

GENERALIDADES DE UN SISTEMA GESTOR DE BASE DE DATOS

1. BASES DE DATOS Y SGBD
2. ARQUITECTURA DE BASES DE DATOS
3. FUNCIONES DE UN SISTEMA GESTOR DE BASE DE DATOS
4. COMPONENTES DE UN SISTEMA GESTOR DE BASE DE DATOS
5. USUARIOS DE LOS SGBD
6. TIPOS DE SGBD
7. SISTEMAS GESTORES DE BASES DE DATOS COMERCIALES Y LIBRES.
8. SGBD RELACIONALES (SQL) FRENE A NO RELACIONALES (NOSQL)

1. BASES DE DATOS Y SGBD

Definición de Base de Datos:

Base de Datos o BD (Data Base, DB)

Es un conjunto de datos relacionados y organizados con cierta estructura.

Hay distintos modelos pero el más común es el relacional.

Para su manipulación necesitamos un SGBD.

Conjunto de datos interrelacionados entre sí, cuyo contenido pueden compartir varios usuarios con integridad máxima (medidas de seguridad para conservar correctos los datos) y redundancia mínima (evitar repeticiones en los datos)

- Datos relacionados

Ejemplo

<u>Alumnos</u>		<u>Notas</u>		<u>Asignaturas</u>
Nº Mat	→	Nº Mat		Cod Asign
Nombre		Cod Asign	←	Nombre
Apellidos		Nota		Profesor
Teléfono				

- Integridad de los datos

- No existan registros duplicados
- Datos dos tablas relacionadas:
 - No puede haber un registro de Notas sin su correspondiente registro relacionado de Alumnos
 - No puede borrarse un Alumno si existen notas para ese alumno
- Que se cumplan ciertas restricciones definidas con anterioridad (por ejemplo que la nota esté entre 0 y 10)
- Medidas de seguridad para conservar correctos los datos ante
 - Fallos del hardware del equipo (caída del sistema) Al recuperar el sistema la información deberá seguir siendo coherente.
 - Errores en el software (aplicaciones con errores por falta de pruebas por parte del programador)

TEMA 1: Generalidades de un S.G.B.D.

- Redundancia mínima

Redundancia significa repetición, a veces resulta inevitable pero en una BD bien diseñada se minimiza

Alumnos: NºMat Nombre Apell Dir

Notas: Nº Mat Cod Asig Nota

Mejor que Notas: Nombre Apell Cod Asig Nota

- Compartición de datos.

Las BD se construyen en general para ser manejadas por varios usuarios, a menudo al mismo tiempo. Es necesario:

- *Gestión de permisos: cada usuario tendrá acceso a determinados datos y operaciones*
- *Gestión de concurrencia: cuando varios usuarios intentan realizar operaciones que afecten a los mismos datos ha de existir un mecanismo que impida la pérdida o inconsistencia de los datos, por ejemplo, un registro está bloqueado por un usuario hasta que este acabe de actualizarlo.*

Definición de Sistema Gestor de Base de Datos (SGBD):

DBMS o Data Base Management System

Es una aplicación, conjunto de programas, que permite a los usuarios definir, crear y mantener bases de datos, proporcionando acceso controlado a las mismas.

Es una herramienta que sirve de interfaz entre el usuario y las bases de datos.

2. ARQUITECTURA DE BASES DE DATOS

Hay tres características importantes que tienen los sistemas de base de datos:

- Separación entre los programas de aplicación y los datos
- Manejo de múltiples vistas por parte de los usuarios
- Uso de un catálogo o diccionario para almacenar el esquema de la BD

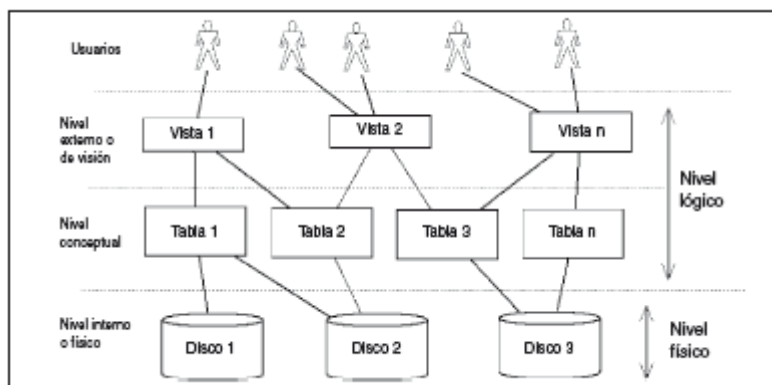
En 1975 el comité ANSI SPARC (American National Standard Institute- Standards Planning and Requirements Committee) propuso una arquitectura de tres niveles para los sistemas de bases de datos. Con ello se separaban los programas de aplicación de la base de datos física.



