

La arquitectura del Gestor de Base de datos SQL Server

La arquitectura interna de las bases de datos en SQL Server está compuesta por 2 tipos de estructura, la estructura lógica y la estructura física. Es muy importante conocer cómo es que estas estructuras están compuestas y cuál es la relación que tienen los objetos de base de datos con cada una de estas estructuras.

Estructura Lógica:

Desde el punto de vista lógico, la base de datos debe tener al menos 1 «FileGroup» el cual contiene a toda la meta data de la misma base de datos, es decir tablas y vistas de sistema, a este «FileGroup» inicial se le conoce como «Primario» y está presente en todas las bases de datos. Todos los objetos de usuario que contengan data ya sean tablas o índices, deben estar ligados a un «FileGroup», esto se puede definir al momento de ejecutar la sentencia DDL de creación del objeto, si no se indica a que «FileGroup» estará ligado ese objeto, este pertenecerá al «FileGroup» por defecto definido en la base de datos. La base de datos solo puede tener definido 1 solo default «FileGroup».

Las bases de datos pueden tener hasta 32767 «FileGroups» definidos, según los límites establecidos para la última versión de SQL Server, la cual es SQL Server 2008 R2. Uno de los propósitos de los «FileGroups» es poder distribuir la data a través de varios discos duros físicos, de esta manera se puede obtener mayor rendimiento en las operaciones de I/O debido a que más de un disco trabajara al mismo tiempo. Otro de los propósitos es poder esconder la ubicación física real de la información a los programadores, ya que para ellos la tabla «X» pertenece al «FileGroup» «A», pero no saben en que data files físicamente se encuentra la información de la tabla «X».

Los «FileGroups» pueden contener 1 o más «Datafiles», y cada uno de estos data files se puede encontrar en unas discos diferentes, lo cual también agilizará las consultas y los ingresos de información a las tablas que se encuentren asignadas a este «FileGroup», debido a que SQL Server distribuirá la información uniformemente a través de todos los «DataFiles» del «FileGroup».

Estructura Física:

Desde el punto de vista físico, como ya hemos visto, tenemos los «DataFiles» que los en realidad los archivos de datos, es decir donde se guarda toda la información de la base de datos. Un «DataFile» solo puede pertenecer a 1 «FileGroup».

Internamente los «DataFiles» están divididos en «Extends» y estos a su vez en «Pages». Las «Pages» son la unidad mínima de almacenamiento dentro de la base de datos. Un «Page» tiene 8 Kb de tamaño en espacio de disco. Un «Extend» tiene 8 «Pages» contiguas que lo conforman, es decir, un «Extend» tiene como tamaño 64 Kb de espacio en disco.

En un «Page» solo puede haber información de 1 sola tabla, es decir el espacio de un «Page» no es compartido entre tablas o índices. En el caso de los «Extends», estos pueden ser de dos tipos:

- «Mixed»: Los cuales son compartidos hasta por 8 objetos, uno por cada «Page».
- «Uniform»: Los cuales solo pertenecen a un solo objeto, es decir que todos los «Pages» pertenecen a un solo objeto.

Normalmente cuando se crea una nueva tabla esta es asignada a un «Extend» de tipo «Mixed», hasta alcanzar la utilización de hasta 8 «Pages», a partir de ese momento se asignan «Extends» de tipo «Uniform» para optimizar el uso del espacio en la tabla.

Los «DataFiles» normalmente tienen 2 extensiones de archivo, las cuales son estándar mas no obligatorias, la extensión «mdf» que se utiliza para el primer «Datafile» perteneciente al «FileGroup» primario, y la extensión «ndf» que se utiliza para los demás data files que se agregan posteriormente a los demás «FileGroups» de la base de datos.

En el caso del «LogFile», este no pertenece a un «FileGroup» en específico, en cambio archivo este ligado directamente a la base de datos. Las bases de datos de SQL Server solo pueden tener un solo «LogFile» activo al mismo tiempo, si bien se pueden crear múltiples «LogFiles» en la base de datos, solo uno podrá ser escrito, ya que solo uno puede estar activo, cuando este archivo se llene, la base de datos pasará a escribir al siguiente archivo de transacciones, y así sucesivamente. Por esta razón no es muy conveniente ni útil tener más de un «LogFile».

En conclusión, espero que sea de ayuda estas explicaciones sobre la arquitectura de una base de datos de SQL Server, si desean temas por favor no duden en solicitarlo, haré lo posible para poder cubrir los temas solicitados en el más corto tiempo.