



© JMA 2009. All rights reserved

Contenidos

1. Reportes SQL de valor (SSRS) Depende de sus bases de datos
2. Optimización de la infraestructura
 - a) Consultas T-SQL Depende de sus bases de datos
 - b) Query Store
 - c) Extended Events
 - d) DMVs
 - e) Planes de Ejecución
3. Mantenimiento SQL SERVER
 - a) Planes de Mantenimiento
 - b) Tareas programadas
 - c) Procesos SQL Server
4. Mantenimiento de índices SQL Server
 - a) Fragmentación índices
 - b) Mantenimiento
 - c) Optimización
5. Conocer todos los servicios que presta SQL

© JMA 2009. All rights reserved

ARQUITECTURA FÍSICA

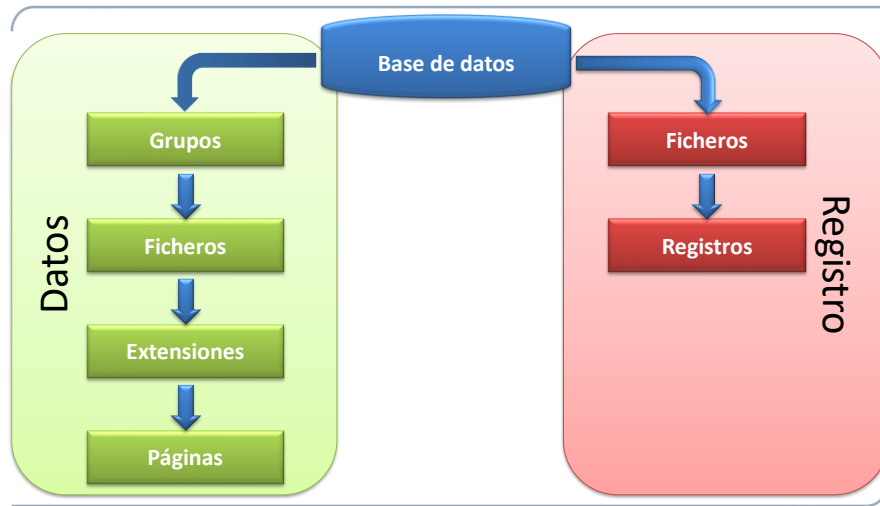
© JMA 2016. All rights reserved

Composición de un BBDD

- Desde el punto de vista lógico
 - Compuesta por objetos
 - Desde el punto de vista físico
 - Páginas, extensiones y ficheros
 - Dos tipos de ficheros
 - Datos
 - Contiene los objetos de la Base de Datos
 - Registro
 - Histórico de modificaciones realizadas en la Base de Datos
 - Sirve para garantizar la integridad de la Base de Datos
-

© JMA 2016. All rights reserved

Estructura Física



© JMA 2016. All rights reserved

Páginas

- La página es la unidad fundamental de asignación de almacenamiento de datos en SQL Server.
- El tamaño de página es de 8 KB (8192 bytes) y no se puede cambiar. Esto significa que las bases de datos de SQL Server tienen 128 páginas por megabyte.
- Cada página empieza con un encabezado de 96 bytes, que se utiliza para almacenar la información del sistema acerca de la página.
- Esta información incluye el número de página, el tipo de página, el espacio disponible en la página y el Id. de unidad de asignación del objeto propietario de la página.

© JMA 2016. All rights reserved

Tipos de Páginas

- **Datos:** Filas de datos con todos los datos excepto los datos text, ntext e image
- **Índice:** Entradas de índices
- **Texto o imagen** (LOB_DATA, ROW_OVERFLOW_DATA): Tipos de datos de objetos grandes (LOB): Datos text, ntext e image, opción text in row
- **Espacio libre en páginas** (PFS): Información acerca del espacio libre disponible en las páginas
- **Mapa de asignación de índices** (IAM): Información acerca de las extensiones utilizadas por una tabla o un índice
- **Mapa de asignación global** (GAM), **Mapa de asignación global secundaria** (SGAM): Información acerca de las extensiones asignadas
- **Mapa cambiado masivamente** (BCM): Información acerca de las extensiones modificadas por operaciones masivas desde la última instrucción BACKUP LOG
- **Mapa cambiado diferencial** (DCM): Información acerca de las extensiones que han cambiado desde la última instrucción BACKUP DATABASE

© JMA 2016. All rights reserved

Tipos de longitudes de datos

- **Longitud predefinida**
 - El dato siempre ocupa lo mismo en función al tipo de dato elegido.
- **Longitud fija**
 - El dato siempre ocupa lo mismo en función al tipo de dato elegido y la longitud definida.
- **Longitud variable**
 - La ocupación viene determinada por la longitud del dato y el tipo, se almacena en paginas de datos.
- **Longitud variable – LOB**
 - La ocupación viene determinada por la longitud del dato y el tipo, se almacena en paginas LOB.
- **Longitud variable – MAX**
 - La ocupación viene determinada por la longitud del dato y el tipo, se almacena en paginas de datos o paginas LOB en función del espacio libre.

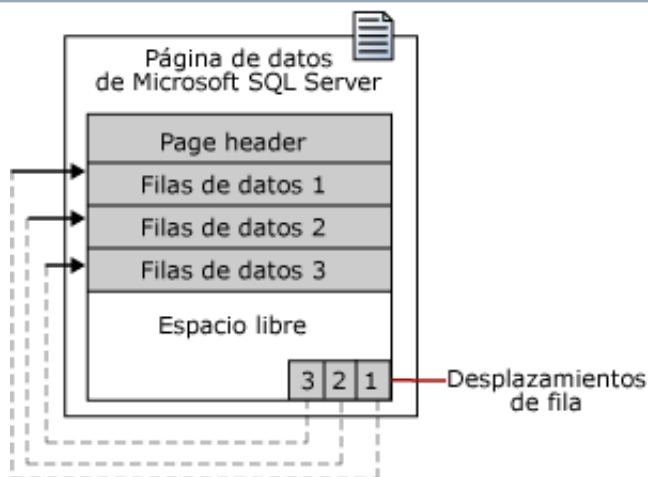
© JMA 2016. All rights reserved

Paginas de Datos

- La página está vinculada a un único objeto.
- La página puede contener varias filas.
- Una fila tiene que estar contenida en una página (más las páginas adicionales para las columnas LOB con un puntero de texto de 16 bytes).
- Las filas de datos son colocadas en posiciones consecutivas, de tal forma que una nueva fila pueda ocupar todo el espacio libre de la página.
- De cada página se pueden aprovechar 8060 bytes para almacenar datos.
- Al final de la página, comienza una tabla de desplazamiento de fila que contiene una entrada para cada fila de la página y registra la distancia del primer byte de la fila desde el inicio de la página. Las entradas en la tabla ocupan 2 bytes y están en orden inverso a la secuencia de las filas de la página.

© JMA 2016. All rights reserved

Paginas de Datos



© JMA 2016. All rights reserved

Paginas LOB

- Los datos LOB se guardan en una cadena enlazada de páginas LOB: divide el dato en bloques de 8 Kb y asigna cada bloque a una página LOB_DATA.
- El enlazamiento determina un máximo de 2^{18} paginas por lo el tamaño máximo de un dato LOB es de 2 Gb.
- Se requiere una cadena LOB para cada celda, columna de la fila, no nula.
- Una página LOB solo contiene datos de una única celda.
- Los datos LOB no nulos ocupan la menos una página LOB, mas el puntero de 16 bytes almacenado a nivel de fila en la página de datos.

© JMA 2016. All rights reserved

Optimización del espacio

- text in row:
 - Al activarlo, los datos LOB se guardan en la página de datos si tienen espacio.
 - Se puede fijar el limite entre 24 hasta 7000.
 - Puede provocar problemas de rendimiento.
- large value types out of row:
 - Activa el desbordamiento de página, cuando la fila no cabe en la página trata la columnas de longitud variable como si fueran LOB.
 - Mueve dinámicamente una o más columnas de longitud variable a páginas de la unidad de asignación ROW_OVERFLOW_DATA, empezando por la columna con el mayor ancho.
 - La longitud máxima de columna sigue siendo 8000, es la combinación de columnas la que la puede superar.
 - Puede empeorar el rendimiento de consultas y ordenaciones, suele ser preferible partición vertical.
- Estas opciones se configuran con el procedimiento almacenado: `sp_tableoption`

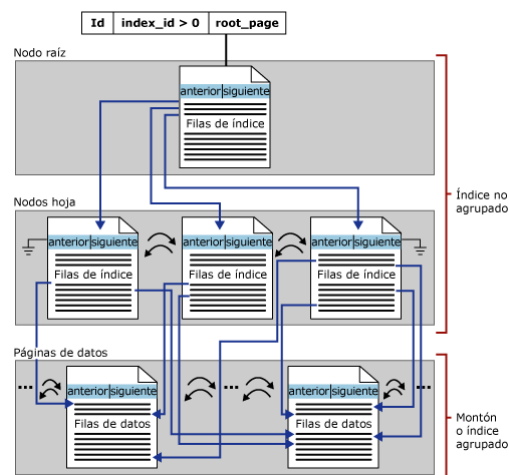
© JMA 2016. All rights reserved

Paginas de Índices

- Un índice es una estructura de disco asociada con una tabla o una vista que acelera la recuperación de filas de la tabla o de la vista.
- Un índice contiene claves generadas a partir de una o varias columnas ordenadas ascendente o descendentemente de la tabla o la vista.
- Dichas claves están almacenadas en una estructura (árbol b) que permite la búsqueda de forma rápida y eficiente de la fila o filas asociadas a los valores de cada clave.
- Es una estructura que maximiza la anchura y minimiza la profundidad.

