

FUNDAMENTOS DE BASES DE DATOS

- **Sistema de Información:** Un sistema de información es un conjunto de elementos diseñados para gestionar y administrar datos e información, organizados y listos para su uso. Puede ser informático o no, pero en este contexto, nos enfocamos en los Sistemas Gestores de Bases de Datos (SGBD).
- **Uso de SGBD:** Los SGBD son sistemas informáticos utilizados para gestionar y administrar bases de datos. Estos sistemas son esenciales en el desarrollo de aplicaciones que requieren almacenamiento y recuperación de datos.
- **Elección de SGBD:** No siempre es necesario utilizar un SGBD. Para aplicaciones pequeñas con pocos datos, usar un SGBD puede ser innecesario y complejo. En tales casos, un simple sistema de archivos podría ser suficiente. Sin embargo, los SGBD ofrecen ventajas como la eficiencia y la capacidad de manejar grandes cantidades de datos.
- **Funciones Deseables:** Se espera que un SGBD proporcione ciertas características deseables, como las operaciones CRUD (Crear, Leer, Actualizar, Eliminar), recuperación de información, consistencia, validez, velocidad, transacciones atómicas, persistencia, capacidad de extensión y seguridad.
- **Funciones de un SGBD:** Un SGBD cumple diversas funciones, incluyendo la creación y organización de la base de datos, control de acceso, evitación de redundancia e inconsistencia de datos, gestión de acceso concurrente, garantía de la correcta ejecución de las transacciones, y respaldo y recuperación de datos.
- **Ejemplos de SGBD:** Se mencionan ejemplos de SGBD como MySQL, Microsoft Access, Oracle, SQL Server y SQLite, cada uno con características y objetivos específicos.

Los componentes y tipos de Sistemas Gestores de Bases de Datos (SGBD).

Componentes de un SGBD:

- Los SGBD suelen dividirse en dos componentes principales: el lado Cliente y el lado Servidor, siguiendo la arquitectura Cliente-Servidor. La funcionalidad principal reside en el lado del servidor, donde se almacenan los datos, mientras que el lado cliente actúa como interfaz de comunicación entre el usuario y el motor del SGBD.
- Dentro del motor del SGBD se encuentran varios componentes, que incluyen:
- Administrador de Almacenamiento: Controla el acceso a los datos en disco y gestiona los archivos y buffers de almacenamiento.
- Procesador de Consultas: Recibe las consultas del usuario y planifica la ejecución óptima de estas consultas a través de un Plan de Ejecución, interactuando con el Administrador de Almacenamiento.
- Gestor de Transacciones: Garantiza la integridad de la base de datos después de la ejecución de transacciones, incluso en caso de fallos.

Tipos de SGBD:

- Según el modelo de datos utilizado, se destacan los siguientes tipos de SGBD, con la posibilidad de que una base de datos pueda pertenecer a varias categorías:
- 1. Relacional: Utiliza el modelo relacional, que se basa en tablas y relaciones entre ellas. Ejemplos incluyen MS Access, Oracle, Microsoft SQL Server y MySQL. Son apropiados para consultas complejas, validación de datos y relaciones entre tablas.
- 2. XML: Almacena datos en un formato jerárquico XML. Se utiliza para datos jerárquicos y transferencia de información entre aplicaciones a través de Internet. Ejemplos de herramientas incluyen xbird. Apropriadas para datos jerárquicos y cuando las herramientas XML cumplen con los requisitos.
- 3. Orientado a Objetos: Representa todos los elementos de la base de datos como objetos siguiendo el paradigma de la Programación Orientada a Objetos (POO). Ejemplos son db4o e InterSystems Caché. Adecuadas cuando la aplicación se desarrolla en un lenguaje orientado a objetos y no se requieren consultas complejas.
- 4. Objeto-Relacional: Extiende el modelo relacional para incluir características de la POO, como herencia y tipos complejos. PostgreSQL es un ejemplo. Son adecuadas para arquitecturas orientadas a objetos y consultas complejas.
- 5. NoSQL: Estos sistemas no utilizan SQL como lenguaje principal de consultas y no garantizan ACID. Destacan por su escalabilidad horizontal. Ejemplos incluyen MongoDB y Cassandra. Apropriadas para manejar grandes cantidades de datos y para escalar.

