

# Presentación de SQL Server

El objetivo de este capítulo es presentar SQL Server de manera global y adquirir una visión general de SQL Server en su conjunto, a saber:

- entender el concepto de SGBDR y el modo de funcionamiento cliente/servidor,
- presentar los componentes de SQL Server y las plataformas de ejecución,
- presentar la arquitectura de administración y de programación,
- presentar el concepto de base de datos y las bases de datos instaladas en el servidor SQL.

SQL Server es un SGBDR (Sistema de Gestión de Base de Datos Relacional) completamente integrado en Windows, lo que permite realizar numerosas simplificaciones a nivel de administración, ofreciendo un máximo de posibilidades.


## 1. ¿Qué es un SGBDR?

SQL Server es un Sistema de Gestión de Base de Datos Relacionales (SGBDR), lo que le confiere una gran capacidad de gestionar los datos, conservando su integridad y su coherencia.



SQL Server se encarga de:

- almacenar los datos,
- verificar las restricciones de integridad definidas,
- garantizar la coherencia de los datos que almacena, incluso en caso de error (parada repentina) del sistema,
- asegurar las relaciones entre los datos definidos por los usuarios.

Este producto está completamente integrado en Windows a varios niveles:

- **Observador de eventos:**  se utiliza el diario de las aplicaciones para registrar los errores generados por SQL Server. Windows centraliza la gestión de errores, lo que facilita el diagnóstico.

**Nota:** en esta obra se utiliza el término genérico **diario** para referirse a los archivos de Traza, LogFiles o archivos de Log.

- **Analizador de rendimientos:**  mediante la adición de numerosos contadores, es sencillo detectar los cuellos de botella y reaccionar de manera más adecuada para evitar estos problemas. Se utiliza toda la potencia del analizador de rendimientos y es posible, dentro de la misma herramienta, tener los contadores tanto sobre SQL Server como sobre Windows y, de esta manera, poder detectar cuál es y dónde se encuentra el verdadero problema.
- **Tratamientos en paralelo:**  SQL Server es capaz de aprovechar las capacidades de las arquitecturas multiprocesador. Cada instancia de SQL Server dispone de su propio proceso de ejecución y los threads de Windows o bien los hilos (si la opción está activada) se ejecutan con el objetivo de explotar al máximo la arquitectura material disponible. Cada instancia de SQL Server ejecuta siempre varios threads Windows. Para tener en cuenta todos los procesadores presentes en el sistema, el parámetro de configuración **max degree of parallelism** debe tener el valor 0. Es el valor por defecto. Para evitar la generación de ejecuciones en paralelo, es suficiente con poner a 1 el valor de este parámetro. Por último, asignando a este parámetro un valor comprendido entre 1 y el número de procesadores, es posible limitar el grado de paralelismo. El valor máximo soportado por el parámetro es 64.
- **Seguridad:** SQL Server es capaz de basarse completamente sobre la seguridad gestionada por Windows, con el objetivo de permitir a los usuarios finales tener un único nombre de usuario y una única contraseña. No obstante, SQL Server gestiona su propio sistema de seguridad para todos los clientes no Microsoft.
- Se utilizan los **servicios Windows** para la ejecución de los componentes de software correspondientes al

servidor. La gestión del servidor (parada, inicio y suspensión) es más fácil y es posible disfrutar de todas las funcionalidades asociadas a los servicios de Windows (inicio automático, ejecución en el contexto de una cuenta de usuario de dominio...).

- **Active Directory:** los servidores SQL 2008 y sus propiedades se registran automáticamente en el servicio de directorio Active Directory. De esta manera es posible efectuar búsquedas en Active Directory para localizar las instancias SQL Server que están funcionando.

