crear archivos secundarios en la base de datos para ampliar el tamaño de la misma. Utilice archivos secundarios en la base de datos para ubicar los archivos de datos en discos físicos distintos cuando no utilice la capacidad de creación de conjuntos de bandas en disco de los sistemas RAID.

Sintaxis parcial

```
ALTER DATABASE baseDeDatos
{ ADD FILE < filespec > [ ,...n ] [ TO FILEGROUP
nombreGrupoDeArchivos ]
| ADD LOG FILE < filespec > [ ,...n ]
| REMOVE FILE nombreArchivoLógico [ WITH DELETE ]
| ADD FILEGROUP nombreGrupoDeArchivos
| REMOVE FILEGROUP nombreGrupoDeArchivos
| MODIFY FILE < filespec >
| MODIFY NAME = nuevoNombreBaseDatos
| MODIFY FILEGROUP nombreGrupoDeArchivos
  {propiedadGrupoDeArchivos | NAME = nuevoNombreGrupoDeArchivos }
| SET < optionspec > [ ,...n ] [ WITH < termination > ]
| COLLATE < nombreIntercalación >
}
```

Ejemplo

El siguiente ejemplo aumenta el tamaño actual del registro y agrega un archivo de datos secundario a la base de datos **Sample**.

```
ALTER DATABASE Sample
    MODIFY FILE ( NAME = 'SampleLog',
        SIZE = 15MB)
GO
```

```
ALTER DATABASE Sample
ADD FILE
(NAME = 'SampleData2' ,
FILENAME='c:\Archivos de programa\
    Microsoft SQL Server\MSSQL\Data\Sample2.ndf',
SIZE=15MB ,
MAXSIZE=20MB)
GO
```

Supervisión y ampliación de un registro de transacciones

Objetivo del tema

Mostrar cómo supervisar y expandir el espacio del registro de transacciones.

Explicación previa

Diseñe cuidadosamente el tamaño del registro para que haya espacio suficiente.

- **Supervisión del registro**
- **Supervisión de las situaciones que causan una actividad intensa en el registro**
 - Carga masiva de datos en la tabla indizada
 - Grandes transacciones
 - Realizar operaciones de imagen o texto registradas
- **Ampliación del registro cuando es necesario**

Cuando una base de datos crece o cuando aumenta la actividad de modificación de los datos, puede que necesite ampliar el registro de transacciones.

Supervisión del registro

Diseñe cuidadosamente el tamaño del registro para no quedarse sin espacio. Si supervisa el registro con regularidad, podrá determinar el momento óptimo para expandirlo.

Advertencia Si el registro de transacciones se queda sin espacio, SQL Server no puede registrar las transacciones y no permite hacer cambios a la base de datos.

Sugerencia

Muestre cómo supervisar el registro con el Monitor de sistema.

Puede supervisar el registro de transacciones con el Administrador corporativo de SQL Server, la instrucción DBCC SQLPERF (LOGSPACE) o con el Monitor de sistema de Microsoft Windows® 2000.

Puede supervisar los registros de transacciones de bases de datos individuales mediante los contadores del objeto SQL Server:Database del Monitor de sistema. Estos contadores incluyen los enumerados en la tabla siguiente.

Contador de objeto	Presenta
Bytes de registro vaciados/seg.	Número de bytes en el búfer del registro cuando se vacía el búfer.
Vaciados del registro/seg.	Número de vaciados del registro.
Esperas al vaciar el registro/seg.	Número de confirmaciones que esperan el vaciado del registro.
Porcentaje utilizado del registro	Porcentaje de espacio del registro en uso.
Tamaño de los archivos de registro (KB)	Tamaño acumulativo de todos los archivos de registro de la base de datos.
Proporción de aciertos de caché del registro	Porcentaje de lecturas de la caché del registro que se leyeron correctamente en la caché del registro.