© JMA 2016. All rights reserved

Paginas de Índices



© JMA 2016. All rights reserved

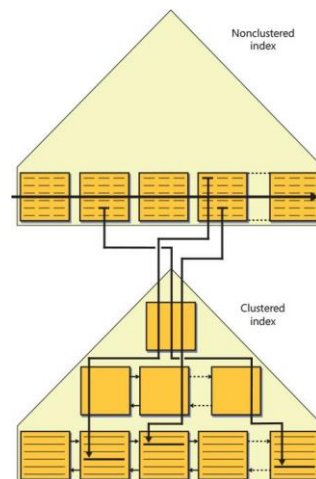
Paginas de Índices

- Están compuestas por registros de indexación, uno por fila de la tabla.
- Cada Registro de indexación contiene:
 - Clave de indexación:
 - compuesta por los valores de las columnas para la fila
 - como máximo de 900 bytes y 16 columnas
 - Localizador de fila:
 - Sin índice agrupado: RowID, el localizador es un puntero que se genera a partir del identificador (Id.) de archivo, el número de página y el número de la fila dentro de la página (8 bytes).
 - Con índice agrupado: es el valor de la clave del índice agrupado para la fila.
 - Columnas Incluidas.

© JMA 2016. All rights reserved

Tipos de índices: Agrupado

- Los índices agrupados ordenan y almacenan las filas de los datos de la tabla o vista de acuerdo con los valores de la clave del índice.
- Son columnas incluidas en la definición del índice.
- Sólo puede haber un índice clúster por cada tabla, porque las filas de datos sólo pueden estar ordenadas de una forma.
- La única ocasión en la que las filas de datos de una tabla están ordenadas es cuando la tabla contiene un índice clúster.
- Cuando una tabla tiene un índice clúster, la tabla se denomina tabla agrupada.
- Si una tabla no tiene un índice clúster, sus filas de datos están almacenadas en una estructura sin ordenar denominada montón.
- Puede ser único, garantiza que la clave de índice no contenga valores duplicados y, por tanto, cada fila de la tabla o vista es en cierta forma única.



© JMA 2016. All rights reserved

Tipos de índices: No agrupado

- Los índices no agrupados tienen una estructura separada de las filas de datos. Se pueden establecer como únicos.
- Un índice no agrupado contiene los valores de clave de índice no agrupado y cada entrada de valor de clave tiene un puntero a la fila de datos que contiene el valor clave.
- El puntero de una fila de índice no agrupado hacia una fila de datos se denomina localizador de fila.
- La estructura del localizador de filas depende de si las páginas de datos están almacenadas en un montón o en una tabla agrupada.
- Si están en un montón, el localizador de filas es un puntero hacia la fila.
- Si están en una tabla agrupada, el localizador de fila es la clave de índice agrupada.
- Se puede incluir columnas sin clave al nivel hoja de un índice no agrupado con el fin de eludir los límites existentes para las claves de índice, 900 bytes y columnas de 16 claves, así como para ejecutar consultas indizadas y totalmente cubiertas.

© JMA 2016. All rights reserved

Filtrado de índices

- Un índice filtrado es un índice no clúster optimizado, especialmente indicado para atender consultas que realizan selecciones a partir un subconjunto bien definido de datos.
- Utiliza un predicado de filtro para indizar una parte de las filas de la tabla.
- Un índice filtrado bien diseñado puede, en relación con los índices de tabla completa:
 - mejorar el rendimiento de las consultas y mayor calidad del plan
 - reducir los costos de mantenimiento
 - reducir el coste de almacenamiento del índice.
- Para diseñar índices filtrados efectivos, es importante entender qué consultas utiliza la aplicación y cómo se relacionan con los subconjuntos de datos.

© JMA 2016. All rights reserved

Otros tipos de índices

- Espacial
 - Un índice espacial proporciona la capacidad de realizar de forma más eficaz determinadas operaciones en objetos espaciales (datos espaciales) en una columna del tipo de datos geometry.
 - El índice espacial reduce el número de objetos a los que es necesario aplicar las operaciones espaciales, que son relativamente costosas.
- XML
 - Representación dividida y permanente de los objetos XML binarios grandes (BLOB) de la columna de tipo de datos xml.
- Texto completo
 - Tipo especial de índice funcional basado en símbolos (token) que crea y mantiene el motor de texto completo de Microsoft para SQL Server.
 - Proporciona la compatibilidad adecuada para búsquedas de texto complejas en datos de cadenas de caracteres.
 - Se almacenan fuera de los ficheros de la base de datos.

© JMA 2016. All rights reserved

Factor de relleno y fragmentación

- El factor de relleno determina la densidad de ocupación de páginas por parte del índice y fija de llenado de las paginas al crear el índice.
- La fragmentación de índices ocurre cuando los índices tienen páginas en las que la ordenación lógica, basada en el valor de clave, no coincide con la ordenación física dentro del archivo de datos.
- La fragmentación de índices provoca un desequilibrio en el llenado de las paginas de índice.
- Los índices muy fragmentados pueden reducir el rendimiento de la consulta y ralentizar la respuesta de la aplicación.
- Con la función del sistema **sys.dm_db_index_physical_stats** se podrá detectar la fragmentación de un índice específico, de todos los índices de una tabla o vista indizada, de todos los índices de una base de datos o de todos los índices de todas las bases de datos.
- Se puede solucionar la fragmentación del índice reorganizándolo o volviéndolo a generar.

© JMA 2016. All rights reserved

Número de Índices

- El número de índices afecta al rendimiento y ocupación de la base de datos.
- Es necesario mantener el número de índices al mínimo estrictamente necesario.
- Las restricciones PRIMARY KEY crean índices de forma automática de tipo UNIQUE. Por defecto como agrupados.
- Las restricciones UNIQUE crean índices UNIQUE de forma automática.
- Como máximo se puede tener:
 - Un índice agrupado por tabla.
 - 249 índices no agrupados (incluidos los creados por restricciones PRIMARY KEY o UNIQUE).

© JMA 2016. All rights reserved

Recomendaciones

- Indexar:
 - Tablas con muchas páginas (no necesariamente filas)
 - Columnas con un amplio rango de valores
 - Columnas que se usan: en consultas, agrupaciones, funciones de agregación y ordenaciones.
 - Columnas usadas en uniones
- NO Indexar:
 - Tablas que ocupen poco (pocas páginas)
 - Tablas con muchas modificaciones pero pocas consultas.
 - Columnas de gran tamaño
- El Asistente para optimización de índices suministra recomendaciones en función a la monitorización del sistema.

© JMA 2016. All rights reserved

Realizar operaciones de índice en línea

- Por defecto, las operaciones generar o volver a generar un índice de mantienen bloqueos exclusivos de los datos subyacentes y los índices asociados.
- Gracias a la opción ONLINE, es posible que usuarios simultáneos obtengan acceso a los datos de la tabla subyacente o del índice, así como a los índices no clúster asociados durante las operaciones de desfragmentación del índice.
- Para volver a generar un índice en línea:
 - Seleccionar Índice>Propiedades>Opciones:
 - Permitir procesamiento DML en línea = True
 - ALTER INDEX ... REBUILD WITH (ONLINE = ON);

© JMA 2016. All rights reserved

Directrices para operaciones en línea

- Se recomienda realizar operaciones de índices en línea en entornos empresariales que funcionan 24 horas al día, siete días a la semana, y en los que resulta fundamental la actividad simultánea de los usuarios durante las operaciones de índices.
- La tabla subyacente no se puede modificar (DDL), truncar o quitar mientras se está llevando a cabo una operación de índice en línea.
- Normalmente, las operaciones de índice en línea son más lentas que las operaciones de índice sin conexión equivalentes, independientemente del nivel de actividad de actualización simultánea.
- Como las estructuras de origen y de destino se mantienen durante la operación de índice en línea, el uso de recursos para insertar, actualizar y eliminar transacciones aumenta, potencialmente hasta el doble.
- Las operaciones de índice a gran escala pueden generar grandes cargas de datos que pueden hacer que el registro de transacciones se llene rápidamente.
- La recompilación de índices en línea puede provocar más fragmentación.