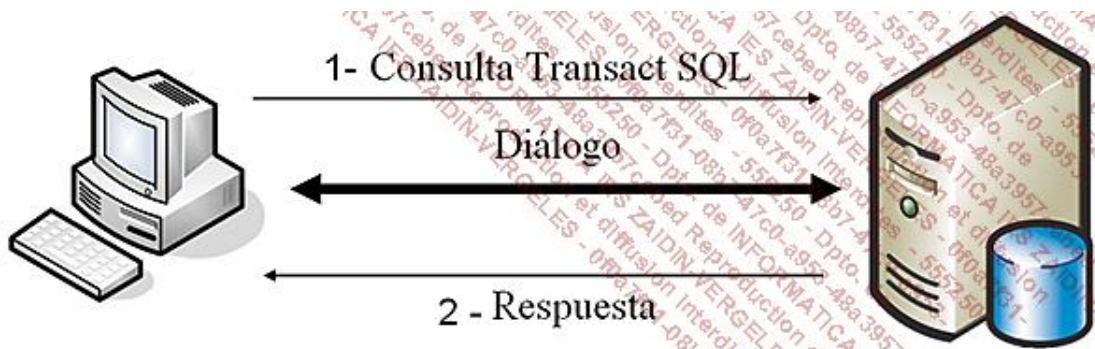
SQL Server puede gestionar dos tipos de bases de datos diferentes:

- las bases **OLTP** (*OnLine Transactional Processing*) que corresponden a las bases de datos en las cuales la información se almacena de manera directa, con el objetivo de reutilizarla más tarde tal y como fue almacenada.
- las bases **OLAP** (*OnLine Analytical Processing*) que contienen información estadística con el objetivo de poder extraer información en forma de cubos multidimensionales para ayudar en la toma de decisiones, por ejemplo. Las estadísticas contenidas en las bases OLAP están basadas en la información contenida en una base OLTP.



## 2. Modo de funcionamiento cliente/servidor

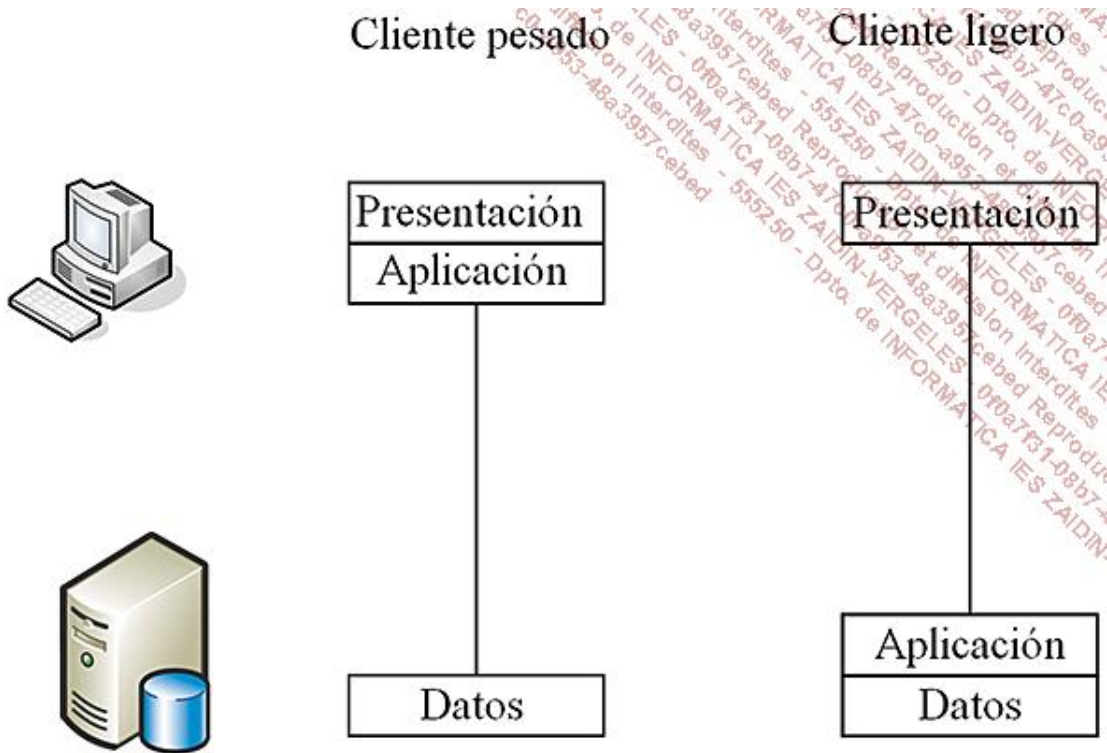
Todas las aplicaciones que utilizan SQL Server para gestionar los datos se apoyan en una arquitectura cliente/servidor. La aplicación cliente está encargada de la puesta en marcha de la interfaz de usuario. Esta aplicación se ejecuta generalmente en varios puestos clientes al mismo tiempo. En lo que respecta al servidor, este está encargado de la gestión de los datos y del reparto de los recursos del servidor entre las diferentes solicitudes (consultas) de los clientes. Las reglas de gestión de la empresa se reparten entre el cliente y el servidor.