Supervisión de las situaciones que causan una actividad intensa en el registro

Algunas situaciones que causan actividad adicional en el registro de transacciones son:

- Cargar información en una tabla que tiene índices. SQL Server registra todas las inserciones y cambios en los índices. Al cargar tablas sin índices, SQL Server sólo registra las asignaciones de extensiones.
- Las transacciones que realizan muchas modificaciones (instrucciones INSERT, UPDATE y DELETE) en una tabla dentro de una misma transacción. Esto suele ocurrir cuando la instrucción no incluye una cláusula WHERE o cuando la cláusula WHERE es demasiado general, lo que hace que afecte a un gran número de registros.
- Agregar o modificar datos de texto o imagen en una tabla.

Ampliación del registro cuando es necesario

El registro de transacciones se puede ampliar con el Administrador corporativo de SQL Server o mediante la instrucción ALTER DATABASE.

Reducción de una base de datos o un archivo

Objetivo del tema

Mostrar cómo se reduce una base de datos.

Explicación previa

Puede reducir toda la base de datos o archivos específicos de la base de datos.

- Reducción de toda la base de datos

```
DBCC SHRINKDATABASE (Sample, 25)
```

- Reducción de un archivo de base de datos

```
DBCC SHRINKFILE (Sample_Data, 10)
```

- Reducción automática de una base de datos

Establecer la opción de base de datos **autoshrink** a True

Cuando hay demasiado espacio asignado o cuando los requisitos de espacio disminuyen, puede reducir toda la base de datos o archivos de datos específicos de la base de datos.

Reducción de toda la base de datos

Para reducir toda la base de datos, puede utilizar el Administrador corporativo de SQL Server o ejecutar la instrucción SHRINKDATABASE de Comprobador de coherencia de base de datos (DBCC, *Database Consistency Checker*). Esto reduce el tamaño de todos los archivos de datos de la base de datos.

SQL Server reduce los archivos de registro mediante una operación de reducción diferida y lo hace como si todos los archivos de registro fueran uno solo. Los archivos de registro se restablecen cuando el registro se trunca; SQL Server intenta reducir los archivos de registro truncados tanto como sea posible, hasta alcanzar el tamaño de destino.

Sintaxis

DBCC SHRINKDATABASE (*baseDeDatos* [, *porcentajeDestino*] [, {NOTRUNCATE | TRUNCATEONLY}])

La siguiente tabla describe las opciones de DBCC SHRINKDATABASE.

Opción	Descripción
<i>porcentajeDestino</i>	Especifica el porcentaje de espacio libre que quedará en el archivo de la base de datos una vez que SQL Server haya reducido la base de datos.
NOTRUNCATE	Hace que SQL Server mantenga el espacio liberado en los archivos de la base de datos. La acción predeterminada es pasar el espacio liberado de los archivos al sistema operativo.
TRUNCATEONLY	Hace que el espacio no utilizado en los archivos de datos pase al sistema operativo y reduce el archivo hasta la última extensión asignada, con lo que disminuye el tamaño del archivo sin mover ningún dato. No se vuelven a asignar filas a páginas no asignadas. Cuando se utiliza esta opción, SQL Server omite <i>porcentajeDestino</i> .

Ejemplo

Este ejemplo reduce el tamaño del archivo **SampleData** para que quede con un 25 por ciento de espacio libre.

```
DBCC SHRINKDATABASE (SampleData, 25)
```

En el ejemplo anterior, si el archivo de la base de datos **Sample** contiene 6 MB de datos, el nuevo tamaño de la base de datos será de 8 MB (6 MB de datos, 2 MB de espacio libre).

Nota SQL Server no reduce un archivo a un tamaño menor que la cantidad de espacio ocupada por los datos. Además, no reduce los archivos a un tamaño menor del especificado en el parámetro SIZE de la instrucción CREATE DATABASE.