© JMA 2016. All rights reserved

Extensiones

- Unidad básica de almacenamiento de datos (E/S) que consta de 8 páginas contiguas físicamente. El SQL Server lee y escribe extensiones completas.
- Las extensiones están compuestas por 64 KB contiguos de espacio en disco, es decir, 16 extensiones por megabyte.
- Para hacer que la asignación de espacio sea eficaz, no se asigna extensiones completas a tablas con pequeñas cantidades de datos.
- Dos tipos de extensiones:
 - Mixtas
 - Pueden contener datos de hasta 8 objetos de la BBDD, uno por página
 - Uniformes
 - Contienen datos de un único objeto de la BBDD
- Cuando se crea un objeto se sitúa en una extensión mixta (<= v.14)
- Según se va llenando de datos, pasan a ocupar una extensión uniforme

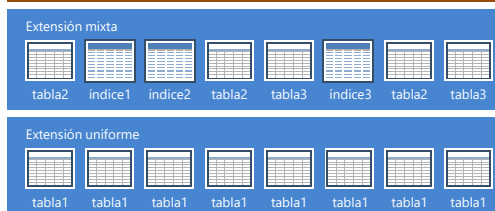
© JMA 2016. All rights reserved

Extensiones y páginas

Extensión de cabecera de fichero



Extensiones



GAMs y SGAMs

Current use of extent	GAM bit setting	SGAM bit setting
Free, not being used	1	0
Uniform, extent, or full mixed extent	0	0
Current use of extent	0	1

© JMA 2016. All rights reserved

Administrar las asignaciones

- Las páginas GAM registran las extensiones que han sido asignadas. Las páginas SGAM registran las extensiones que actualmente se están utilizando como extensiones mixtas y además tienen al menos una página sin utilizar. Utilizan un bit por cada extensión y cada una cubre 64.000 extensiones o 4 GB de datos.
- Las páginas Espacio disponible en páginas (PFS) registran el estado de asignación de cada página, si una página concreta está asignada y la cantidad de espacio libre en cada página. Cada PFS cubre 8088 páginas o 64 Mb de datos.
- Una página del Mapa de asignación de índices (IAM) gestiona el espacio utilizado por los objetos y asigna las extensiones en una parte de 4 GB de un archivo de base de datos usado por una unidad de asignación (IN_ROW_DATA, ROW_OVERFLOW_DATA, LOB_DATA). Cada una cubre 64.000 extensiones o 4 GB de datos.

© JMA 2016. All rights reserved

Ficheros de Datos

- De dos tipos
 - Primario (.mdf): Un fichero de datos primario por BBDD (uno y solo uno). Almacena la información del catálogo y las tablas del sistema de la Base de Datos
 - Secundario (.ndf): Ficheros adicionales que pueden contener datos. Utilizados para balancear cargas y mejorar rendimiento
- Los archivos tienen dos nombres:
 - logical_file_name es el nombre que se utiliza para hacer referencia al archivo en las instrucciones Transact-SQL.
 - os_file_name es el nombre del archivo físico en disco.
- Se dimensionan en Megabytes (16 extensiones por Megabyte, 128 páginas por Megabyte)
 - Los ficheros pueden crecer automáticamente: porcentualmente o valor absoluto
 - Se puede impedir o limitar el crecimiento a un número máximo de megas (Máximo 16 TB)
 - Balancean la ocupación

© JMA 2016. All rights reserved

Grupos de Ficheros

- Permiten una gestión más óptima del espacio de una Base de Datos
- Como mínimo cuentan con uno, el PRIMARY
 - No puede ser borrado de la Base de Datos
 - Contiene el fichero Primario (.mdf):.
- El sistema balancea la carga de los datos proporcionalmente a los ficheros de datos de cada grupo
 - Permite un mejor rendimiento en la E/S
- Cada tabla se puede almacenar en grupos distintos, un grupo para las filas, otro para los LOB y cada índice en un grupo.
- Uno de los grupos de ficheros debe ser especificado como soporte de datos por defecto, se le asignaran los elemento que no indiquen lo contrario.
- Los grupos pueden ser marcados de solo lectura, con lo que se impide escribir en los ficheros del grupo.
- Permiten balancean la carga de rendimientos

© JMA 2016. All rights reserved

Particiones

- La creación de particiones en una base de datos mejora el rendimiento y simplifica el mantenimiento. Al dividir una tabla grande en tablas individuales más pequeñas, las consultas que tengan acceso únicamente a una parte de los datos pueden ejecutarse con mayor rapidez, ya que deben recorrer menos datos. Las tareas de mantenimiento (por ejemplo, volver a generar los índices o hacer copias de seguridad de una tabla), pueden ejecutarse con mayor rapidez.
- El particionamiento horizontal divide una tabla en varias particiones que contiene el mismo número de columnas pero menos filas. Se implementa con:
 - Funciones y Esquemas de partición
 - Tablas individuales (se reúnen en Vistas con consultas UNION)
- El particionamiento vertical divide una tabla en varias tablas que contienen menos columnas. La división de fila agrupa columnas de la tabla original y las reparte en varias tablas con menos columnas. Cada fila lógica de una tabla dividida coincide con la misma fila lógica en las demás tablas, con la misma PRIMARY KEY como FOREIGN KEY (requiere JOINS).

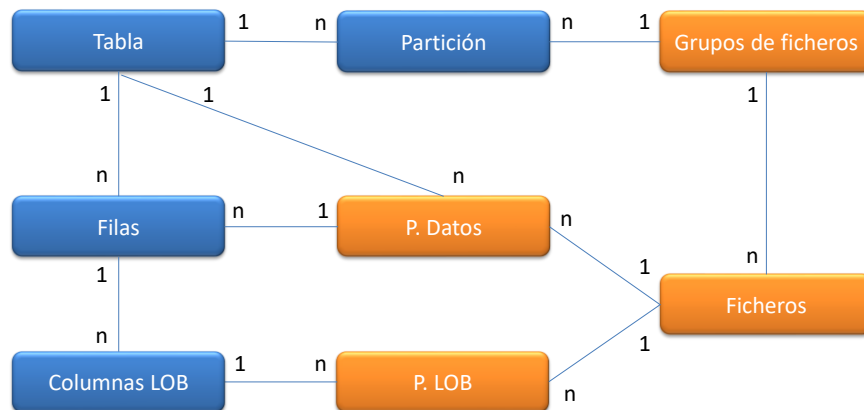
© JMA 2016. All rights reserved

Tablas e índices con particiones

- Sólo se admiten en las ediciones Enterprise.
- La partición facilita el uso de tablas e índices grandes, ya que permite administrar y tener acceso a subconjuntos de datos de forma rápida y eficaz, a la vez que mantiene la integridad de la recopilación de datos.
- Los datos se dividen en sentido horizontal y se reparten por más de un grupo de archivos dentro de la misma base de datos, puede tener un máximo de 1.000 particiones.
- La tabla o el índice se tratarán como una sola entidad lógica cuando se realicen consultas o actualizaciones en los datos.
- Útiles para tablas muy grandes con una división lógica establecida y mejorar el aprovechamiento de recursos.
- Una **función de partición** define la forma de asignar las filas de una tabla o un índice a un conjunto de particiones a partir de los valores de determinadas columnas, denominadas columnas de partición.
- Un **esquema de particiones** asigna cada partición especificada con la función de partición a un grupo de archivos.

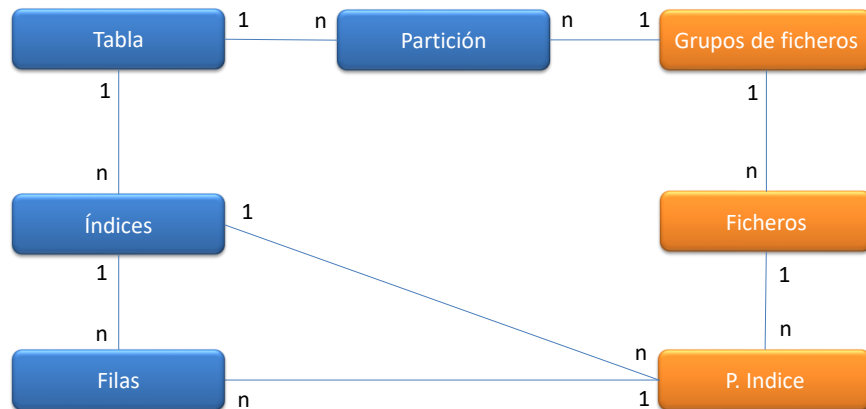
© JMA 2016. All rights reserved

Tabla: Almacenamiento de Datos



© JMA 2016. All rights reserved

Tabla: Almacenamiento de Índices



© JMA 2016. All rights reserved

Operaciones de Partición

- Crear función de partición
`CREATE PARTITION FUNCTION PF (int) AS RANGE RIGHT FOR VALUES (2014, 2015)`
- Crear esquema de partición
`CREATE PARTITION SCHEME PS AS PARTITION PF TO (FG_HIST, FG_ACT, FG_FUT);`
- Quitar una partición y mezclar sus valores en la partición anterior
`ALTER PARTITION FUNCTION PF() MERGE RANGE(2015);`
- Añadir una nueva partición
`ALTER PARTITION SCHEME PS NEXT USED FG2016;`
`ALTER PARTITION FUNCTION PF() SPLIT RANGE(2016);`

© JMA 2016. All rights reserved

Compresión de datos

- SQL Server admite la compresión de fila y de página para las tablas e índices.
- Ayuda para reducir el tamaño de la base de datos.
- Puede contribuir a mejorar el rendimiento de las cargas de trabajo que hacen un uso intensivo de las operaciones de E/S porque los datos se almacenan en menos páginas y las consultas deben leer menos páginas del disco.
- Requieren recursos de CPU adicionales en el servidor de base de datos para comprimir y descomprimir los datos, mientras los datos se intercambian con la aplicación.
- Para las tablas e índices con particiones, la opción de compresión se puede configurar para cada partición una configuración de compresión diferente.