*Modo de funcionamiento cliente/servidor*

Podemos distinguir tres casos:

- las reglas se implementan completamente en el cliente, llamado **cliente pesado**. Esta solución permite liberar recursos a nivel del servidor, aunque aparecen problemas de actualización de los clientes y de desarrollo de otras aplicaciones.
- las reglas se definen completamente en el servidor. El cliente es, por lo tanto, un **cliente ligero**. Esta solución permite obtener clientes que consumen pocos recursos materiales y una centralización de las reglas, lo que flexibiliza las actualizaciones. Sin embargo, el servidor consume muchos recursos y el nivel de interacción con el usuario corre el riesgo de degradarse, ya que el conjunto de restricciones se verifica en el momento en que el usuario envía su solicitud (consulta) al servidor.
- las reglas de la empresa se definen en una tercera máquina, llamada **Middle Ware**, con el objetivo de disminuir los recursos utilizados por el cliente y el servidor, conservando en todo momento la centralización de las reglas.



La arquitectura cliente/servidor permite un despliegue óptimo de las aplicaciones clientes en numerosos puestos siempre conservando una gestión centralizada de los datos (en el servidor), lo que hace posible compartir la información en el interior de la empresa.

También es posible tener varias aplicaciones cliente en el mismo servidor de base de datos. Esta posibilidad ofrece numerosas funcionalidades, aunque debe asegurarse que la carga de trabajo a la que se somete el servidor no sea demasiada para las capacidades de la máquina. Esta arquitectura cliente/servidor se respeta en todas las herramientas que permiten acceder a la información contenida en el servidor SQL, por lo tanto, también en las herramientas de administración, aunque estas no estén instaladas en el servidor.

**Todas las peticiones que provienen de los clientes y se dirigen al servidor deben escribirse en Transact-SQL.** Este lenguaje de consulta de base de datos respeta la norma ANSI SQL-92. SQL proporciona un conjunto de comandos para gestionar los objetos y manipular los datos de las bases de datos. Transact SQL está enriquecido con numerosas funcionalidades, no normalizadas, con el objetivo de ampliar las capacidades del servidor. De esta manera, es posible definir procedimientos almacenados en el servidor.

