

# SQL Server

## Sumario

- 1 Arquitectura de almacenamento
  - ◆ 1.1 Páxinas e extensións
  - ◆ 1.2 Ficheiros e grupos de ficheiros
    - ◇ 1.2.1 Páxinas de arquivo de datos
    - ◇ 1.2.2 Grupos de arquivos
- 2 Bases de datos SQL Server
- 3 Tipos de datos SQL Server
- 4 Niveis de aillamento
- 5 Índices
- 6 Copias de seguridade SQL Server
  - ◆ 6.1 Copia de seguridade completa
  - ◆ 6.2 Copia de seguridade diferencial

## Arquitectura de almacenamento

SQL Server organizase a partires de ficheiros no SO, e cando creamos unha BD, definiremos o nome e localización dos mesmos.

### Páxinas e extensións

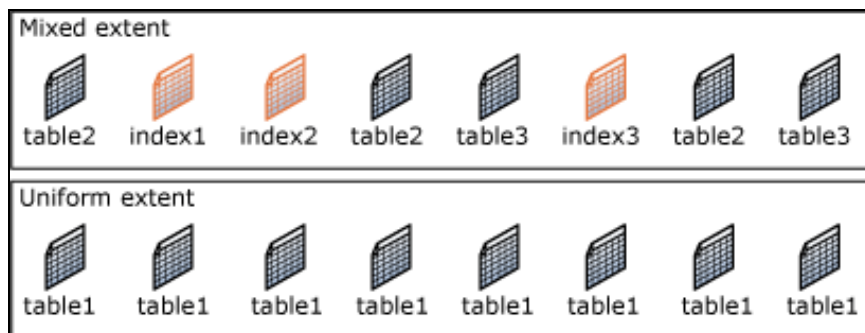
Os ficheiros da BD estarán organizados en bloques de 8Kb chamados **páxinas**, que a súa vez se organizarán en **extensións** de 64Kb de tamaño. Normalmente as páxinas asignase para un único propósito (táboa, índice, etc). SQL server distingue varios tipos de páxinas:

- Páxinas de datos: Almacenarán todos os tipos de datos salvo aqueles para tipos moi longos como text, ntext, xml, etc. A maioría das páxinas da BD serán de este tipo
- Páxinas de índices: Almacenarase os índices dos datos, que normalmente consistirán na clave ou claves do índice máis certa información adicional utilizada para localizar os datos.
- Páxinas de obxectos moi grandes: Posto que SQL server non pode distribuir unha fila en varias páxinas de datos, as filas moi grandes deben almacenarse dun xeito alternativo. Tipos de datos como text, ntext, xml, etc almacenarán un enderezo na páxina de datos apuntando a onde están realmente os datos.
- Outras páxinas: Existen outro tipo de páxinas que SQL utilizará para almacenar datos referentes de *Global Allocation Map*, *Index Allocation Map*, etc.

As filas de datos situanse dentro da páxina secuencialmente, empezando xusto despois da cabeceira. Ao final da táboa teremos unha táboa de desprazamentos das filas, con unha entrada por cada fila indicando a posición relativa da mesma dentro da páxina. As entradas de esta táboa estarán en orde inverso a secuencia de filas reais dentro da páxina. Todos os datos almacenaríanse nas páxinas de datos (máximo de 8Kb) excepto os tipos Text/Image.

Unha extensión é a unidade básica de xestión do espazo. Estará formada por 8 páxinas contiguas, ou 64Kb. Isto significa que teremos 16 extensións por Mb. SQL dispón de 2 tipos de extensións:

- Extensións uniformes asignadas a un obxecto único; as 8 páxinas poden ser utilizadas unicamente polo obxecto que as posúe.
- Extensións mixtas compartidas por ata 8 obxectos.



Unha táboa nova ou un índice xeneralmente asigna extensións mixtas, e no momento en que medra máis de 8 páxinas cambia a unha extensión uniforme, ademais as seguintes extensións asignadas tamén serán uniformes. Se a táboa ou o índice xa ocupa máis de 8 páxinas no momento da creación xa será uniforme dende un principio.

