

# Características generales de los SGBD

Conexiones en la red. Paralelismo. Distribución. *Performance*. Escalabilidad. Transparencia. Confiabilidad. Procesamiento de transacciones en línea. Procesamiento para toma de decisión.

La arquitectura de un SGBD está influida en gran medida por el sistema informático subyacente en el que se ejecuta la BD. En la arquitectura se reflejan aspectos que condujeron al desarrollo de diferentes estructuras:

- Conexión en la red
- Arquitectura centralizada
- Paralelismo
- Distribución

No confundir el SGBD con la arquitectura que se elige para implementarlo. Algunos SGBD solo se pueden implantar en una de las arquitecturas y otros en todas ellas.

## Conexión en la red

La conexión de varias computadoras en red permite que algunas tareas se ejecuten en un sistema servidor y que otras se ejecuten en sistemas clientes.

Esta división de trabajo condujo al desarrollo de bases de datos cliente-servidor.

El procesamiento en una arquitectura cliente-servidor, dentro de los DBMS, facilita el acceso web a las bases de datos, por lo tanto cualquier LAN o WAN puede considerarse un sistema cliente/servidor desde el momento en que el cliente solicita servicios (datos, impresión, etc.).

## Arquitectura centralizada

El SGBD está implementado en una sola plataforma (ordenador) desde donde se gestiona directamente, de modo centralizado, la totalidad de los recursos. Es la arquitectura más clásica de los centros de proceso de datos tradicionales. Se basa en tecnologías sencillas, muy experimentadas y de gran robustez (*hardware*).

Los DBMS corporativos hacen uso de servidores *mainframe* que usan este tipo de arquitectura.

## Paralelismo

Un DBMS paralelo usa una colección de recursos (procesadores, discos y memoria) para realizar trabajo en paralelo, junto con los servicios de una red de alta rapidez, sistema operativo y sistema de almacenamiento para coordinar la división del trabajo entre recursos. El grado de recursos compartidos determina las arquitecturas para procesamiento de bases de datos paralelas

El estándar de clasificación de arquitecturas tiene diferentes enfoques:

- SE: todo compartido, memoria y discos se comparten entre una colección de procesadores, usualmente está relacionado como una sola computadora de multiprocesamiento y no como una arquitectura de base de datos paralela. No es necesario el particionamiento porque los procesadores acceden a todos los datos.
- SD: discos compartidos, cada procesador tiene su memoria privada, pero los discos se comparten entre todos los procesadores. No es necesario el particionamiento porque los procesadores acceden a todos los datos.
- SN: nada compartido, los datos se deben particionar entre los procesadores.

Para flexibilidad adicional, las arquitecturas básicas se extienden mediante *clustering*. Un clúster es un acoplamiento firme de dos o más computadoras, de modo que se comparten como una sola computadora.

La necesidad de procesamiento de consultas en paralelo condujo al desarrollo de los sistemas de bases de datos paralelos. En todas las arquitecturas de bases de datos paralelas, el hecho de compartir recursos es transparente a las aplicaciones.

El código de aplicación (SQL y enunciados en lenguaje de programación) no necesita ser modificado para sacar ventaja del procesamiento de bases de datos paralelas.

Los conflictos de diseño que influyen en el rendimiento de arquitecturas de bases de datos paralelas son: equilibrio de carga, coherencia de caché y comunicación de interprocesadores.

El procesamiento en paralelo permite dentro una computadora acelerar las actividades del sistema de BD. Proporciona a las transacciones unas respuestas más rápidas. Da la capacidad de ejecutar más transacciones por segundo.

Las consultas pueden procesarse aprovechando el paralelismo brindado por el sistema informático subyacente.

## Distribución

Los DBMS distribuidos soportan solicitudes globales que usan datos almacenados en más de un sitio autónomo. Un sitio es cualquier computadora controlada localmente con una dirección de red única y distribuida geográficamente.

La distribución de datos en distintos sitios de una organización permite que los datos residan en donde se han generado, o en donde sean más necesarios, y continuar siendo accedidos desde otros lugares.

Las solicitudes globales son consultas que combinan datos desde más de un sitio y transacciones que actualizan datos en más de un sitio. Pueden implicar una colección de enunciados que acceden a datos locales en algunos enunciados y a datos remotos en otros.

Los datos locales están controlados por el sitio en el que un usuario normalmente se conecta; los datos remotos implican un sitio diferente en el que un usuario puede incluso no tener cuenta de acceso. Si todas las solicitudes requieren datos de un solo sitio, no se requiere capacidades de procesamiento de bases de datos distribuidas.

Las bases de datos distribuidas son potencialmente útiles para organizaciones que operan en múltiples ubicaciones con control local de recursos computacionales.

Guardar copias de la BD en diferentes sitios permite que las operaciones sobre la BD continúen procesándose aun si algún sitio deja de estar disponible por cualquier motivo (inundación, incendio, terremoto, etc.).