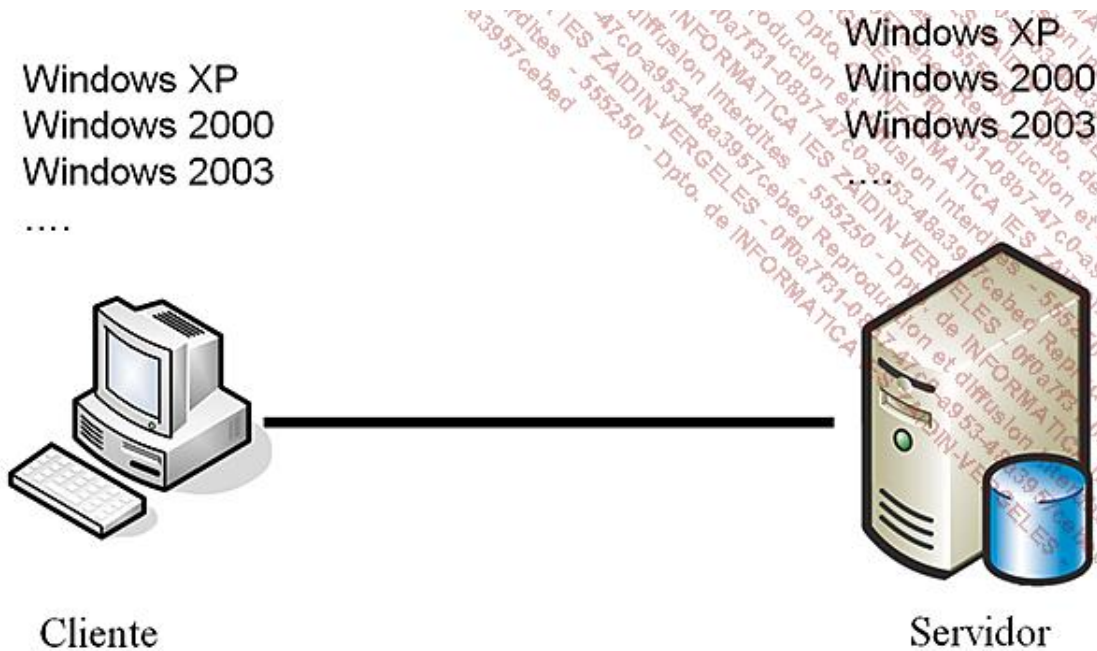
### 3. Las posibles plataformas

Es importante distinguir dos casos: por un lado las plataformas posibles para el cliente y por otro las plataformas para el servidor.

Las plataformas cliente que se presentan aquí son los puestos sobre los que se pueden instalar las herramientas de administración SQL Server. No se trata de los puestos que alojan una aplicación que se conecta a una instancia SQL Server para gestionar los datos.

De una manera sintética, las herramientas cliente de administración pueden instalarse sobre todos los sistemas operativos Windows 2003, Windows XP Pro o cualquier sistema posterior.

Por el contrario, para la parte servidor, las disponibilidades en términos de plataforma dependen de la edición SQL Server elegida. Sin embargo, para acoger una instancia de base de datos en producción, es necesario disponer de un servidor de rendimiento fiable. Por lo tanto, se recomienda una plataforma Windows 2003. La edición de Windows 2003 se escogerá en función de las restricciones impuestas por la edición SQL Server elegida y de las restricciones relacionadas con el entorno técnico. La instalación de una instancia bajo Windows XP se reservará para puestos portátiles.



En el caso en el que el cliente tenga instalada una aplicación específica, la gama de plataformas aumenta significativamente, en particular, gracias a que el controlador **JDBC** permite el acceso a una instancia de SQL Server desde una aplicación escrita en java. La gama se ampliará en el caso de una aplicación **ASPX** que ofrezca una interfaz web. Un simple navegador de Internet permite iniciar la aplicación.

#### 4. Los componentes de SQL Server

El **motor de base de datos de SQL Server** o Database Engine está compuesto por varios programas. Algunos se ejecutan en forma de servicios y otros tienen una interfaz de usuario gráfica o una línea de comandos.