Reducción de un archivo de base de datos

El tamaño de un archivo de datos de una base de datos se puede reducir mediante el Administrador corporativo de SQL Server o al ejecutar la instrucción DBCC SHRINKFILE.

Sintaxis

DBCC SHRINKFILE ({*archivo* | *IdArchivo* } [, *tamañoDestino*] [, {EMPTYFILE | NOTRUNCATE | TRUNCATEONLY}])

La siguiente tabla describe las opciones de DBCC SHRINKFILE.

Opción	Descripción
<i>tamañoDestino</i>	Especifica el tamaño deseado para el archivo de datos, en megabytes, expresado como número entero. Si no se especifica, DBCC SHRINKFILE reduce el tamaño tanto como sea posible.
EMPTYFILE	Migra todos los datos desde el archivo especificado a otros archivos del mismo grupo de archivos. SQL Server ya no permite colocar datos en los archivos en los que se ha utilizado la opción EMPTY_FILE. Utilice esta opción para eliminar el archivo mediante la instrucción ALTER DATABASE.

Ejemplo

Este ejemplo reduce el tamaño del archivo de datos **sample** a 10 MB.

DBCC SHRINKFILE (Sample, 10)

Reducción automática de una base de datos

La reducción automática no está habilitada de forma predeterminada. Si la opción de base de datos **autoshrink** se establece en True, puede definir una opción de base de datos para recuperar automáticamente el espacio no utilizado. También puede cambiar esta opción con el Administrador corporativo de SQL Server.

Al reducir una base de datos o un archivo de datos, tenga en cuenta los siguientes hechos y directrices:

- La base de datos resultante debe ser mayor que el tamaño de la base de datos **model** o que los datos existentes en la base de datos o el archivo de datos.
- Antes de reducir una base de datos o un archivo de datos, debe hacer una copia de seguridad de la base de datos y de la base de datos **master**.
- DBCC SHRINKDATABASE y SHRINKFILE se ejecutan de forma diferida; por lo tanto, es posible que no vea reducirse inmediatamente el tamaño de los archivos o de la base de datos.
- DBCC SHRINKFILE le permite reducir la base de datos a un tamaño inferior al especificado cuando se creó o se modificó la base de datos, pero nunca a un tamaño inferior al que ocupan los datos.

Eliminación de una base de datos

Objetivo del tema

Ilustrar cómo se elimina una base de datos.

Explicación previa

Elimine una base de datos sólo cuando esté seguro de que ya no la necesita.

■ Métodos de eliminación de una base de datos

- Administrador corporativo de SQL Server
- Instrucción DROP DATABASE

```
DROP DATABASE Northwind, pubs
```

■ Restricciones de la eliminación de bases de datos

- Mientras se está restaurando
- Cuando se conecta a ella un usuario
- Cuando se publica como parte de la duplicación
- Si es una base de datos de sistema

Puede eliminar una base de datos cuando ya no la necesite. La eliminación de una base de datos quita la base de datos y los archivos de disco que utiliza la base de datos.

Métodos de eliminación de una base de datos

Para eliminar bases de datos, puede utilizar el Administrador corporativo de SQL Server o ejecutar la instrucción DROP DATABASE.

Sintaxis

```
DROP DATABASE baseDeDatos [...n]
```

Ejemplo

En este ejemplo, se eliminan varias bases de datos con una sola instrucción.

```
DROP DATABASE Northwind, pubs
```

Al eliminar una base de datos, tenga en cuenta los siguientes hechos y directrices:

- Con el Administrador corporativo de SQL Server, sólo puede eliminar una base de datos a la vez.
- Con Transact-SQL, puede eliminar varias bases de datos a la vez.
- Después de eliminar una base de datos, todos los Id. de inicio de sesión que utilizaban esa base de datos como base de datos predeterminada ya no tendrán una base de datos predeterminada.

Nota Haga una copia de seguridad de la base de datos **master** después de eliminar una base de datos.