## Ficheiros e grupos de ficheiros

SQL Server posúe 3 tipos de ficheiros:

- **Primarios:** É o punto de partida da BD; apuntará a outros ficheiros da BD. Cada BD terá un único ficheiro primario. Terán a extensión \*.mdf.
- **Secundario:** Os ficheiros secundarios, xunto co primario, formarán todos os ficheiros de datos. Algunhas BD poden non conter ningún ficheiro secundario. Terán a extensión \*.ndf.
- **Rexistro:** Información necesaria utilizada en caso de ter que recuperar a BD. Polo menos debe haber un ficheiro de log por cada BD, inda que pode haber máis de un. Terán a extensión \*.ldf.

A localización de todos os ficheiros da BD estará no ficheiro primario e na [base de datos mestra](#).

Os arquivos da BD terán 2 nomes:

- **logical\_file\_name:** É o nome utilizado para facer referencia ao arquivo nas instrucións Transact-SQL.
- **os\_file\_name:** É o nome do arquivo físico incluíndo a ruta de acceso ao cartafol.

## Páxinas de arquivo de datos

As páxinas dun arquivo de datos SQL están numeradas secuencialmente dentro do ficheiro, comenzado por cero para a primeira páxina do arquivo. Cada arquivo da BD ten un número identificador único. Por tanto para identificar unicamente unha páxina dentro da BD precisaremos o número do ficheiro e o número da páxina. O seguinte exemplo mostra os números de páxina dunha BD que ten un arquivo principal de 4MB e un arquivo secundario de 1Mb

A primeira páxina de cada arquivo é especial, contendo información acerca dos atributos do ficheiro. Unha das páxinas do sistema almacenadas no arquivo de datos principal e no arquivo de rexistro principal é unha páxina de inicio da BD con información acerca dos atributos da mesma.

## Grupos de arquivos

Os arquivos da BD poden estar agrupados para facilitar a súa administración. Teremos 2 tipos de grupos:

- **Principal:** Conterá o arquivo de datos principal e os demais arquivos non asignados especificamente a outro grupo. Todas as páxinas das táboas do sistema estarán asignadas ao grupo primario.
- **Definido polo usuario:** Calquera grupo definido mediante a clausula FILEGROUP nunha instrución SQL CREATE DATABASE ou ALTER DATABASE. Os ficheiros de rexistro ou log nunca formarán parte de ningún grupo. Ningún ficheiro poderá ser membro de máis de un grupo. As táboas, os índices ou obxectos moi grandes poden asociarse cun grupo específico, nese caso todas as súas páxinas pertencerán a ese grupo.

[Ver documentación arquitectura de almacenamento](#)

## Bases de datos SQL Server

### Bases de datos do sistema

Nome	Descrición
master	Mantén información de todas as bases de datos instaladas no servidor. Modifícase cada vez que creamos unha BD nova, se xestiona unha conta ou se cambian os parámetros de configuración. Debe facerse unha copia de seguridade regularmente.
model	Proporciona unha plantilla para as novas BD. Se queremos que unha nova BD teña certas propiedades ou permisos debemos incluír estes cambios nesta BD e todas as novas BD os herdarán.
tempdb	Espacio temporal para o procesamento de consultas e outras tarefas. Esta BD recrease cada vez que arrancam SQL Server baseándose na BD model.
msdb	Utilizada polos servizos SQL Server Agent para manexar alertas, notificacións e tarefas programadas.

# Tipos de datos SQL Server

Tipos de datos SQL Server

## Niveis de aillamento

- READ UNCOMMITTED
- READ COMMITTED
- REPEATABLE READ
- SNAPSHOT: Especifica que os datos lidos por calquera instrución dunha transacción sexan a versión coherente, desde o punto de vista transaccional, dos datos existentes ao comezo da transacción. A transacción unicamente pode recoñecer as modificacións de datos confirmadas antes do comezo da mesma. As instrucións que se executen na transacción actual non verán as modificacións de datos efectuadas por outras transaccións despois do inicio da transacción actual. O efecto é o mesmo que se obtería se as instrucións dunha transacción obtivesen unha instantánea dos datos confirmados tal como atopábanse ao comezo da transacción.
- SERIALIZABLE

[Ver niveis de aillamento SQL Server](#)

## Índices

Un índice é unha estrutura de disco asociada cunha táboa ou unha vista que acelera a recuperación de filas da táboa ou da vista. Un índice contén claves xeradas a partir dunha ou varias columnas da táboa ou a vista. Ditas claves están almacenadas nunha estrutura (árbore b) que permite que SQL Server busque de forma rápida e eficiente a fila ou filas asociadas aos valores de cada clave.