© JMA 2016. All rights reserved

Nivel de compresión

- El tipo de compresión se puede establecer en ROW, PAGE o NONE
 - ROW: Se realiza a nivel de fila. Todas las columnas de longitud fija se tratan como si fueran de longitud variable. Ocupan lo que el dato contenido. No afecta a las columnas LOB.
 - PAGE: Se realiza a nivel de página completa, empieza con la compresión ROW, a la que se le aplica los algoritmos de compresión por prefijos y patrones (diccionarios).
- La compresión conlleva una sobrecarga de datos que puede superar la ocupación sin comprimir.

```
ALTER TABLE Tabla REBUILD WITH (DATA_COMPRESSION = PAGE);
ALTER TABLE TablaParticionada REBUILD PARTITION = 2 WITH
(DATA_COMPRESSION = ROW) ;
```

© JMA 2016. All rights reserved

Estimación de la ocupación

- Para calcular el espacio necesario para almacenar datos en una tabla:
 1. Calcular el espacio necesario para las filas:
 1. Calcular el numero de paginas de datos.
 2. Calcular el numero de páginas LOB por cada columna, sumar todas las columnas y multiplicar por el número de filas.
 2. Calcular el espacio necesario para los índices:
 1. Por cada índice, calcular el numero de paginas de índice.
 2. Sumar el numero de páginas de cada índice.
 3. Sumar el espacio necesario para las filas con el espacio necesario para los índices.
- El resultado viene expresado en páginas, para obtenerlo en megas, dividirlo entre 128.

© JMA 2016. All rights reserved

Estimar tamaños

- Tabla no agrupada (montón)
 - IN_ROW_DATA
 - $LF = \sum(TMC) + 24 * NCL$
 - $FxP = FLOOR(8096 / LF)$
 - $NPD = CEILING(NF / FxP)$
 - Por columna LOB
 - $NPL = CEILING(TML / 8000) * NF$
 - Por índice
 - $LRI = \sum(TMC) + 8$
 - $FxN = FLOOR(8096 / LRI)$
 - $NPI = CEILING(NF / FxN)$
 - Total
 - $NP = NPD + \sum(NPL) + \sum(NPI)$
- Tabla no agrupada (montón)
 - Por índice
 - $LRI = \sum(TMC) + \sum(TMCA)$
 - $FxN = FLOOR(8096 / LRI)$
 - $NPI = CEILING(NF / FxN)$
- En mb: $NP / 128$

Abreviaturas:

NF: Número de filas (inicial, previsto, periodo temporal, ...)
 LF: Longitud de fila (bytes)
 TMC: Tamaño medio de columna (bytes)
 FxP: Filas por páginas
 NPD: Número de páginas de datos
 TML: Tamaño medio del dato LOB (bytes)
 NPL: Número de páginas por columna LOB
 LRI: Longitud de registro de índice (bytes)
 FxN: Filas por páginas nodo de índice
 NPI: Número de páginas de índices
 TMCA: Tamaño medio de columna del índice agrupado

© JMA 2016. All rights reserved

Gestión automatizada

- Se puede automatizar la gestión del espacio asignado a una base de datos, fundamentalmente para sistemas no administrados.
- Crecimiento (solo a nivel de ficheros):
 - Se puede permitir que los ficheros crezcan automáticamente.
 - El crecimiento de ficheros provoca fragmentación en el disco duro con la consecuente reducción de rendimiento.
- Decrecimiento (solo a nivel de base de datos):
 - La opción de base de datos AUTO_SHRINK marca que cuando la base de datos tenga un 75% libre, reduzca todos los ficheros dejando un 25% de espacio libre a cada uno.
- Los automatismos degradan el rendimiento al entrar en periodos de máxima actividad de la base de datos.
- Los cambios de tamaño conllevan bloqueos a nivel de fichero y de base de datos.

© JMA 2016. All rights reserved

Registro de Transacciones

- Usados por el motor para grabar los cambios generados durante el uso de la Base de Datos
 - Registra tanto transacciones de usuarios como transacciones implícitas del sistema
 - El registro de transacciones es un archivo de registro circular.
- Utilizados para mantener la integridad de la Base de Datos y repararla en caso de fallo del sistema
- Toda base de datos debe tener al menos uno como mínimo.
- La extensión del archivo transaction log es .ldf
- Al igual que los ficheros de datos:
 - Se dimensionan en megas.
 - Pueden crecer automáticamente: porcentualmente o valor absoluto
 - Se puede impedir o limitar el crecimiento a un número máximo de megas (Máximo 12 TB)
- La falta de espacio puede impedir operar con la base de datos.

© JMA 2016. All rights reserved

Transacciones

- Una transacción es una secuencia de operaciones realizadas como una sola unidad lógica de trabajo.
- Una unidad lógica de trabajo debe exhibir cuatro propiedades, conocidas como propiedades ACID (atomicidad, coherencia, aislamiento y durabilidad), para ser calificada como transacción:
 - Atomicidad: Una transacción debe ser una unidad atómica de trabajo, tanto si se realizan todas sus modificaciones en los datos, como si no se realiza ninguna de ellas.
 - Coherencia: Cuando finaliza, una transacción debe dejar todos los datos en un estado coherente, es decir, se deben cumplir todas las reglas de integridad de todos los datos.
 - aislamiento: Las modificaciones realizadas por transacciones simultáneas se deben aislar de las modificaciones llevadas a cabo por otras transacciones simultáneas.
 - Durabilidad: Una vez concluida una transacción, sus efectos son permanentes en el sistema. Las modificaciones persisten aún en el caso de producirse un error del sistema.

© JMA 2016. All rights reserved

Especificar y exigir transacciones

- Los programadores de SQL son los responsables de iniciar y finalizar las transacciones en los puntos que exijan la coherencia lógica de los datos.
- El programador debe asegurar la atomicidad de las transacciones.
- SQL Server proporciona:
 - Servicios de bloqueo que preservan el aislamiento de la transacción.
 - Servicios de registro que aseguran la durabilidad de la transacción:
 - Aún en el caso de que falle el hardware del servidor, el sistema operativo o el propio SQL Server, SQL Server utiliza registros de transacciones, al reinicio, para deshacer automáticamente las transacciones incompletas en el momento en que se produjo el error en el sistema.
 - Características de administración de transacciones que exigen la atomicidad y coherencia de la transacción:
 - Una vez iniciada una transacción, debe concluirse correctamente o SQL Server deshacerá todas las modificaciones de datos realizadas desde que se inició la transacción.

© JMA 2016. All rights reserved

Tipos de transacciones

- Transacciones de confirmación automática
 - Éste es el modo predeterminado de SQL Server. Cada instrucción individual de Transact-SQL se confirma cuando termina. No tiene que especificar instrucciones para controlar las transacciones.
- Transacciones explícitas
 - Se inicia con BEGIN TRANSACTION, se confirma con COMMIT y se descarta con ROLLBACK.
- Transacciones implícitas
 - Establezca el modo de transacción implícita a través de una función de la API o la instrucción SET IMPLICIT_TRANSACTIONS ON de Transact-SQL. La siguiente instrucción inicia automáticamente una nueva transacción. Cuando se concluye la transacción, la instrucción de Transact-SQL siguiente inicia una nueva transacción.
- Transacciones distribuidas
 - Implican a varias instancias y se dividen en Fases de preparación y de confirmación.

© JMA 2016. All rights reserved

Información registrada

- Registro de la operación lógica (SQL).
 - Para confirmar la operación lógica, se vuelve a ejecutar.
 - Para deshacer la operación lógica, se ejecuta la operación lógica inversa.
- Registro de las imágenes (páginas) anterior y posterior.
 - Para confirmar la operación, se aplica la imagen posterior.
 - Para deshacer la operación, se aplica la imagen anterior.
- En el registro de transacciones se registran muchos tipos de operaciones, que incluyen:
 - El inicio y el final de cada transacción.
 - Todas las modificaciones de los datos (inserción, actualización y eliminación). Esto incluye las modificaciones de las tablas del sistema hechas por procedimientos almacenados del sistema o por instrucciones del lenguaje de definición de datos (DDL).
 - Las asignaciones o cancelaciones de asignación de extensiones.
 - La creación o eliminación de una tabla o un índice.

