

Principales características de un SGBD

Algunas de sus principales características son:

1. Permite una vista muy centralizada y clara de los datos para que sean accedidos de la mejor manera posible.
2. Se encargan de gestionar adecuadamente los datos, evitando a los usuarios o programas que les requieren, tener que entender dónde se encuentran físicamente los datos.
3. Estos sistemas disponen de un lenguaje de programación llamado SQL (Structured Query Language) para poder proteger y acceder a los datos.
4. La necesidad de requerir de un lenguaje para su acceso y su autonomía como sistema, proporcionan integridad y seguridad a los datos.
5. Suelen disponer de un sistema de bloqueo para el acceso simultáneo, lo que le da un plus de seguridad a la integridad de los datos.
6. Estos sistemas de base de datos disponen de API's (Application Programming Interface) muy visuales e intuitivas para poder gestionar los datos.
7. Un correcto SGBD proporcionará economías de escala en el procesamiento de grandes cantidades de datos ya que está hecho para ese tipo de operaciones.
8. Los SGBD proporcionan un nivel de abstracción entre la estructura lógica de la base de datos y el esquema físico que describe el contenido físico usado por la base de datos.
9. El programa de gestión de almacenamiento y su gestión de datos (servidor) es totalmente independiente del programa con el cuál se realizan las consultas (cliente).
10. Los SGBD realizan eficientes almacenamientos de los datos, pero estos se hacen de forma oculta para el usuario y nada tiene que ver con lo que finalmente se le presenta.
11. Son capaces de gestionar distintos tipos de bases de datos, por ejemplo: bases de datos relacionales (suelen ser las estándar) y bases de datos orientadas a objetos.
12. Multiplicidad de acceso a los datos.

El funcionamiento resumido y principal de un SGBD es el siguiente: un programa servidor de base de datos accede a esta, la lee y organiza, y posteriormente los programas del lado del cliente acceden a esta para gestionarla a su gusto. Por tanto, en un sistema de gestión de base de datos interactúan 3 actores: bases de datos, programa cliente y servidor.

Utilizar un SGBD aporta multitud de ventajas y beneficios para la gestión de bases de datos, debido a su agilidad, optimización, gran número de servicios, etc... y todo esto se refleja en un uso de la memoria y del CPU mucho mayor que un sistema de almacenamiento de archivos normal. Es quizás la única "desventaja", pero obvia debido al gran trabajo que realizan estos sistemas.

Los avances en la gestión de los datos han sido de gran ayuda para empresas, organizaciones y distintas instituciones ya que la gestión de los datos es un factor fundamental en muchos ámbitos.

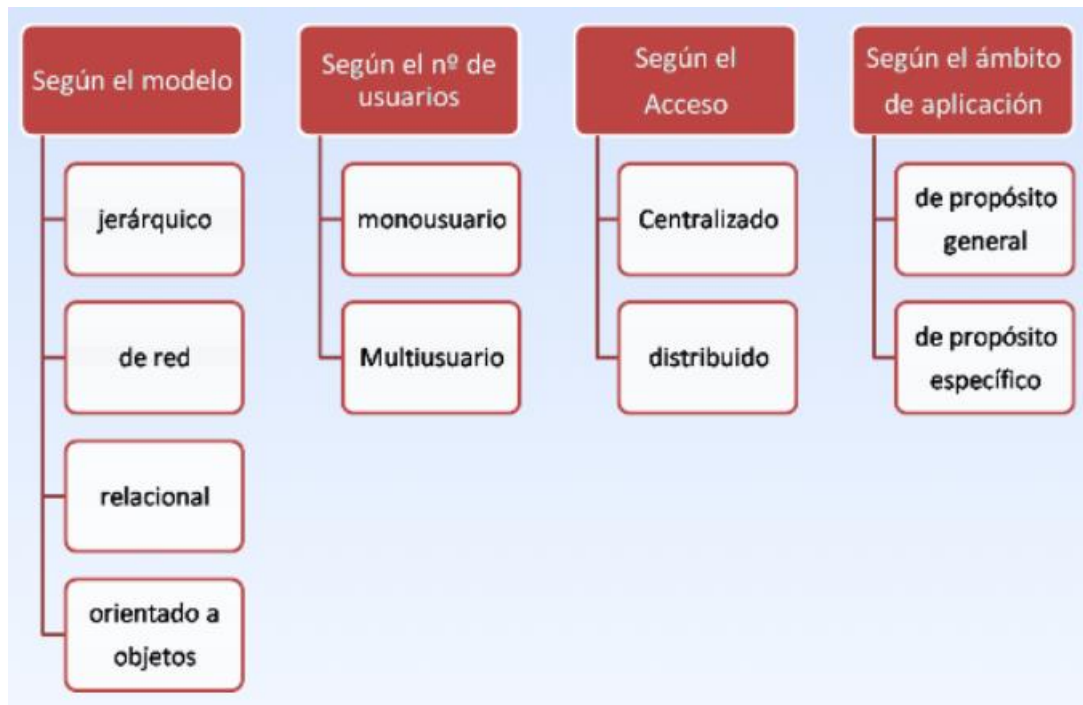
¿Cómo se clasifican los sistemas de gestión para las bases de datos?