Unha táboa ou unha vista pode conter os seguintes tipos de índices:

- **Agrupado:** ordenan e almacenan as filas dos datos da táboa ou vista de acordo cos valores da clave do índice. Son columnas incluídas na definición do índice. Só pode haber un índice clúster por cada táboa, porque as filas de datos só poden estar ordenadas dunha forma. A única ocasión na que as filas de datos dunha táboa están ordenadas é cando a táboa contén un índice clúster. Cando unha táboa ten un índice clúster, a táboa denomínase táboa agrupada. Se unha táboa non ten un índice clúster, as súas filas de datos están almacenadas nunha estrutura sen ordenar denominada chea.
- **Non agrupado:** Teñen unha estrutura separada das filas de datos. Un índice non agrupado contén os valores de clave de índice non agrupado e cada entrada de valor de clave ten un punteiro á fila de datos que contén o valor crave. O punteiro dunha fila de índice non agrupado cara a unha fila de datos denomínase localizador de fila. A estrutura do localizador de filas depende de se as páxinas de datos están almacenadas nunha chea ou nunha táboa agrupada. Se están nunha chea, o localizador de filas é un punteiro cara á fila. Se están nunha táboa agrupada, o localizador de fila é a clave de índice agrupada. Pode agregar columnas sen clave ao nivel folla dun índice non agrupado co fin de eludir os límites existentes para as claves de índice, 900 bytes e columnas de 16 claves, así como para executar consultas indizadas e totalmente cubertas. Para obter máis información, vexa Índice con columnas incluídas.

Tanto os índices agrupados como o non agrupados poden ser únicos. Isto significa que dúas filas non poden ter o mesmo valor para a clave de índice. Pola contra, o índice non é único e varias filas poden compartir o mesmo valor de clave.

[Ver documentación sobre índices](#)

[Sintaxe do CREATE INDEX de SQL Server](#)

## Copias de seguridade SQL Server

O ámbito dunha copia de seguridade de datos pode ser a base de datos completa, parcial ou un conxunto de arquivos ou grupos de arquivos. En todos estes casos, SQL Server admite copias de seguridade completas e diferenciais:

### Copia de seguridade completa

Unha copia de seguridade completa inclúe todos os datos dunha base de datos determinada ou un conxunto de grupos de arquivos ou arquivos, así como unha cantidade suficiente do rexistro como para permitir a recuperación de datos.

## Copia de seguridade diferencial

Unha copia de seguridade diferencial baséase na última copia de seguridade completa dos datos. Esta denomínase base da copia de seguridade diferencial ou base diferencial. Unha base diferencial é unha copia de seguridade completa de datos de lectura/escritura. **Unha copia de seguridade diferencial inclúe só os datos que cambiaron desde a última base diferencial.** Polo xeral, as copias de seguridade diferenciais que se realizan pouco despois da súa base son máis pequenas e rápidas, aforrando así tempo con respecto á copia de seguridade completa. Por tanto, o uso de copias de seguridade diferenciais acelera o proceso de realización de copias de seguridade frecuentes e reduce o risco de perda dos datos. Polo xeral, unha base diferencial utilízase en varias copias de seguridade diferenciais sucesivas. No momento da restauración, restáurase primeiro a copia de seguridade completa, seguida da copia de seguridade diferencial máis recente. A medida que a base de datos actualízase, a cantidade de datos que inclúen as copias de seguridade diferenciais aumenta. Desta forma, o proceso de creación e restauración da copia de seguridade é máis lento. Co tempo, deberase crear outra copia de seguridade completa para obter unha nova base diferencial que se utilizará noutra serie de copias de seguridade diferenciais.

[Ver documentación backups](#)