Sugerencia

Mencione que SQL Server no permite eliminar las bases de datos **master**, **model** y **tempdb**, pero permite eliminar la base de datos del sistema **msdb**.

Restricciones de la eliminación de bases de datos

Las siguientes restricciones se aplican a la eliminación de bases de datos.

No se puede eliminar:

- Una base de datos que esté en proceso de restauración.
- Una base de datos que un usuario haya abierto para leer o escribir en ella.
- Una base de datos que esté publicando alguna de sus tablas como parte de la duplicación de SQL Server.
- Una base de datos de sistema.

◆ Introducción a las estructuras de datos

Objetivo del tema

Presentar las estructuras de datos.

Explicación previa

En esta sección aprenderá acerca de las estructuras de datos de SQL Server.

- Cómo se almacenan los datos
- Tipos de páginas y extensiones
- Páginas que administran el espacio de los archivos
- Páginas que realizan el seguimiento de tablas e índices

Esta sección describe las estructuras de datos que SQL Server utiliza para almacenar datos.

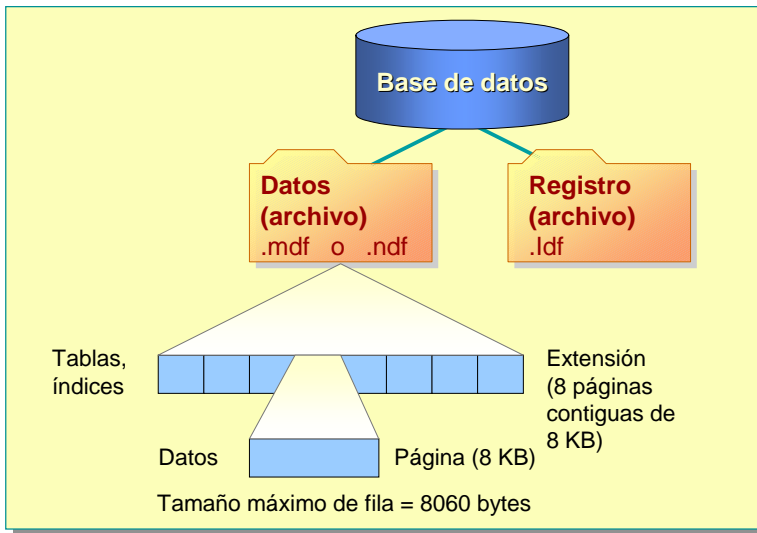
Cómo se almacenan los datos

Objetivo del tema

Describir cómo está estructurada la base de datos.

Explicación previa

Al crear una base de datos, es importante comprender cómo SQL Server almacena los datos.



Al crear una base de datos, es importante comprender cómo SQL Server almacena los datos para poder calcular y especificar la cantidad de espacio de disco que hay que asignar a la base de datos. Tenga en cuenta los siguientes hechos y directrices acerca del almacenamiento de los datos:

- Todas las bases de datos tienen un archivo de datos principal, que se identifica mediante la extensión de nombre de archivo .mdf, y uno o varios archivos de registro de transacciones, que se identifican mediante la extensión de nombre de archivo .ldf. Una base de datos puede tener también archivos de datos secundarios, que se identifican mediante la extensión de nombre de archivo .ndf. Estos archivos físicos tienen nombres del sistema operativo y nombres de archivo lógicos que se pueden utilizar en instrucciones Transact-SQL.
- Cuando se crea una base de datos, se hace una copia de la base de datos **model**, que incluye las tablas del sistema. El tamaño mínimo de una base de datos debe ser igual o mayor que el tamaño de la base de datos **model**.
- SQL Server almacena, lee y escribe datos en bloques de 8 kilobytes (KB) de espacio contiguo en disco que se denominan *páginas*. Esto significa que una base de datos puede almacenar 128 páginas por megabyte.
- Las filas no pueden abarcar más de una página. Por tanto, la máxima cantidad de datos de una fila, quitando el espacio necesario para la cabecera de la fila, es de 8060 bytes.
- Todas las páginas se almacenan en *extensiones*. Una extensión consta de ocho páginas contiguas, o 64 KB. Por tanto, una base de datos tiene 16 extensiones por megabyte.
- Los archivos de los registros de transacciones contienen toda la información necesaria para recuperar la base de datos en caso de un error del sistema. De forma predeterminada, el tamaño del registro de transacciones es el 25 por ciento del tamaño de los archivos de datos. Utilice esta proporción como punto de partida y ajústela según las necesidades de la aplicación.