Clasificación según accesos simultáneos soportados:

- Monousuario: Estos SGBD permiten la conexión de un único usuario a la Base de Datos en un momento dado. Generalmente, no tienen controles de usuario avanzados, y la Base de Datos se asemeja a un simple archivo al que se accede desde la aplicación como cualquier otro documento. No admiten control de concurrencia ni características de sistemas multiusuarios. Ejemplos incluyen Ms Access y LibreOffice Base.
- Multiusuario: Estos SGBD permiten múltiples conexiones simultáneas a una misma Base de Datos. Ofrecen soporte para la gestión de cuentas de usuario, lo que incluye el control de concurrencia necesario en sistemas multiusuarios. Ejemplos de SGBD multiusuario son MySQL, PostgreSQL, Microsoft SQL Server, Oracle, MongoDB y Cassandra.

Clasificación según tamaño:

- SGBD Ligero: Estos son SGBD pequeños que ocupan muy poco espacio (incluso menos de 1 megabyte) y proporcionan algunas funciones esenciales. Son ideales para entornos donde no se necesita un alto rendimiento o donde la potencia de

cálculo es limitada, como dispositivos móviles. SQLite es uno de los ejemplos más conocidos en esta categoría.

- SGBD de Ofimática: Son más robustos que los SGBD ligeros y ofrecen funciones adicionales. Se utilizan comúnmente en aplicaciones de oficina para bases de datos pequeñas (generalmente unos pocos megabytes). Ejemplos en esta categoría son Ms Access y LibreOffice Base.
- SGBD de Alto Rendimiento: Estos SGBD son ideales para entornos que gestionan grandes cantidades de datos o aplicaciones que requieren funciones avanzadas, como control de usuarios, concurrencia y soporte para transacciones. Ejemplos incluyen MySQL, Microsoft SQL Server, Oracle y PostgreSQL. También se pueden considerar como SGBD de alto rendimiento a MongoDB y Cassandra, que ofrecen características diferentes pero igualmente potentes.

Tipos de Bases de Datos:

- Bases de Datos Centralizadas: Son las más comunes y se utilizan en grandes corporaciones. Todas las operaciones y datos se gestionan en un único equipo que almacena y gestiona la Base de Datos. Esto simplifica la implementación pero puede tener desventajas en cuanto a la disponibilidad y la recuperación de fallos.
 - Ventajas: Fácil implementación y diseño sencillo.
 - Desventajas: Un fallo en el equipo central afecta a todos los datos almacenados en él.
- Bases de Datos Distribuidas: Estas Bases de Datos consisten en múltiples bases de datos relacionadas lógicamente, ubicadas en diferentes lugares físicos. Pueden comunicarse a través de redes locales (LAN) o a través de Internet. Esto permite operaciones autónomas o distribuidas. Aunque son más complejas de implementar y mantener, ofrecen ventajas como la disponibilidad y la ubicación de datos.
 - Ventajas: Menor costo de hardware, mayor disponibilidad y ubicación estratégica de datos.
 - Desventajas: Mayor complejidad, desafíos de seguridad y falta de experiencia en casos raros de fallos.

Arquitectura de una aplicación:

La arquitectura de una aplicación suele seguir un modelo de capas, donde se separan lógicamente las distintas funcionalidades de la aplicación en tres capas principales:

1. Capa de Presentación: Esta capa es la interfaz de usuario que interactúa con el usuario final. Muestra la aplicación y permite que el usuario interactúe con ella.
2. Capa de Negocio: En esta capa se encuentra la lógica de la aplicación, donde se realizan procesos de tratamiento de datos y se aplican reglas comerciales. Los resultados se envían a la capa de presentación.

3. Capa de Datos: Aquí residen los datos y se conecta con el SGBD. Se encarga de recibir y enviar datos entre la capa de negocio y la Base de Datos.

En aplicaciones de tamaño medio, es común que estas capas estén separadas lógicamente, aunque no siempre físicamente. La separación física puede depender de la naturaleza de la aplicación y sus necesidades de rendimiento. La arquitectura de 3 capas es un enfoque típico, donde la capa de presentación reside en el equipo del usuario, mientras que las capas de negocio y datos pueden ubicarse en un servidor remoto o varios servidores según las demandas de rendimiento de la aplicación. Esto proporciona escalabilidad y distribución de carga cuando es necesario.