Su adaptabilidad ayuda a gestionar todas las bases de datos, ya que contribuye a crear una buena relación y comunicación entre ellas sin importar el área a la que pertenezcan ni el tipo de datos que tengan que almacenar.

<

Este sistema de gestión de datos puede establecerse bajo distintos rubros operacionales, en función del tipo de información que necesite administrar, de la cantidad de herramientas que deba integrar o al número de usuarios que deben trabajar con estos datos.

Los tipos de sistemas de gestión de bases de datos se basan en la forma en que estas organizan los datos almacenados, esto es: el modelo de datos, la cantidad de usuarios, y su distribución.



Con base en modelos de datos

Sistema en Jerárquico: Se basa en un esquema de árbol para organizar los datos. Puede representarlos en tablas, las filas se componen por cada registro y las columnas por cada atributo. Se relacionan por medio de correspondencias.

Sistema en Red: Es una estructura con relaciones más complejas, pues cada registro puede rastrearse desde distintos caminos.

Sistema Relacional: Almacena datos en filas y columnas conformando tablas conectadas por claves comunes.

Orientado a Objetos: La base de datos orientada a objetos, o Object-Oriented Database, representa los datos en forma de objetos y clases. El objeto puede ser desde un resultado de búsqueda a una tabla; y una clase es una colección de objetos.

Los objetos tienen la capacidad de encapsular tanto un estado como un comportamiento de un objeto. Además, también puede almacenar las relaciones que tiene con otros objetos e, incluso, agruparse con otros objetos para formar objetos complejos. De esta forma, el objeto puede ser referenciado o nombrado posteriormente, como una unidad sin tener que entrar en sus complejidades. Este tipo de base de datos están generalmente escritas en lenguajes de programación orientados a objetos como Java, C o Smalltalk.

Base de datos NoSQL

El nombre de la base de datos NoSQL proviene de Not only SQL o, en español, no solo SQL. Esto se debe a que este tipo de base de datos suele evitar el uso del SQL o lo usa de apoyo, pero no como consulta. El hecho de evitar el SQL es porque se usa para proyectos en los que se necesita trabajar en la base de datos con un gran volumen. En las bases de datos con lenguaje SQL, los distintos atributos de un elemento, están en diferentes columnas, mientras que en una NoSQL todos los atributos se encuentran en una misma columna, ahorrando espacio. Algunos ejemplos de lenguajes usados por bases de datos NoSQL son: JSON (JavaScript Object Notation); CQL (Contextual Query Language, anteriormente conocido como Common Query Language); o GQL (Graph Query Language). Además, este tipo de bases de datos no suele permitir las uniones lógicas o joins. Esta restricción se debe al extremadamente alto volumen de datos que suelen manejar.

<< La base de datos NoSQL surgió con la aparición de las redes sociales (como Instagram, Facebook o Twitter) debido al incremento de datos que suponía>>

Base de datos gráfica o de grafos

En cuanto a la base de datos gráfica, también conocida como orientada a grafos, se diferencian de las anteriores en que están especializadas en establecer relaciones entre los datos de forma visual y navegar por dichas relaciones. Para leer la información, hay que leer los nodos o conectores (puntos de conexión de los datos de las tablas), generando un lenguaje natural. Algunos ejemplos de bases de datos de gráficas son Neo4j, PostGis, y Amazon Neptune.