Sugerencia

Al diseñar una base de datos o la capacidad de la misma, es importante conocer el tamaño máximo de las filas y extensiones.

Por ejemplo, como las filas no pueden ser más grandes que una página, una fila cuyo tamaño sea 4035 bytes puede caber en una página de 8060 bytes.

Tipos de páginas y extensiones

Objetivo del tema

Presentar los tipos de páginas y extensiones.

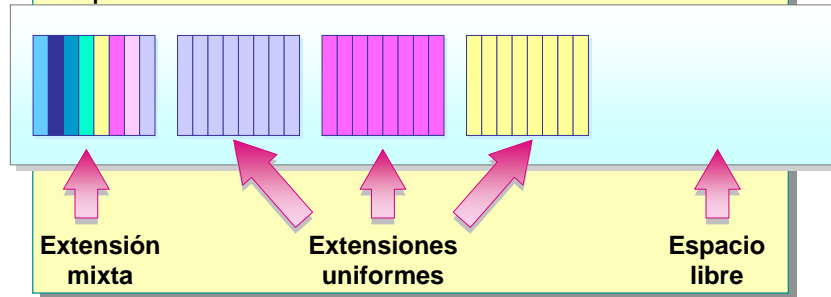
Explicación previa

La mayor parte de la asignación en la base de datos se produce en bloques de ocho páginas llamados extensiones.

■ Tipos de páginas

- Páginas que realizan el seguimiento de la asignación de espacio
- Páginas que contienen datos de índice y usuario

■ Tipos de extensiones



Las páginas y las extensiones constituyen las estructuras de datos principales en la base de datos física de SQL Server.

Tipos de páginas

SQL Server utiliza varios tipos de páginas: algunas realizan el seguimiento de la asignación de espacio y otras contienen datos del usuario y de los índices. Las páginas que realizan el seguimiento de las asignaciones contienen información densamente empaquetada. Esto permite a SQL Server mantenerlas de forma eficiente en memoria para su fácil seguimiento.

Tipos de extensiones

SQL Server utiliza dos tipos de extensiones:

- Las extensiones que contienen páginas de dos o más objetos se llaman *extensiones mixtas*. Cada tabla se inicia como una extensión mixta. Las extensiones mixtas se utilizan principalmente en páginas que realizan el seguimiento del espacio y contienen objetos pequeños.
- Las extensiones que tienen ocho páginas asignadas a un solo objeto se llaman *extensiones uniformes*. Se utilizan cuando las tablas o los índices requieren más de 64 KB de espacio.

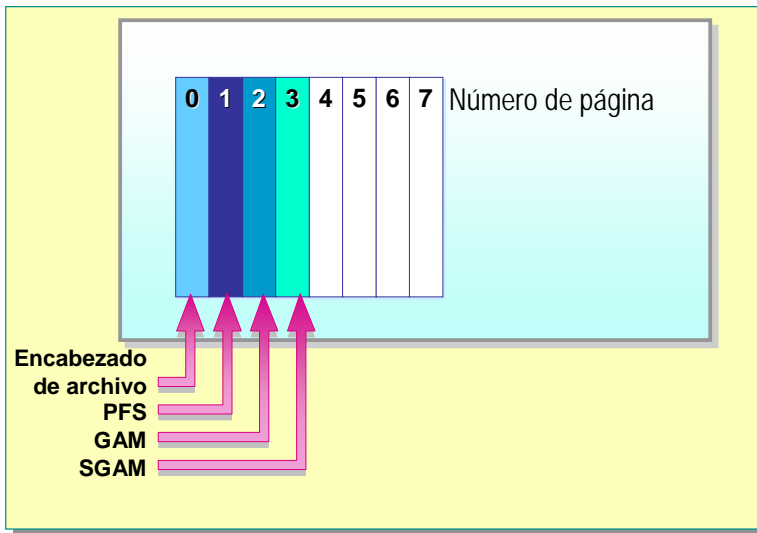
Páginas que administran el espacio de los archivos