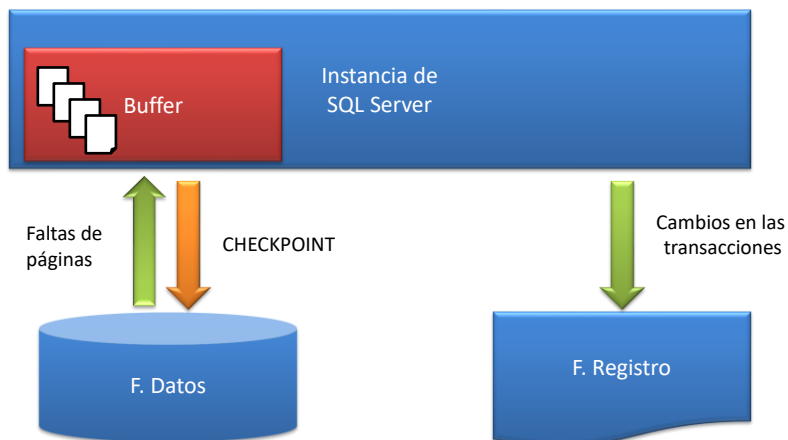
© JMA 2016. All rights reserved

Operativa

- Toda operación se registra en el registro de transacciones en el momento de producirse.
- Los datos se cargan en memoria, en caso de no estar ya, y modifican en memoria, pero no se graban en los ficheros de datos una vez terminadas las modificaciones.
- Los datos se graban periódicamente en los ficheros de datos en los denominados puntos de comprobación (CHECKPOINT).
 - Se marca el inicio del punto de comprobación en el registro de transacciones.
 - Se graban las páginas desfasadas en los ficheros de datos.
 - Se marca el final del punto de comprobación en el registro de transacciones.
- Se reserva el cauce de lectura para la carga de páginas de datos y el de escritura para la grabación en el fichero de transacciones.

© JMA 2016. All rights reserved

Lectura/Escritura



© JMA 2016. All rights reserved

Puntos de comprobación

- Se ejecutan de forma fija:
 - Cuando se ejecuta una instrucción CHECKPOINT. Se insertan puntos de comprobación en la base de datos actual de la conexión.
 - Cuando se utiliza ALTER DATABASE para cambiar una opción de base de datos. ALTER DATABASE inserta puntos de comprobación en la base de datos cuando se cambian las opciones de la base de datos.
 - Cuando se detiene una instancia de SQL Server por: Ejecutar una instrucción SHUTDOWN o se para el servicio.
- Se genera periódicamente de manera automática en cada base de datos para reducir el tiempo que tardaría la instancia en recuperar la base de datos.
 - Se genera cuando el número de registros alcanza el número que estima que puede procesar durante el tiempo especificado en la opción intervalo de recuperación.
 - En el modelo de recuperación sencilla, se genera puntos adicionales cuando el registro está ocupado en un 70 por ciento.

© JMA 2016. All rights reserved

Modelos de recuperación

- Sencillo:
 - Uso mínimo del registro de transacciones, se eliminan las transacciones una vez superado su punto de comprobación.
- Completo:
 - Uso extensivo del registro de transacciones, se eliminan las transacciones cuando se aseguran mediante una copia de seguridad.
- Carga masiva:
 - Modelo completo pero con registro mínimo de las operaciones de carga masiva. Truncan la secuencia de restauración.
- Cuando se produce la recuperación:
 - Al arrancar la instancia
 - Al restaurar la base de datos

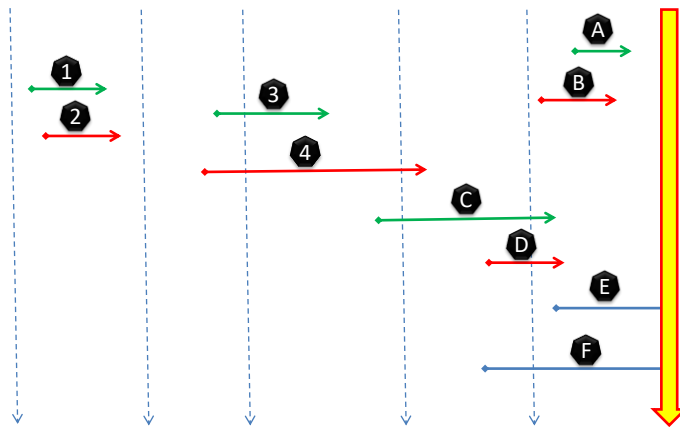
© JMA 2016. All rights reserved

Proceso de recuperación

- Relanza las transacciones iniciadas y confirmadas después del último punto de comprobación.
- Ignora las transacciones iniciadas y canceladas después del último punto de comprobación.
- En las transacciones iniciadas antes del último punto de comprobación y confirmadas después del último punto de comprobación, sobrescribe los ficheros de datos con la imagen posterior almacenada en el registro de transacciones.
- En las transacciones iniciadas antes del último punto de comprobación y descartadas después del último punto de comprobación, sobrescribe los ficheros de datos con la imagen anterior almacenada en el registro de transacciones.
- En las transacciones iniciadas antes del último punto de comprobación y no completadas después del último punto de comprobación, sobrescribe los ficheros de datos con la imagen anterior almacenada en el registro de transacciones.

© JMA 2016. All rights reserved

Recovery



© JMA 2016. All rights reserved

Bloqueos

- Balanceo: simultaneidad frente a coherencia
- Estrategias: optimista y pesimista
- Problemas de simultaneidad
- Grados de aislamiento
- Modos de interbloqueo
- Granularidad y jerarquías de bloqueo
- Interbloqueos

© JMA 2016. All rights reserved

Problemas de simultaneidad

- Actualizaciones perdidas:
 - Este problema surge cuando dos o más transacciones seleccionan la misma fila y, a continuación, la actualizan de acuerdo con el valor seleccionado originalmente.
 - Ninguna transacción es consciente de las otras transacciones.
 - La última actualización sobrescribe las actualizaciones realizadas por las otras transacciones y, en consecuencia, se pierden datos.
- Dependencia no confirmada (lectura no actualizada o desfasada):
 - Este problema se produce cuando una transacción selecciona una fila que está siendo actualizada por otra transacción.
 - La transacción que llega en segundo lugar lee datos que todavía no han sido confirmados y que la transacción que actualiza la fila puede modificar.

© JMA 2016. All rights reserved

Problemas de simultaneidad

- **Análisis incoherente (lectura irrepitable):**
 - Este problema se produce cuando una transacción obtiene acceso a la misma fila varias veces y en cada ocasión lee datos diferentes.
 - El análisis incoherente es similar a la dependencia no confirmada en tanto que una transacción está modificando los datos que está leyendo una segunda transacción.
 - Sin embargo, en el caso del análisis incoherente, los datos que lee la segunda transacción están confirmados por la transacción que realizó el cambio.
 - Además, el análisis incoherente comprende varias lecturas (dos o más) de la misma fila, y cada vez hay otra transacción que modifica la información; de ahí el término lectura irrepitable.

© JMA 2016. All rights reserved

Problemas de simultaneidad

- **Lecturas fantasmas o irrepitibles:**
 - Este problema se produce cuando se realiza una acción de insertar o eliminar en una fila y ésta pertenece a un intervalo de filas que está leyendo una transacción.
 - La primera lectura que hizo la transacción en el intervalo de filas muestra una fila que ya no existe en la segunda lectura o en lecturas sucesivas porque otra transacción la ha eliminado.
 - De forma similar, como consecuencia de una inserción realizada por otra transacción, la segunda lectura o las lecturas sucesivas de la transacción muestran una fila que no existía en la primera lectura.

© JMA 2016. All rights reserved

Niveles de aislamiento

- **READ UNCOMMITTED**

Implementa las lecturas no confirmadas o el bloqueo de nivel de aislamiento 0, lo que significa que no hay bloqueos compartidos y que los bloqueos exclusivos no están garantizados. Cuando se establece esta opción, es posible leer datos no confirmados, los valores pueden cambiar y pueden aparecer y desaparecer filas en el conjunto de datos antes del final de la transacción. Esta opción tiene el mismo efecto que establecer NOLOCK en todas las tablas y en todas las instrucciones SELECT de la transacción. Se trata del menos restrictivo de los cuatro niveles de aislamiento.

- **READ COMMITTED (predeterminada)**

Especifica que se mantengan los bloqueos compartidos mientras se leen datos para evitar lecturas no actualizadas, pero se pueden modificar los datos antes del final de la transacción, lo que provoca lecturas no repetibles o datos fantasmas. Esta opción es la predeterminada en SQL Server.

© JMA 2016. All rights reserved

Niveles de aislamiento

- **REPEATABLE READ**

Se establecen bloqueos para todos los datos utilizados en la consulta, lo que impide que otros usuarios los actualicen, aunque es posible insertar nuevas filas fantasmas en los datos que otro usuario establezca, de modo que se incluyan en lecturas posteriores de la misma transacción. Como la simultaneidad es inferior que el nivel de aislamiento predeterminado, sólo se debe usar esta opción cuando sea necesario.

- **SERIALIZABLE**

Se establece un bloqueo de intervalo en el conjunto de datos, lo que impide que otros usuarios actualicen o inserten filas en el conjunto de datos hasta que finalice la transacción. Es el más restrictivo de los cuatro niveles de aislamiento. Al ser menor la simultaneidad, sólo se debe utilizar esta opción cuando sea necesario. Esta opción tiene el mismo efecto que establecer HOLDLOCK en todas las tablas y en todas las instrucciones SELECT de la transacción.