Con base en la cantidad de usuarios

Según este criterio, depende de la cantidad de personal que necesite trabajar con las bases de datos. Es entonces que el sistema de gestión de base de datos puede ser para que solo un usuario acceda a la vez o uno que admita a varios al mismo tiempo.

Permite ser más específicos con el tipo de usuario que podrá visualizar y realizar movimientos en la base de datos. Dependerá de cuánta privacidad requiera la empresa.

Con base en su distribución

Sistema de gestión de datos centralizados: Tanto el sistema de gestión como las bases de datos se almacenan y operan desde un único sitio. Toda la información de cada área y cada herramienta de administración de datos se verán reflejados desde una sola plataforma.

Así todas tus tablas con tus datos ya organizados estarán en un solo lugar.

Sistema de gestión de datos distribuidos: La base de datos y el sistema de gestión están conectados por una red informática a pesar de estar distribuidos en distintos lugares. Por ejemplo, cada departamento de la empresa tiene una base de datos distinta y en el software de gestión continuarán repartidos de esta forma. Pero esto no impide el acceso para la gestión de cada una.

Consecuentemente, la base de datos distribuida tiene una mayor disponibilidad de los datos debido a sus múltiples ubicaciones. De esta forma, si una de las bases de datos fallase, se podría seguir funcionando correctamente, aunque puede que un poco ralentizado. Sin embargo, esta misma característica de estar distribuido puede provocar que haya duplicidad de los datos y un menor nivel de seguridad.

Dentro de las bases de datos distribuidas hay 2 tipos: homogéneas y heterogéneas. Las homogéneas son conscientes de las otras ubicaciones y cooperan en el procesamiento de las solicitudes. Además, tienen el mismo esquema y el mismo sistema de gestión de base de datos (DBMS). Mientras que las heterogéneas cooperan de forma limitada el procesamiento ya que pueden tener el DBMS y los esquemas diferentes e, incluso, que los sitios no se conozcan entre sí. En resumen, las homogéneas se utilizan generalmente dentro de una misma empresa u organización y, por eso, tienen el mismo DBMS y se conocen. Las heterogéneas se suelen utilizar entre distintas empresas u organizaciones en las que cada una tiene su propio DBMS y, es posible, que ni conozcan cuáles son las otras ubicaciones.

<<La base de datos distribuida más conocida en la actualidad es el Blockchain. La cual se hizo famosa gracias a las criptomonedas como Bitcoin y Ethereum>>

Base de datos ¿en la nube o en local?

Cada vez hay más proveedores que ofrecen la opción de tener la base de datos en la nube. Esto se debe al aumento de la demanda de este tipo. Sin embargo, todavía hay empresas que se deciden por la opción on-premise (en local).

La tendencia a ir a la nube se debe a varios motivos, entre ellos: la rapidez de implementación; las facilidades de acceso fuera del sistema; no tener que hacerse cargo del mantenimiento de los servidores. Sin embargo, algunas empresas que prefieren tener las bases de datos en local.

Aquellas empresas que tienen la base de datos en local, pueden trabajar con el software en local o de forma híbrida (base de datos en local, pero el software en la nube). Las razones por las que tener la base de datos en local pueden ser: porque les compensa porque tienen servidores propios donde pueden alojarlas sin que les suponga un coste extra; permite tener todo bajo su control; y, otras, porque piensan que teniendo los datos en su propia empresa

tendrán una mayor seguridad TIC. Este último punto es muy relativo. Si una empresa no tiene bien protegido su hardware, la seguridad podría verse comprometida. Además, la nube también tiene mejoras continuas de seguridad.

Principales motores de bases de datos

En la actualidad existen numerosos gestores de bases de datos, los cuales permiten almacenar y acceder a la información de manera ágil y flexible. A continuación, vemos cuáles son algunos de los más empleados.

MySQL

MySQL es el gestor de bases de datos más utilizado en el mundo. Se trata de un sistema multiusuario y multihilo que se emplea en la mayoría de páginas web actuales y en apps de software libre.

Entre las ventajas principales del gestor de bases de datos MySQL está su sencillez de uso y su buen rendimiento. Además, resulta muy sencillo de instalar y configurar y permite soporte multiplataforma. Por contra, presenta algún problemas de escalabilidad, es decir, no resulta tan eficaz cuando se trata de manejar grandes volúmenes de datos.