La necesidad de alta disponibilidad de la BD condujo al desarrollo de los sistemas de base de datos distribuidos para poder manejar datos distribuidos geográfica o administrativamente a lo largo de múltiples sistemas de bases de datos.

Para acomodar la distribución de datos son necesarias capas adicionales de descripción de datos.

La arquitectura de esquema para un DBMS distribuido firmemente integrado contiene capas adicionales para fragmentación y asignación.

El esquema de fragmentación contiene la definición de cada fragmento, mientras que el esquema de asignación contiene la ubicación de cada fragmento.

Un fragmento puede definirse como un subconjunto vertical (operación de proyecto), un subconjunto horizontal (operación de restricción) o un fragmento mezclado (combinación de operaciones de proyecto y de restricción).

Un fragmento se asigna a un sitio, aunque en ocasiones a múltiples sitios.

Si el DBMS distribuido soporta copiado, puede asignarse un fragmento a múltiples sitios. En algunos DBMS distribuidos que soportan copiado, una copia de un fragmento se considera la copia primaria y las otras copias son secundarias. Solo se garantiza que la copia primaria sea la actual.

## Sistema Gestor de Bases de Datos (SGBD)

Independientemente de la estructura, ¿qué características deben tener las bases de datos?

- *Performance*
- Transparencia
- Escalabilidad
- Disponibilidad / integridad (confiabilidad)

### *Performance*

- Capacidad para ejecutar operaciones en paralelo
- *Multi-threading*
- Tener paralelismo de *input/output*
- Capacidad para ejecutar operaciones utilitarias en paralelo (*backups/restores*) todo en línea
- Paralelismo para el soporte de aplicaciones:
  - Paralelismo de I-O de disco
  - Paralelismo de utilitarios
  - Paralelismo en consultas y procesamiento
- El particionamiento de los datos debe ser independiente de la estructura de *hardware*.
- Paralelismo de I-O de disco:
  - Posibilidad trabajar con tablas particionadas
  - El motor de BD debe permitir la ejecución concurrente de I/O
- Paralelismo de utilitarios:
  - Índices
  - *Backup*
  - *Restore*

- Paralelismo en consultas y procesamiento:
  - Es el más complejo de implementar debido a que una consulta o una modificación de los datos involucra varias operaciones atómicas y resulta complicado hacerlas en paralelo y coordinarlas.
  - El SGBD debe seleccionar operaciones atómicas las cuales pueden ser replicadas y luego ser procesadas en forma concurrente. Luego el resultado se combina como una única operación.
  - Paralelismo horizontal: es la forma de crear múltiples copias de una operación que corre en paralelo.
  - Otra forma de paralelismo: mientras una operación clasifica registros según el valor de un campo clave, en forma concurrente existe una operación que accede a los datos y no espera que estos sean procesados, dejándolos listos para su procesamiento.

## Transparencia

Para los DBMS, transparencia significa que los detalles internos de los servicios de las transacciones son invisibles, mejorando así la productividad de los programadores y analistas de bases de datos.

Los DBMS ofrecen dos servicios para garantizar que las transacciones tienen las propiedades ACID:

- Transparencia de recuperación: comprende acciones para manejar las fallas, como errores en la comunicación y colapsos del *software*. El DBMS restaura automáticamente una base de datos a un estado consistente después de una falla.
- Transparencia de concurrencia: comprende acciones para controlar la interferencia entre varios usuarios simultáneos de la base de datos. Los usuarios perciben la base de datos como un sistema para un solo usuario, aunque haya varios usuarios simultáneos.

En el procesamiento de bases de datos distribuidas, la transparencia se relaciona con la independencia de datos. Si la distribución de la base de datos es transparente, los usuarios pueden escribir consultas sin conocimiento de su distribución, esto se conoce como transparencia de fragmentación.

El procesamiento de bases de datos paralelas en las arquitecturas sin nada compartido (SN) involucra transparencia de fragmentación.

## Escalabilidad

Propiedad que tiene cualquier sistema para crecer dinámicamente en base a sus necesidades, utilizando los recursos disponibles (*hardware*), sin operaciones complementarias, al incrementar la cantidad de usuarios, procesos, etc.

La BD debe adaptarse sin inconvenientes a *mayor storage*.

Si se agrega más *hardware* a los servidores que soportan la BD, éste debe poder utilizarlo automáticamente y solapar procesos o utilizar más memoria.

La escalabilidad se analiza desde transacciones de lectura.

Dos tipos de escalabilidad:

- Escalabilidad vertical: Se escala sobre el mismo server, con una CPU más rápida o agregando CPU's, agregando una controladora más rápida, agregando más memoria.
- Escalabilidad horizontal: Se utilizan múltiples servers que trabajan en forma transparente distribuyendo la carga de trabajo.

La escalabilidad permite adaptarse rápidamente a los cambios en las organizaciones, producto de reducir a un mínimo los tiempos de planificación y adaptarse a los cambios tecnológicos.