© JMA 2016. All rights reserved

Niveles de aislamiento

- **SNAPSHOT**

Especifica que los datos leídos por cualquier instrucción de una transacción sean la versión coherente, desde el punto de vista transaccional, de los datos existentes al comienzo de la transacción.

La transacción únicamente puede reconocer las modificaciones de datos confirmadas antes del comienzo de la misma.

Las instrucciones que se ejecuten en la transacción actual no verán las modificaciones de datos efectuadas por otras transacciones después del inicio de la transacción actual.

El efecto es el mismo que se obtendría si las instrucciones de una transacción obtuviesen una instantánea de los datos confirmados tal como se encontraban al comienzo de la transacción.

© JMA 2016. All rights reserved

Niveles de aislamiento

Nivel de aislamiento	Actualización perdidas	Lectura no actualizada	Lectura no repetible	Lectura Fantasma
Lectura no confirmada (READ UNCOMMITTED)	Sí	Sí	Sí	Sí
Lectura confirmadas (READ COMMITTED)	Sí	No	Sí	Sí
Lectura repetible (REPEATABLE READ)	No	No	No	Sí
Instantánea (SNAPSHOT)	No	No	No	No
Serializable (SERIALIZABLE)	No	No	No	No

© JMA 2016. All rights reserved

Modos de bloqueo

- Los modos de bloqueo determinan el modo en que las transacciones simultáneas pueden tener acceso a los recursos.
 - **Compartido (S)**: Se utiliza para operaciones de lectura que no cambian ni actualizan datos, como la instrucción SELECT.
 - **Actualizar (U)**: Se utiliza en recursos que se pueden actualizar. Evita una forma común de interbloqueo que se produce cuando varias sesiones leen, bloquean y actualizan recursos.
 - **Exclusivo (X)**: Se utiliza para operaciones de modificación de datos, como INSERT, UPDATE o DELETE. Garantiza que no puedan realizarse varias actualizaciones simultáneamente en el mismo recurso.
 - **Intención**: Se utiliza para establecer una jerarquía de bloqueos. Los tipos de bloqueo de intención son: **intención compartido (IS)**, **intención exclusivo (IX)** y **compartido con intención exclusivo (SIX)**.
 - **Esquema**: Se utiliza cuando se ejecuta una operación que depende del esquema de una tabla. Hay dos tipos de bloqueo de esquema: **modificación del esquema (Sch-M)** y **modificación de estabilidad (Sch-S)**.
 - **Actualización masiva (BU)**: Se utiliza cuando se copian datos de forma masiva en una tabla y se especifica la sugerencia TABLOCK.
 - **Intervalo de claves**: Protege el intervalo de filas que lee una consulta cuando se utiliza el nivel de aislamiento de transacciones serializables. Garantiza que otras transacciones no puedan insertar filas que podrían incluirse como respuesta de las consultas de la transacción serializable si las consultas se volvieron a ejecutar.

© JMA 2016. All rights reserved

Granularidad y jerarquías de bloqueo

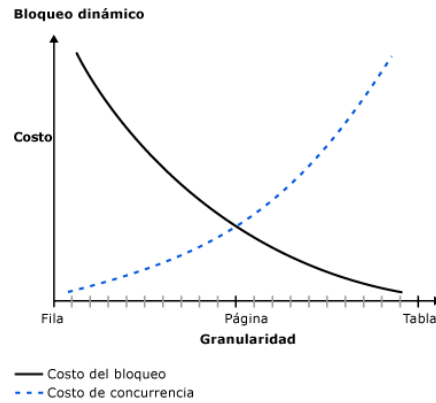
La extensión de bloqueo es el proceso de convertir muchos bloqueos concretos en menos bloqueos más generales, lo que reduce la sobrecarga del sistema al tiempo que aumenta la probabilidad de contención de simultaneidad.

- **RID**: Identificador de fila que se utiliza para bloquear una sola fila de un montón.
- **KEY**: Bloqueo de fila dentro de un índice que se utiliza para proteger intervalos de claves en transacciones serializables.
- **PAGE**: Página de 8 kilobytes (KB) de una base de datos, como páginas de datos o de índices.
- **EXTENT**: Grupo contiguo de ocho páginas, como páginas de datos o de índices.
- **HoBT**: Montón o árbol b. Bloqueo que protege un árbol B (índice) o las páginas de datos del montón en una tabla que no posee un índice agrupado.
- **TABLE**: Tabla completa, con todos los datos e índices.
- **FILE**: Archivos de la base de datos.
- **APPLICATION**: Recurso especificado por la aplicación.
- **METADATA**: Bloqueos de metadatos.
- **ALLOCATION_UNIT**: Unidad de asignación.
- **DATABASE**: Base de datos completa.

© JMA 2016. All rights reserved

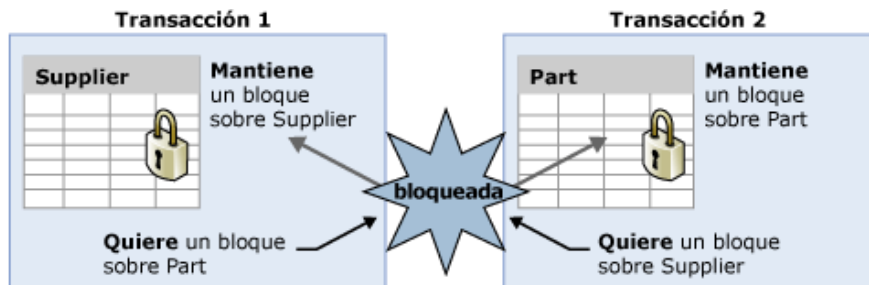
Bloqueo dinámico

- La utilización de bloqueos de bajo nivel, como los de fila, aumenta la simultaneidad reduciendo la probabilidad de que dos transacciones soliciten bloqueos de los mismos datos al mismo tiempo. También aumenta el número de bloqueos y los recursos necesarios para administrarlos.
- Los bloqueos de alto nivel de tabla o página producen una sobrecarga menor, pero a costa de reducir la simultaneidad.



© JMA 2016. All rights reserved

Interbloqueos



© JMA 2016. All rights reserved

Interbloqueos

- Un interbloqueo se produce cuando hay una dependencia cíclica entre dos o más subprocesos para algún conjunto de resultados.
- Detección de interbloqueos: un subproceso de supervisión de bloqueos que periódicamente inicia una búsqueda por todas las tareas de una instancia
 - El intervalo predeterminado es de 5 segundos
 - Baja hasta sólo 100 milisegundos
 - Segundo intento, solución de interbloqueo
- Solución de interbloqueo
 - Prevención mediante el Motor de consultas y plan de ejecución
 - Expiración por tiempos de espera
 - sesión, transacción, SET LOCK_TIMEOUT, ...
 - Finalización de subproceso (error 1205)
 - Menor nivel (SET DEADLOCK_PRIORITY)
 - reversión menos costosa (deshacer)
 - a igualdad: aleatoriamente

© JMA 2016. All rights reserved

Interbloqueos

- Para evitar los interbloqueos:
 1. No abra una transacción mientras examina los datos si es posible.
 2. Las transacciones deben ser atómicas.
 3. Haga la transacción lo más corta posible.
 4. La transacción no debería depender de interfaz de usuario.
 5. Haga un uso inteligente de los niveles más bajos de aislamiento de las transacciones.
 6. Maneje siempre que sea posible los conjuntos de datos en el mismo orden

© JMA 2016. All rights reserved

Información de bloqueo

- Los bloqueos pueden reducir el rendimiento de la instancia.
- SQL Server ofrece varias formas de obtener información acerca de la actividad de bloqueo actual en una instancia:
 - Se puede realizar una consulta de la vista de administración dinámica **sys.dm_tran_locks** para obtener información acerca del estado actual de bloqueo de una instancia del Database Engine (Motor de base de datos).
 - Mediante el SQL Server Profiler, se puede especificar la categoría de eventos de los bloqueos para capturar información acerca de los bloqueos de eventos en una traza.
 - En el Monitor de sistema, se pueden especificar contadores del objeto de bloqueos para supervisar el nivel de bloqueo de una instancia del Motor de base de datos.