PostgreSQL

Sistema gestor de bases de datos Open Source más avanzado, multiplataforma y capaz de trabajar con proyectos grandes sin aumentar su complejidad. PostgreSQL es un sistema para gestionar bases de datos de muy alto nivel, completamente de software libre y con una licencia BSD (Berkeley Software Distribution), compatible con cualquier uso, ya sea personal o comercial. MySQL tiene una licencia dual, lo que significa que para proyectos comerciales habría que pagar por su uso. Sin embargo, PostgreSQL tiene una única licencia totalmente abierta para cualquier uso.

SQLite

Se trata de una biblioteca en C que permite realizar transacciones de datos. Su principal ventaja es que no necesita usar un servidor ni configuraciones, por lo que ocupa mucho menos espacio que otros gestores. Además, ofrece un buen rendimiento y cumple con los criterios de atomicidad, consistencia, aislamiento y durabilidad.

MongoDB

Es el gestor de base de datos NoSQL más empleado de la actualidad. Se trata de un sistema basado en ficheros que utiliza el lenguaje BSON. Su popularidad es tal que lo utilizan empresas tan conocidas como Google, Facebook o Cisco.

MariaDB

Es un gestor de bases de datos muy parecido a MySQL. De hecho, nace como una evolución de este programa, tras la compra de MySQL por Oracle. Tiene la mayoría de funcionalidades de MySQL pero incluye algunas extensiones extra. Además, es de código abierto y compatible al 100% con MySQL.

SQL Server

Los servidores SQL Server suelen presentar como principal característica una alta disponibilidad al permitir un gran tiempo de actividad y una conmutación más rápida. Todo esto sin sacrificar los recursos de memoria del sistema. Gracias a las funciones de memoria integradas directamente en los motores de base de datos SQL Server y de análisis, mejora la flexibilidad y se facilita el uso. Pero quizá su característica más destacada es que ofrece una solución robusta que se integra a la perfección con la familia de servidores Microsoft Server.

Características de Microsoft SQL Server :

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporte de procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red solo acceden a la información.
- Permite administrar información de otros servidores de datos

Versiones de SQL Server

La evolución de SQL Server no se ha detenido desde que en 1988 Microsoft desarrolló una base de datos variante de Sybase SQL para IBM OS/2. Esta fue la primera versión de Microsoft SQL Server que incluía varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas. Esto sirvió de entrada de Microsoft al mercado de las bases de datos en ese momento dominado por Oracle e IBM.

A partir de ahí van apareciendo diversas versiones, siendo estos los lanzamientos más importantes:

SQL Server 2000

SQL Server 2005

SQL Server 2008

SQL Server 2012

SQL Server 2014

SQL Server 2016

SQL Server 2017

SQL Server 2019

Para qué se utiliza SQL Server

Para responder a esta pregunta, quizá lo más destacado es que desde el principio Microsoft ha entendido que cada contexto es diferente, así que junto a cada lanzamiento Microsoft ofrece también diversas ediciones con diferentes conjuntos de características y dirigidos a diferentes usuarios. Estas ediciones son:

DataCenter

Enterprise

Standard

Web

Business Intelligence

Express

Cuál elegir depende en gran medida del uso al que queremos destinar nuestro servidor, siendo la más generalista la SQL Standard edition. Esta versión ofrece más funciones de administración de datos e inteligencia empresarial básicas para las cargas de trabajo que no son críticas con recursos de TI mínimos.

Por el contrario, SQL Server Business Intelligence es una solución de alto rendimiento más avanzada que permite a las organizaciones desarrollar e implementar soluciones de BI corporativas de autoservicio seguro, escalable y administrable.

Las versiones anteriores cuentan con diferentes licencias con un coste asociado que puede resultar prohibitivo para pequeñas empresas o desarrolladores. Por esa razón, si quieres adentrarte en el mundo de la administración de bases de datos, Microsoft también dispone de la versión SQL Server Express, una edición gratuita de SQL Server, ideal para estudiantes o desarrolladores que quieran desarrollar aplicaciones de escritorio, para la Web y para servidores pequeños.