Objetivo del tema

Describir las páginas que administran el espacio de los archivos.

Explicación previa

Las primeras cuatro páginas de cada archivo indican a SQL Server las páginas que están en uso.

**Sugerencia**

Haga énfasis en los siguientes puntos: (1) La página PFS realiza el seguimiento de cada página. (2) Las páginas GAM y SGAM realizan el seguimiento de las extensiones, no de las páginas. (3) Mediante las páginas GAM y SGAM, SQL Server puede buscar espacio cuando se necesitan nuevas páginas. (4) Las páginas PFS, GAM y SGAM se pueden repetir tanto como sea necesario en archivos de gran tamaño.

La primera extensión de cada archivo es una extensión mixta que contiene una página de encabezado de archivo seguida de tres páginas de asignación. SQL Server asigna esta extensión mixta cuando se crea el archivo de datos principal y utiliza estas páginas internamente.

Página de encabezado de archivo

La página de encabezado de archivo contiene los atributos del archivo, como el nombre de la base de datos que posee el archivo, su grupo de archivos, el tamaño mínimo y el incremento de crecimiento. Es la primera página de cada archivo (página 0).

Página PFS

La página de Espacio libre de página (PFS, *Page Free Space*) es una página de asignación que contiene información acerca del espacio libre disponible en las páginas de un archivo. La página 1 de cada archivo es una página PFS. SQL Server agrega otras páginas PFS cuando es necesario.

Cada página PFS puede realizar el seguimiento de 8.000 páginas contiguas, lo que equivale a casi 64 MB de datos. Por cada página, la página PFS contiene un byte que realiza el seguimiento de lo siguiente:

- Si la página ha sido asignada.
- Si la página está en una extensión mixta o uniforme.
- Una aproximación de la cantidad de espacio disponible en la página.

Páginas GAM y SGAM

SQL Server utiliza páginas de Mapa de asignación global (GAM, *Global Allocation Map*) y Mapa de asignación global secundaria (SGAM, *Secondary Global Allocation Map*) para determinar la ubicación de extensiones libres o extensiones mixtas con páginas libres.

Páginas GAM

La página GAM es una página de asignación que contiene información acerca de las extensiones asignadas. La página 2 de cada archivo es una página GAM. SQL Server agrega otras páginas GAM cuando es necesario.

Cada página GAM cubre 63.904 extensiones, o, lo que es lo mismo, casi 4 gigabytes (GB) de datos. La página GAM contiene un bit por cada extensión que abarca. El bit se establece en 0 si la extensión está asignada y en 1 si está libre.

Páginas SGAM

La página SGAM es una página de asignación que contiene información acerca de las extensiones mixtas asignadas. La página 3 de cada archivo es una página SGAM. SQL Server agrega otras páginas SGAM cuando es necesario.

Las páginas SGAM realizan el seguimiento de las extensiones mixtas que tienen actualmente como mínimo una página sin utilizar. También cubren 63.904 extensiones. Un bit establecido en 0 indica que una extensión es uniforme o mixta sin páginas libres. Un bit establecido en 1 indica una extensión mixta con una o más páginas libres.