##### Componentes Servidor

SQL Server se ejecuta en forma de servicios Windows. Según las opciones de instalación elegidas, puede tener más o menos servicios. Los principales servicios son:

- **SQL Server:** es el servidor de base de datos propiamente dicho. Si no se inicia este servicio, no es posible acceder a la información. Gracias a este servicio, SQL Server asegura la gestión de las consultas de usuario. Se hace referencia a este servicio con el nombre **MSSQLSERVER** para la instancia por defecto y **MSSQLSERVER \$nombreInstancia** en el caso de cualquier otra instancia con un nombre dado.
- **SQL Server Agent:** este servicio se encarga de la ejecución de las tareas planificadas, la vigilancia de SQL Server y el seguimiento de las alertas. Está directamente unido a una instancia de SQL Server. En el gestor de servicios, se hace referencia a él con el nombre **SQL Server Agent(MSSQLSERVER)** para la instancia por defecto y **SQL Server Agent(nombreInstancia)** en el caso de cualquier otra instancia con un nombre dado.
- **Microsoft Full Text Search:** este servicio está encargado de gestionar la indización de los documentos de tipo texto almacenados en SQL Server y gestionar igualmente las búsquedas por palabras clave.



Es posible instalar varias instancias de SQL Server en el mismo puesto.

## **Conectividad Cliente**

La instalación de los componentes de conectividad en los puestos cliente permite realizar la gestión de la red, la DB Library para los programas en acceso nativo, el soporte OLE-DB y ODBC.



## **Herramientas de gestión**

Las tareas de administración se realizan mediante la utilización de herramientas. La mayor parte de estas herramientas tienen una interfaz gráfica intuitiva y de fácil manejo. Sin embargo, las tareas administrativas deben meditararse muy bien antes de realizarse. La utilización de algunas herramientas implica que el componente servidor correspondiente esté instalado.

Estas herramientas son:

- **SQL Server Management Studio** para realizar todas las operaciones relativas al servidor de base de datos.
- **SQL Server Configuración Manager** para gestionar los servicios unidos a SQL Server.
- **SQL Server Profiler** para el seguimiento y análisis de la carga de trabajo de una instancia SQL Server.
- **Database Engine Tuning Advisor** para optimizar el funcionamiento del servidor de base de datos.

Además de estas herramientas, SQL Server contiene **Business Intelligence Development Studio** para la programación de trabajos que se inscriben dentro del marco del análisis multidimensional de los datos.

Por último, todas las herramientas y el funcionamiento de SQL Server están perfectamente documentados.

## **Los componentes**

Las diferentes **piezas de software** proporcionadas por SQL Server se articulan siempre en torno al motor de base de datos relacionales, que trata de manera eficiente la información almacenada en formato relacional y en formato xml.

- **SQL Server Analysis Service** permite un potente análisis de los cubos de datos definidos por medio de Business Intelligence Development Studio.
- **SQL Server Integration Service (SSIS)** es una herramienta de importación y exportación de datos fácil de utilizar, al mismo tiempo que altamente parametrizable.
- **Reporting Services** permite elaborar informes de análisis de los datos.
- **La replicación** de los datos en diferentes instancias permite situar los datos lo más cerca posible de los usuarios y reducir los tiempos de procesamiento.
- **Service Broker** permite el trabajo en modo asíncrono y facilita de esta manera la gestión de los picos de fuerte actividad almacenando las peticiones de trabajo antes de tratarlas.
- **La integración de CLR** en SQL Server permite desarrollar procedimientos y funciones utilizando los lenguajes VB.Net y C#. La integración de CLR no pretende sustituir a Transact SQL aunque se presenta como un complemento con el objetivo de poder realizar una codificación simple y de buen rendimiento para el conjunto de las funcionalidades que deben estar presentes en el servidor.
- **Los puntos de destino http** permiten a SQL Server albergar sus propios servicios y facilitar de esta manera la integración del servidor en un contexto heterogéneo.



## **Memoria AWE**

La puesta en marcha del API AWE consigue una gestión de memoria más eficaz. Permite gestionar, en los sistemas 32 bits, más de 4 Gb de memoria. La edición enterprise es capaz de gestionar hasta 64 Gb de memoria. La utilización de AWE es posible mediante la activación de la opción de configuración awe enabled con sp\_configure.



awe enabled es una opción de configuración avanzada.