- **Nivel interno o físico.**
 - Describe la estructura física de la BD, el almacenamiento físico de los datos, mediante el esquema interno
 - Se describen los archivos que contienen la información, su organización, ubicación, la forma de acceso a sus registros, los tipos de registro, su longitud, los campos que los componen, los índices, etc.
- **Nivel conceptual**
 - Describe la estructura de toda la BD mediante el esquema conceptual o global
 - A partir de los requerimientos de los usuarios se obtiene una visión global.
 - Se describen, por ejemplo mediante el modelo Entidad-Relación, las entidades, atributos, relaciones, operaciones de los usuarios y restricciones, ocultando los detalles de las estructuras físicas de almacenamiento.
- **Nivel externo o de visión.**
 - Es el más cercano a los usuarios. Representa la visión individual de un usuario o de un grupo de usuarios.
 - Se describen varios esquemas externos o vistas de usuarios.
 - Cada esquema describe la parte de la BD que interesa a un grupo de usuarios *(Unos usuarios podrán ver unos datos, unas consultas, hacer determinadas actualizaciones...)*

Esta arquitectura de tres niveles garantiza la independencia de datos:

- Independencia física. Es la capacidad de modificar el almacenamiento interno, el nivel físico, sin que afecte al conceptual ni al externo. Por ejemplo reorganizar ciertos ficheros físicos para mejorar el rendimiento o añadir nuevos archivos de datos porque los que había se han llenado.
- Independencia lógica. Es la capacidad de poder modificar el esquema conceptual sin que afecte a los niveles internos ni externos. Por ejemplo: si se elimina una entidad los esquemas externos que no la tengan no se verán afectados.



3. FUNCIONES DE UN SISTEMA GESTOR DE BASE DE DATOS

La función principal de un SGBD es permitir que los usuarios realicen las cuatro operaciones fundamentales posibles: inserción o creación, consulta, actualización y borrado de datos y estructuras de datos de forma eficiente.

Para ello cuenta con un **diccionario de datos** o catálogo donde se almacenan las descripciones de los **datos de la base de datos (metadatos)**. Es un listado organizado de todos los datos que pertenecen al sistema.

Funciones o tareas de un SGBD:

- Mantenimiento (gestión) del diccionario de datos

- Garantizar la consistencia e integridad de los datos.

Garantizando que todas las actualizaciones de una determinada transacción se realicen o no se realice ninguna. (*por ejemplo en una inmobiliaria dar de baja un empleado y reasignar todos sus inmuebles*)

- Establecer y cumplir restricciones

Facilitando la introducción de restricciones de integridad (claves primarias, foráneas, reglas de integridad...)

- Control de la concurrencia

Ha de asegurar la posibilidad de un acceso concurrente por parte de varios usuarios garantizando la consistencia de los datos por ejemplo mediante el bloqueo de un registro mientras está siendo actualizado.

- Capacidad para integrarse en un sistema de comunicación para poder trabajar a través de red

- Control de la privacidad de la BD

A nivel de usuarios otorgando los permisos adecuados.

- Recuperación de la BD en caso de fallos. Mantenimiento de copias de seguridad.

- Proporcionar herramientas de administración. *Para administrar la base de datos de forma efectiva, normalmente vienen incluidas en el SGBD pero otras veces son creadas por terceros o por el propio administrador.*

4. COMPONENTES DE UN SISTEMA GESTOR DE BASE DE DATOS

Son los elementos que permiten realizar las funciones del punto anterior. Suelen incluir, entre otros, los siguientes:

➤ Lenguajes de los SGBD

Un SGBD ofrece distintos lenguajes para los distintos usuarios que van a trabajar con él:

Estos lenguajes se clasifican en:

- Lenguajes de definición de datos (LDD o DDL Data Description Language):
 - Se utiliza para especificar los distintos esquemas (distintos niveles de abstracción) de la BD:
 - Esquema interno: estructuras de almacenamiento
 - Esquema conceptual: diseñar la BD
 - Esquema externo: definir las vistas de usuarios
 - Lo utilizan los diseñadores y administradores de la BD
- Lenguajes de manipulación de datos (LMD o DML Data Manipulation Language)
 - Se utiliza para leer y actualizar los datos de la BD, es decir, para realizar consultas, inserciones, eliminaciones y modificaciones.
- Lenguajes de control de datos (LCD o DCL Data Control Language)
 - Se utilizan para controlar el acceso a la información de la BD definiendo usuarios, privilegios y tipos de acceso así como el control de la seguridad de los datos
 - De esta tarea se encarga el administrador.