La siguiente tabla resume los valores de los bits de las páginas GAM y SGAM:

Si la página GAM está establecida en...	Y la página SGAM está establecida en...	Entonces...
1	0	Es una extensión disponible. Esta extensión no está en uso.
0	1	Es una página disponible. Esta extensión mixta tiene una o varias páginas sin asignar.
0	0	Es una extensión que no tiene nada disponible. La extensión se asigna como una extensión uniforme o como una extensión mixta llena.

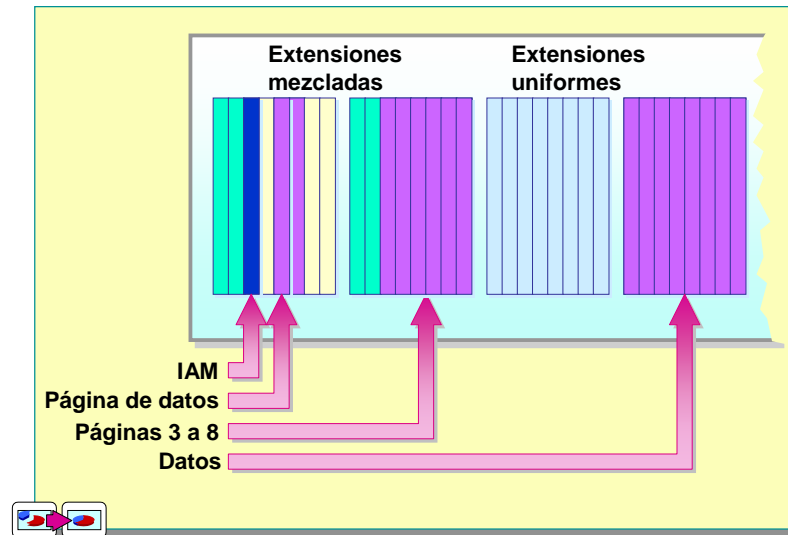
Páginas que realizan el seguimiento de tablas e índices

Objetivo del tema

Examinar la función de las páginas IAM.

Explicación previa

Las páginas IAM realizan el seguimiento de la relación entre un objeto y las extensiones o páginas que utiliza.



Sugerencia

Ésta es una diapositiva animada. Consulte las Notas para el instructor si necesita ayuda para desplazarse por esta diapositiva.

Haga énfasis en que cada tabla e índice tiene su propia página IAM. SQL Server puede utilizar la página IAM para buscar las extensiones y páginas asignadas al objeto.

SQL Server asigna inicialmente a cada tabla e índice una página de asignación y, al menos, una página de datos de una extensión mixta. A medida que el objeto crece, SQL Server asigna hasta siete páginas más de extensiones mixtas cuando es necesario. Cuando el objeto supera ocho páginas, SQL Server asigna páginas adicionales de las extensiones uniformes.

SQL Server utiliza cuatro tipos de páginas para administrar tablas e índices. Pueden aparecer en cualquier lugar del archivo. Se trata de las páginas IAM, Datos, Texto/Imagen e Índice.

Página IAM

La página IAM es una página de asignación que contiene información acerca de las extensiones que utiliza una tabla o índice.

La página IAM contiene la ubicación de las ocho páginas iniciales y un mapa de bits de extensiones que indica las extensiones que están en uso para ese objeto. Una sola página IAM puede realizar el seguimiento de hasta 512.000 páginas de datos. SQL Server agrega más páginas IAM cuando las tablas son grandes.

Las páginas IAM siempre se asignan a partir de extensiones mixtas y pueden aparecer en cualquier parte de un archivo o grupo de archivos. SQL Server intenta agrupar las páginas IAM para que la recuperación sea más rápida.

Página Datos

La página Datos incluye contenido que no es del tipo de datos **text**, **ntext** e **image**.

Página Texto/Imagen

La página Texto/Imagen incluye contenido del tipo de datos **text**, **ntext** e **image**.

Página Índice

La página Índice contiene estructuras de índice.