En el caso en el que la gestión de la memoria AWE esté activada, SQL Server puede, gracias a Windows 2003, encargarse de la adición de memoria en caliente. Esto es posible únicamente si la plataforma utilizada permite realizar una operación como esta.

➤ En caso de que SQL Server se ejecute en una plataforma Windows Server 2003 o 2008, la gestión de la memoria AWE es dinámica. Sin embargo, en el caso de una ejecución bajo Windows 2000, la gestión de esta memoria es estática.

### **Reporting Services**

**Reporting Services** permite la creación de informes para presentar de la mejor manera posible la información contenida en SQL Server. Estos informes, almacenados en un servidor IIS, se pueden diseñar gracias al generador de informes de Business Intelligence Development Studio.



Los informes están disponibles en formato HTML aunque pueden existir en formato pdf. La gestión de estos informes a nivel de seguridad, planificación de las generaciones sucesivas... está asegurada por medio de Reporting Services.



### **Analysis Services**

**Analysis Services** es una herramienta que permite construir cubos multidimensionales de análisis de datos. Algunas funcionalidades OLAP ya estaban presentes en las versiones anteriores de SQL Server, aunque con Analysis Services SQL Server permite realizar análisis completos de los cubos de análisis que puedan estar definidos.

Para mejorar el rendimiento en el procesamiento de los cubos, es posible instalar varias instancias del motor Analysis Service en el mismo servidor.

Analysis Service se basa enteramente en la interfaz de Business Intelligence Development Studio.

### **Puntos de destino http**



Los **puntos de destino http** permiten a SQL Server responder de manera directa a las consultas http, es decir, sin pasar por un servidor IIS. SQL Server ofrece de esta manera la posibilidad de exponer los procedimientos y funciones a través de servicios Web.

➤ Los puntos de destino http sólo están disponibles si SQL Server se ejecuta en una plataforma Windows 2003 o Windows 2008.

### **Service Broker**

**Service Broker** permite una gestión asíncrona de las consultas. Más exactamente, Service Broker permite a una aplicación cliente enviar numerosas peticiones de servicios y SQL Server puede tratar estas peticiones (mensajes) una detrás de otra. La cola de mensajes permite regular la carga de trabajo en el servidor y absorber ciertos momentos de actividad puntuales.

Service Broker dispone de un mecanismo seguro que le permite garantizar el tratamiento de los mensajes. Service Broker utiliza SQL Server para conservar la cola de espera de los mensajes que aún no han sido tratados.

### **CLR**

La integración de **CLR (Common Language Runtime)** en SQL Server permite aumentar considerablemente las posibilidades ofrecidas en términos de programación. La presencia de CLR no afecta a Transact SQL. Ambos son complementarios. Transact SQL es perfecto para escribir los procedimientos o funciones para los que hay un tratamiento intensivo de los datos. Por el contrario, en los casos en que el volumen de datos manipulados no es grande, CLR permite escribir con sencillez tratamientos complejos, ya que se beneficia de toda la riqueza de CLR.

CLR permite igualmente definir sus propios tipos de datos o bien nuevas funciones de cálculo agregado.

Por último, CLR permite a los desarrolladores de aplicaciones desarrollar los procedimientos y funciones en SQL Server conservando sus lenguajes favoritos (VB.Net o C# por ejemplo), y por lo tanto, sin necesidad de dominar Transact SQL.

En caso de que el código esté escrito en Visual Studio, la integración de la versión compilada en SQL Server y el mapeo CLR-Transact SQL se realizan de manera automática. Es posible realizar el desarrollo sin utilizar Visual Studio, pero la integración con SQL Server se realizará de manera manual, lo que es una tarea tediosa.