## Integridad / disponibilidad

Las BD deben dar confiabilidad/robustez/integridad, por intermedio del *software* y del *hardware*. Dos factores a analizar:

- Administración *on-line*: Utilitarios de operación continua que permitan realizar tareas *on-line*.
- Robustez:
  - Reducir la posibilidad de fallas y realizar recuperos en forma transparente.
  - *Fault tolerant*: se llaman así a los sistemas con alta disponibilidad, implementando redundancia de *hardware* (*mirroring* de disco) y redundancia de datos (*soft mirroring* / replicación).

- Redundancia datos
  - *Soft mirroring (clusters)*:
    - Duplicar unidades de datos en discos distintos.
    - Las lecturas se pueden distribuir
    - Las escrituras son transparentes
    - Ante una falla las operaciones se realizan sobre el medio no dañado
    - Para los usuarios es totalmente transparente
    - En el medio dañado una vez reparado se deben sincronizar las copias de los dispositivos
  - Replicación:
    - Similar al *mirroring*, pero se puede localizar en forma remota.
    - Se trata de copiar los datos en diferentes lugares, puede ser en un solo equipo o en varios discos.
    - Esta forma de resguardo no asegura alta disponibilidad, intenta ganar en *performance*.

## Confiabilidad y disponibilidad

- Confiabilidad: Un SGBD es confiable si bajo las condiciones de uso determinadas por sus especificaciones cumple con su misión específica en un período determinado. Se utiliza sobre sistemas críticos. Se utiliza un parámetro crítico para evaluar una BD sobre la base de las probabilidades de ocurrencias de fallas. (MTBF *Mean-Time-Between-Failures*).
- Disponibilidad: Porción de tiempo en que la BD se encuentra apta para cumplir su misión, respecto al tiempo que debió haber cumplido su misión y no lo hizo. Para medir la disponibilidad se utiliza el parámetro MTTR (Tiempo Medio de Reparaciones) y el MTBF de la siguiente forma:

$$\text{DISPONIBILIDAD} = \text{MTBF} / (\text{MTBF} + \text{MTTR})$$

Para diseñar un SGBD distribuido confiable es necesario definir los tipos de falla que pueden ocurrir:

- Falla de transacciones: No se cumple con el modelo ACID (atomicidad – concurrencia – aislamiento - durabilidad).
- Fallas de sistema: Son las fallas propias del *hardware* (procesador, memoria, etc.) donde opera el SGBD.

- Fallas del medio de almacenamiento: Es el lugar físico donde se almacenan los datos.
- Fallas de comunicación: Son fallas en el medio de comunicación propiamente dicho que producen pérdida de mensajes.

## Modos en que trabaja una base de datos

- OLTP: Procesamiento de transacciones en línea
- DSS: Sistemas de soporte de decisión (o sistemas de asistencia de decisiones), también llamados OLAP
- OLCP: Procesamiento complejo en línea

### OLTP: Procesamiento de transacciones en línea

El procesamiento de transacciones comprende el lado operativo de las bases de datos. El motor de la base de datos está optimizado para trabajar con un conjunto de transacciones en paralelo y cortas, en otras palabras, se envían al servidor transacciones pequeñas de corta duración. Normalmente realizan la lectura de porciones de información pequeñas.

En este modo de trabajo se generan muchas operaciones de inserción y/o modificación. Los tiempos de ejecución de las transacciones son cortos.

La ejecución de reportes y toma estadísticas degradan la performance de la BD, volviendo lentas todas las operaciones. La ejecución de grandes reportes y copias de respaldo suelen realizarse en momentos de poco uso de la BD.

### DSS: Sistemas de soporte de decisión (o sistemas de asistencia de decisiones), también llamados OLAP

El motor de la base de datos está optimizado para realizar lecturas de una gran cantidad de datos, sumarización de datos, es decir soporta procesos largos que van a necesitar leer muchos datos para la generación de reportes, análisis estadísticos, etc. Generalmente realizan lectura de porciones de datos muy grandes.

En este modo de trabajo es casi nula la generación de operaciones de inserción y/o modificación.

Los tiempos de ejecución de las transacciones suelen ser importantes.

Suelen utilizarse para la ejecución de reportes, toma estadísticas.



Para este modo de trabajo suelen utilizarse imágenes relativamente estáticas de la base de datos OLTP para asegurar su integridad mientras se ejecutan los procesos largos.

#### OLCP: Procesamiento complejo en línea

Es un intermedio o un mix entre el modo OLTP y DSS, intenta alcanzar un balance entre los dos modos reflejando la necesidad de obtener:

- Alto rendimiento de la base (muy buena *performance*)
- Completa disponibilidad de los datos
- Capacidad de realizar copias de respaldo en línea (*backup* en caliente)
- Ejecución de grandes consultas e informes mientras los usuarios actualizan la información diaria de la organización
- Información disponible en todo momento, sin límite de acceso a los usuarios OLTP y DSS.
- Destinada a aquellas organizaciones que necesitan obtener informes críticos en tiempo real.