© JMA 2016. All rights reserved

Ajustes de instancia

- Management Studio - Propiedades:
 - Memoria (máxima y mínima)
 - Procesador (afinidad)
 - Seguridad (modo y auditorias)
 - Conexiones (número y valores por defecto)
 - Configuración de base de datos (por defecto)
 - Opciones Avanzadas (paralelismo, red, ...)
- Transact SQL:
 - `sp_configure 'show advanced options', 1;`
 - `RECONFIGURE;`
- Especificaciones de producto para SQL Server

© JMA 2016. All rights reserved

Buffer Pool Extension

- A partir de SQL Server 2014 (12.x), en versiones Estándar o superior, la extensión del grupo de búferes proporciona una perfecta integración de una extensión de la memoria de acceso aleatorio no volátil (es decir, una unidad de estado sólido) con el grupo de búferes del Motor de base de datos para mejorar considerablemente el rendimiento de E/S.
- El almacenamiento SSD se utiliza como extensión del subsistema de memoria en lugar del subsistema de almacenamiento en disco.
- Establecer la opción de extensión del grupo de búferes:

```
ALTER SERVER CONFIGURATION
SET BUFFER POOL EXTENSION ON
(FILENAME = 'F:\SSDCACHE\Example.BPE', SIZE = 50 GB);
```

© JMA 2016. All rights reserved

Propiedades de la base de datos

- Management Studio :
 - Nombre, Propietario, Modelo de recuperación, Nivel de compatibilidad y Tipo de contención.
 - Grupos de archivos (Filas, FILESTREAM, Datos con optimización para memoria) y Archivos.
 - Automatismos, contención, estado, recuperación y valores por defecto.
- Transact SQL:
 - CREATE DATABASE y ALTER DATABASE
 - ALTER DATABASE SET
 - Función: DATABASEPROPERTYEX

© JMA 2016. All rights reserved

Automatismos

- Automatismos
 - Crear estadísticas automáticamente: Se crean automáticamente las estadísticas de optimización que faltan.
 - Creación automática de estadísticas incrementales: Si se desea utilizar la opción incremental cuando se crean estadísticas por partición.
 - Actualizar estadísticas automáticamente: Se actualizan automáticamente las estadísticas de optimización no actualizadas.
 - Actualizar estadísticas automática y asincrónicamente: Las consultas que inician una actualización automática de estadísticas obsoletas no esperan a que las estadísticas se actualicen antes de la compilación. Las consultas posteriores usan las estadísticas actualizadas si están disponibles.
 - Cerrar automáticamente: La base de datos se cierra y se liberan sus recursos después de que salga el último usuario.
 - Reducir automáticamente: Cuando la base de datos tenga un 75% libre, se reducen todos los ficheros dejando un 25% de espacio libre a cada uno.
- Motor de consultas:
 - Estimación de cardinalidad heredada, Examen de parámetros, Correcciones del optimizador de consultas, Max DOP (Paralelismo), Parametrización

© JMA 2016. All rights reserved

Optimizar la configuración de base de datos

- Colocar los datos y los archivos de registro en unidades independientes evita cuellos de botellas y balancea la carga entre los diferentes dispositivos.
- La compresión de datos puede contribuir a mejorar el rendimiento de las cargas de trabajo que hacen un uso intensivo de las operaciones de E/S porque los datos se almacenan en menos páginas y las consultas deben leer menos páginas del disco. No obstante, se requieren recursos de CPU adicionales en el servidor de base de datos para comprimir y descomprimir los datos, mientras los datos se intercambian con la aplicación.
- Las nuevas versiones suelen incorporar importantes mejoras de rendimiento, el nivel de compatibilidad de una base de datos puede limitar el acceso a dichas mejoras

© JMA 2016. All rights reserved

RAID

- RAID (matriz redundante de discos independientes) es un sistema de discos, denominado matriz, que contiene varias unidades de disco para proporcionar mayor rendimiento, confiabilidad y capacidad de almacenamiento a un costo menor. Las matrices con tolerancia a errores están divididas en seis niveles RAID: del 0 al 5. Cada nivel utiliza un algoritmo diferente para implementar la tolerancia a errores. Con SQL Server se utilizan normalmente los niveles RAID 0, 1 y 5.
- Una matriz de discos de hardware mejora el rendimiento de E/S debido a que las funciones de E/S, tales como la creación de bandas y reflejos, se controlan de forma eficaz mediante el firmware. Por el contrario, una RAID basada en el sistema operativo representa un menor costo pero consume ciclos del procesador. Cuando el costo representa un problema y se necesita redundancia y alto rendimiento, los volúmenes RAID-5 son una buena solución.

© JMA 2016. All rights reserved

RAID

- La creación de bandas de datos (RAID 0) es la configuración RAID con mejor rendimiento, pero si un disco tiene errores, se deja de tener acceso a todos los datos del conjunto de bandas. Una técnica común de instalación de los sistemas de administración de bases de datos relacionales consiste en configurar la base de datos en una unidad RAID 0 y, a continuación, colocar el registro de transacciones en una unidad reflejada (RAID 1). Puede obtener el mejor rendimiento de E/S de disco para la base de datos y mantener la posibilidad de recuperar los datos a través de un registro de transacciones reflejado, siempre y cuando realice copias de seguridad de la base de datos con regularidad.
- Si es necesario recuperar rápidamente los datos, es posible reflejar el registro de transacciones y colocar la base de datos en un disco RAID 5, aunque con un rendimiento menor. RAID 5 proporciona redundancia de todos los datos de la matriz, permitiendo el cambio de unidades sin que se produzca tiempo de inactividad del sistema.

© JMA 2016. All rights reserved

Dimensionado de la B.D. tempdb

- La base de datos del sistema **tempdb** es un recurso global disponible para todos los usuarios de la instancia de SQL Server, utilizada para:
 - Objetos temporales de usuario que se hayan creado explícitamente, como: tablas y procedimientos temporales, variables de tabla o cursores.
 - Objetos internos creados por Motor de base de datos de SQL Server, como: tablas de trabajo para almacenar resultados intermedios para colas u ordenación.
 - Versiones de fila generadas por las transacciones de modificación de datos en una base de datos que utiliza transacciones de lectura confirmada que usan transacciones de aislamiento de versiones de fila o de aislamiento de instantáneas.
 - Versiones de fila que se hayan generado mediante transacciones de modificación de datos para características, como: operaciones de índice en línea, conjuntos de resultados activos múltiples (MARS) y desencadenadores AFTER.

© JMA 2016. All rights reserved

Consideraciones sobre tempdb

- tempdb se vuelve a crear cada vez que se inicia SQL Server, de forma que el sistema siempre se inicia con una copia limpia de la base de datos.
- El tamaño de tempdb puede afectar al rendimiento de un sistema, si el tamaño es demasiado pequeño, el procesamiento del sistema podría estar demasiado ocupado con el crecimiento automático de la base de datos y, por tanto, no podría satisfacer las necesidades de carga de trabajo cada vez que se inicia SQL Server.
- La ausencia de espacio en tempdb puede acarrear el bloqueo de la instancia, los ficheros deben crecer automáticamente.

© JMA 2016. All rights reserved

Operaciones no disponibles sobre tempdb

- Agregar grupos de archivos, quitar el grupo de archivos principal, el archivo de datos principal o el archivo de registro.
- Activar reducir o cerrar automáticamente.
- Cambiar el nombre de la base de datos o del grupo de archivos principal.
- Cambiar la intercalación o el propietario de la base de datos (es propiedad de **sa**).
- Establecer la base de datos en OFFLINE o READ_ONLY.
- Crear una instantánea, eliminar la base de datos o eliminar el usuario **guest**.
- Realizar una copia de seguridad o restaurar la base de datos.
- Habilitar el mecanismo de captura de cambios en los datos.
- Participar en el reflejo de la base de datos.
- Ejecutar DBCC CHECKALLOC o DBCC CHECKCATALOG.

© JMA 2016. All rights reserved

Optimizar el rendimiento de tempdb

- Establecer el tamaño de los archivos en un valor lo suficientemente alto para contener la carga de trabajo habitual del entorno, evitando que se expanda con demasiada frecuencia.
- Debe establecerse que crezca automáticamente para las situaciones no previstas, con un incremento suficiente que evite hacerlo con demasiada frecuencia.
- La división de tempdb en varios archivos de datos del mismo tamaño proporciona un alto grado de eficiencia paralela en las operaciones que usan tempdb.
- Colocar la base de datos tempdb en un subsistema de E/S rápido. Cree bandas en disco si hay muchos discos conectados directamente.
- Colocar la base de datos tempdb en discos diferentes de los que usan las bases de datos de usuario.