Los Sistemas de Gestión de Bases de Datos (SGBD) son software diseñados para administrar y gestionar datos de manera eficiente. Hay varios tipos de SGBD, cada uno con sus propias características y casos de uso específicos. Aquí tienes un resumen de los tipos de SGBD más comunes junto con ejemplos de cada uno:

Nombre	Características	Ejemplos
SGBD Relacionales	Almacenan datos en tablas relacionadas y utilizan SQL (Lenguaje de Consulta Estructurada) para consultar y manipular datos	<ul style="list-style-type: none">- <i>MySQL</i>- <i>PostgreSQL</i>- <i>Oracle Database</i>- <i>Microsoft SQL Server</i>
SGBD NoSQL	Están diseñados para manejar grandes volúmenes de datos no estructurados o semiestructurados. No utilizan un esquema fijo como en las bases de datos relacionales	<ul style="list-style-type: none">- <i>MongoDB</i>- <i>Cassandra</i>- <i>Couchbase</i>
SGBD de Columnas	Almacenan datos en columnas en lugar de filas, lo que los hace adecuados para consultas analíticas y procesamiento de datos a gran escala	<ul style="list-style-type: none">- <i>Apache HBase</i>,- <i>Amazon Redshift</i>
SGBD en Memoria	Almacenan datos en la memoria principal en lugar de en disco, lo que los hace extremadamente rápidos para la recuperación de datos, pero limitados en	<ul style="list-style-type: none">- <i>Redis</i>- <i>Memcached</i>

	capacidad de almacenamiento	
SGBD Orientados a Grafos	Diseñados para modelar y consultar datos con relaciones complejas. Son ideales para aplicaciones que involucran redes, recomendaciones y análisis de redes sociales	<ul style="list-style-type: none"> - <i>Neo4j</i> - <i>Amazon Neptune</i>
SGBD de Tiempo Real	Se utilizan para gestionar y analizar datos en tiempo real, como datos de sensores, registros de aplicaciones y flujos de eventos	<ul style="list-style-type: none"> - <i>Apache Kafka</i> - <i>Apache Flink</i>
SGBD Distribuidos	Distribuyen datos a través de múltiples servidores o nodos para lograr alta disponibilidad y escalabilidad horizontal	<ul style="list-style-type: none"> - <i>Apache Cassandra</i> - <i>Amazon DynamoDB</i>
SGBD Empotrados	Son sistemas de gestión de bases de datos integrados en aplicaciones y dispositivos, lo que significa que no requieren una instalación separada	<ul style="list-style-type: none"> - <i>SQLite</i> - <i>H2 Database</i>
SGBD Multi-Modelo	Permiten el uso de múltiples modelos de datos (por ejemplo, documentos, grafos y clave-valor) en una sola base de datos	<ul style="list-style-type: none"> - <i>ArangoDB,</i> - <i>MarkLogic</i>
SGBD de Almacén de Datos	Especializados en análisis de datos a gran escala y almacenamiento de datos históricos para inteligencia empresarial y análisis	<ul style="list-style-type: none"> - <i>Google BigQuery</i> - <i>Snowflake</i>

Estos son algunos de los tipos más comunes de SGBD, y cada uno tiene sus propias ventajas y desventajas dependiendo de los requisitos específicos de un proyecto o aplicación. La elección del SGBD adecuado dependerá de factores como la escala de datos, la complejidad de las relaciones y los objetivos del sistema.

Fundamentos de Bases de Datos

1. Sistema de Información

2. ¿Es necesario siempre usar un SGBD?

3. Sistema gestor de Bases de Datos

Características deseables

1. CRUD
2. Recuperación de la información
3. Consistencia
4. Validez
5. Velocidad
6. Transacciones atómicas
7. Persistencia y soporte para Backups
8. Capacidad de extender sus características
9. Seguridad

Funciones

1. Crear y organizar la Base de Datos
2. Control de acceso
3. Evitar la redundancia e inconsistencia de los datos
4. Evitar anomalías en el acceso concurrente
5. Garantizar la correcta ejecución de las transacciones *
6. Respaldo y recuperación

Componentes

1. Administrador de Almacenamiento
2. Procesador de consultas
3. Gestor de transacciones

Tipos

Relacional
XML
Orientado a objetos
Objeto-relacional
NoSQL

Clasificación según accesos simultáneos soportados

Monousuario
Multiusuario

Clasificación según tamaño

SGBD ligero
SGBD Ofimática
SGBD de alto rendimiento

ACID
Atomicidad (Atomicity)
Consistencia (Consistency)
Aislamiento (Isolation)
Durabilidad (Durability)

4. Tipos de Bases de Datos

Bases de Datos centralizadas

Bases de Datos distribuidas

5. Arquitectura de una aplicación