© JMA 2016. All rights reserved

AUTOMATIZACIÓN DE TAREAS ADMINISTRATIVAS

© JMA 2009. All rights reserved

Tareas habituales del administrador

- **Mantenimiento preventivo**
 - Operaciones rutinarias:
 - Copias de seguridad
 - Importación y exportación de datos
 - Replicaciones
 - Mantenimiento de datos (paso a históricos, consolidaciones, ...)
 - Mantenimiento de estructuras (reindexación, liberación de espacio, ...)
 - Supervisión del sistema y acciones correctivas:
 - Espacio en la base de datos y del registro
 - Rendimientos
 - **Mantenimiento reactivo**
 - Resolución de incidencias
-

© JMA 2009. All rights reserved

Agente de SQL Server

- SQL Server Agent es el servicio responsable de automatizar las tareas administrativas de SQL Server
- Se debe configurar el servicio SQL Server Agent:
 - para inicio automático
 - para reiniciarlos de forma automática si se detienen de forma inesperada
 - que la cuenta de inicio de sesión esté asignada a la función sysadmin
- La cuenta de inicio de sesión DEBERÍA:
 - Usar el modo de autenticación de Windows
 - Utilizar una cuenta de usuario de dominio de Windows
 - Asignar esta cuenta al grupo local Administradores

© JMA 2009. All rights reserved

Base de Datos msdb

- La Base de Datos msdb se encarga de almacenar la información referente al servicio SQL Server Agent, y a la planificación de trabajos.
- Entre la información que almacena se encuentra la información relativa a:
 - Trabajos
 - Alertas
 - Operadores
 - Programaciones
 - Planes de Mantenimiento
 - Historial
- Para utilizar el Agente SQL Server, los usuarios deben ser miembros de una o más de las siguientes funciones fijas de base de datos:
 - SQLAgentUserRole
 - SQLAgentReaderRole
 - SQLAgentOperatorRole

© JMA 2009. All rights reserved

Correo electrónico de base de datos

- El Correo electrónico de base de datos permite a las aplicaciones de base de datos enviar mensajes de correo electrónico a los usuarios.
- Los mensajes enviados pueden incluir resultados de consultas y archivos de cualquier recurso de la red.
- Utiliza el protocolo estándar SMTP
- Está organizado en Perfiles: Públicos y Privados
- Cada perfil puede incluir varias cuentas de conmutación por error.
- Se encuentra desactivado de forma predeterminada.
- La base de datos msdb es la base de datos host del Correo electrónico de base de datos.
- Para enviar correo debe pertenecer a la función de BD DatabaseMailUserRole en msdb.
- Características
 - Seguridad de los perfiles, Regulador del tamaño y Extensiones de archivo prohibidas.
 - Configuración integrada, Registro, Auditoría y Compatibilidad con HTML
 - Procedimiento sp_send_dbmail, vistas y procedimientos sysmail_

© JMA 2009. All rights reserved

Operadores

- Un operador es la personas responsable del mantenimiento, total o parcial, de una o varias bases de datos o instancias de SQL Server
- Métodos de envío de Notificaciones:
 - Correo electrónico
 - Localizador (por correo electrónico) (SMS, "pager" o busca)
 - Envío de red (net send)
- Se puede fijar los horarios de uso del localizador (Programación del localizador)
- Un operador a prueba de errores recibe la notificación cuando:
 - no se pueden enviar mensajes al localizador de ninguno de los operadores responsables de la alerta
 - no puede tener acceso a las tablas del sistema en la base de datos msdb

© JMA 2009. All rights reserved

Trabajos

- Un trabajo es un conjunto de pasos secuenciales
- La definición de un paso puede usar:
 - Programas ejecutables y comandos del sistema operativo.
 - Instrucciones Transact-SQL, incluidos los procedimientos almacenados y los procedimientos almacenados extendidos.
 - Secuencias de comandos Microsoft ActiveX
 - Tareas de réplica.
 - Tareas de Analysis Services.
 - Paquetes de Integration Services.
- El flujo del trabajo se controla indicando el siguiente paso o deteniéndolo cuando se ejecute correctamente o cause un error, así como los reintentos después de dar error.
- El propietario del trabajo debe tener los permisos adecuados para ejecutar todos los pasos del trabajo.
- Se les pueden asignar Operadores a los que se notifican el resultado de las ejecuciones.
- Los trabajos se pueden organizar en categorías.

© JMA 2009. All rights reserved

Programaciones

- Un trabajo se puede ejecutar manualmente, según su programación, en respuesta a una alerta o dentro de un plan de mantenimiento.
- Se puede programar trabajos para que se inicien automáticamente:
 - Al iniciarse SQL Server Agent.
 - A una hora específica (sólo una vez).
 - De forma periódica (diaria, semanal o mensualmente).
 - Cuando la CPU esté inactiva (configurable a nivel de servicio).
- Cada trabajo acepta varias programaciones (programación múltiple).
- Los trabajos se pueden desactivar, en cuyo caso no se iniciarán según la programación

© JMA 2009. All rights reserved

Historial de trabajos

- Los trabajos son automáticos pero no deben quedar desatendidos
- Se recomienda el uso de listas de control (CHECKLIST) para su supervisión.
- Revisión del historial de un trabajo individual:
 - Resultado de cada paso del trabajo: correcto o erróneo
 - Duración de ejecución
 - Errores y mensajes
- Se puede supervisar la actividad actual mediante el Monitor de actividad de trabajo.
- Configuración del tamaño del historial de trabajos:
 - Conservar información acerca de cada trabajo
 - Sobrescribir historial cuando se alcance el tamaño máximo

© JMA 2009. All rights reserved

Alertas

- Una alerta indica que ha ocurrido un suceso. Una alerta responde a un tipo de evento específico, pueden ser de los siguientes tipos :
 - Eventos de SQL Server
 - un error específico (Número de error)
 - un error de la gravedad específica (Nivel de gravedad)
 - en una base de datos específica
 - con una la cadena de texto especifica en el mensaje del suceso
 - Condiciones de rendimiento de SQL Server, dependiente de un objeto, contador e instancia.
 - Eventos de WMI, debe definir el Espacio de nombres y la Consulta
- En respuesta a la alerta se puede:
 - Ejecutar un trabajo
 - Mandar una notificación a uno o varios operadores

© JMA 2009. All rights reserved

Planes de mantenimiento

- Permiten configurar y agrupar las principales tareas de mantenimiento necesarias para garantizar el buen funcionamiento de la base de datos.
- El plan puede contar con una o varias de las siguientes tareas:
 - Reorganizar los datos de las páginas de datos y de índices mediante una nueva generación de los índices con un nuevo factor de relleno
 - Comprimir archivos de datos mediante la eliminación de las páginas de la base de datos que estén vacías
 - Actualizar las estadísticas de los índices para el optimizador de consultas.
 - Realizar comprobaciones de coherencia interna de los datos y de las páginas de datos de la base de datos
 - Realizar copias de seguridad de la base de datos y de los archivos de registro de transacciones
 - Configurar el trasvase de registros (sincronizar bases de datos)

© JMA 2009. All rights reserved

Herramientas del administrador

- Vistas del sistema
- Procedimientos y funciones del sistema
- Informes del SSMS
- Utilidades del sistema operativo
- Directivas
- DBCC: Data Base Command Console

© JMA 2009. All rights reserved

Administración basada en directivas

- La administración basada en directivas es un sistema para administrar una o varias instancias de SQL Server similar a la existente en el Windows.
- Las Facetas son un conjunto de propiedades lógicas que modelan el comportamiento o las características de ciertos aspectos administrados. Las facetas vienen predefinidos.
- Una Condición es una expresión booleana que especifica un conjunto de estados permitidos para una o mas facetas.
- Una Directiva de cuando, como y sobre que se aplica una condición.
- Modos de evaluación
 - A petición. Este modo evalúa la directiva cuando lo especifica el usuario directamente.
 - Al cambiar: impedir. Este modo automatizado utiliza desencadenadores DDL para evitar las infracciones de las directivas.
 - Al cambiar: sólo registrar. Este modo automatizado utiliza la notificación de eventos para evaluar una directiva cuando se realiza un cambio relevante.
 - Al programar. Este modo automatizado utiliza un trabajo del Agente SQL Server para evaluar una directiva periódicamente.

© JMA 2009. All rights reserved

DBCC: Instrucciones de mantenimiento

Comando	Descripción
DBREINDEX	Regenera uno o varios índices de una tabla de la base de datos especificada.
DBREPAIR	Quita una base de datos dañada (obsoleta). utilizar DROP DATABASE
INDEXDEFRAG	Desfragmenta los índices agrupados y secundarios de la tabla o la vista especificada.
SHRINKDATABASE	Reduce el tamaño de los archivos de datos de la base de datos especificada.
SHRINKFILE	Reduce el tamaño del archivo de datos o de registro especificado de la base de datos relacionada.
CLEANTABLE	Recupera espacio correspondiente a columnas de longitud variable y a columnas de texto que se han quitado.
UPDATEUSAGE	Informa y corrige las imprecisiones de la tabla sysindexes, que pueden provocar que el procedimiento almacenado del sistema sp_spaceused informe incorrectamente del uso del espacio.

© JMA 2009. All rights reserved

DBCC: Instrucciones de validación

Comando	Descripción
CHECKALLOC	Comprueba la coherencia de las estructuras de asignación de espacio en disco para una base de datos determinada.
CHECKCATALOG	Comprueba la coherencia de las tablas del sistema y de éstas entre sí, en la base de datos especificada.
CHECKCONSTRAINTS	Comprueba la integridad de una restricción especificada o de todas las restricciones de una tabla determinada.
CHECKDB	Comprueba la asignación y la integridad estructural de todos los objetos de la base de datos especificada.
CHECKFILEGROUP	Comprueba la asignación y la integridad estructural de todas las tablas (de la base de datos actual) del grupo de archivos especificado.
CHECKIDENT	Comprueba el valor de identidad actual de la tabla especificada y, si fuera necesario, corrige el valor de identidad.
CHECKTABLE	Comprueba la integridad de las páginas de datos, de índices y de columnas text, ntext e image de la tabla especificada o la vista indexada.
NEWALLOC	Idéntica a DBCC CHECKALLOC