SQL (Structured Query Language), lenguaje de consulta estructurado, es un lenguaje surgido de un proyecto de investigación de IBM para el **acceso a bases de datos relacionales**. Actualmente se ha convertido en un estándar de lenguaje de bases de datos, y la mayoría de los sistemas de bases de datos lo soportan, desde sistemas para ordenadores personales, hasta grandes ordenadores.

Tipos de Sentencias en SQL

El lenguaje SQL proporciona un gran repertorio de sentencias que se utilizan en variadas tareas, como consultar datos de la base de datos, crear, actualizar y eliminar objetos, controlar el acceso a la base de datos y a los objetos. Dependiendo de las tareas, podemos clasificar las sentencias SQL en varios tipos:

- **DML : (Data Manipulation Language)**

Manipulación de datos, por ejemplo

SELECT	Recupera filas de la base de datos
INSERT	Añade nuevas filas en la base de datos

TEMA 1: Generalidades de un S.G.B.D.

DELETE	Suprime filas en la base de datos
UPDATE	Modifica datos de las filas de la base de datos

- **DDL : (Data Definition Language)**

Definición de datos

CREATE	Table, view, index, synonym...	Crear objetos
DROP		Borrar objetos
ALTER		Modificar la definición de un objeto

- **DCL : (Data Control Language)**

Control de accesos, restricciones

GRANT	Concede privilegios de acceso a usuarios
REVOKE	Suprime privilegios de acceso a usuarios

➤ Diccionario de datos

El diccionario de datos describe la base de datos y sus objetos.

El diccionario incluye dos tipos de objetos:

- Tablas: tablas del diccionario de datos
- Vistas: accesibles por los usuarios autorizados

Toda la información sobre la definición y gestión de los datos de una BD está recogida en el diccionario de datos o repositorio, en el que se describen, entre otras cosas:

- Estructura lógica y física de la BD: estructuras de los ficheros, espacio asignado y utilizado por los objetos, relaciones...
- Definición de todos los objetos: tablas, vistas, índices, disparadores, procedimientos, funciones, etc
- Valor por defecto de las columnas
- Información sobre las restricciones de integridad (constraints)
- Nombre de usuarios
- Los privilegios y roles otorgados a los usuarios
- Auditoría. Seguimiento que hace un sistema gestor sobre las consultas.
- Estadísticas de utilización, tales como la frecuencia de las transacciones y el número de accesos realizados a los objetos de la base de datos.

TEMA 1: Generalidades de un S.G.B.D.**➤ Objetos**

- Tablas
- Columnas
- Funciones, procedimientos
- Dominios
- Restricciones de tabla
- Disparadores
- Sinónimos
- Secuencias
- Espacios de tablas, zona de disco donde se va a guardar la información
- Usuarios
- Índices
- Cluster, tablas físicamente juntas
- Paquetes
- Privilegios
- Roles, conjunto de privilegios

➤ Diversas herramientas para

- Mantener la seguridad de acceso
- Mantener la integridad de los datos
- Control de concurrencia
- Control de recuperación
- Gestión del diccionario
- Programación de aplicaciones
- Importación / exportación de datos
- Replicación
- Sincronización
- Optimizar las consultas
- Gestionar las transacciones
- Planificar operaciones y procesos
- Mantenimiento de copias de seguridad

5. USUARIOS DE LOS SGBD

➤ Administradores: DBA (DataBase Administrator)

- Diseño físico e implementación
- Control de seguridad
- Control de concurrencia
- Mantiene el sistema
- Conocen en profundidad el SGBD

Trabajan en el nivel de abstracción físico relacionado con el almacenamiento.

Distinguimos los administradores del propio sistema gestor encargados de la instalación y configuración del sistema, del control de acceso a los recursos, de la seguridad y de la monitorización y optimización del sistema gestor.

Por su parte, los administradores de bases de datos se encargan del diseño físico de la misma, implementación y mantenimiento de la base de datos.

➤ Diseñadores de la base de datos

- Diseñan de forma lógica la base de datos, identifican los datos, las relaciones y restricciones
- Son independientes del SGBD que se utilice, de los programas de aplicación, de los lenguajes.
- Tiene que implicar a todos los usuarios de la base de datos
- Debe tener conocimiento exhaustivo de la empresa para la que se diseña la base de datos.

Realizan el diseño lógico de la base de datos, debiendo identificar los datos, las relaciones entre datos y las restricciones sobre los datos y sus relaciones. El diseñador de la base de datos debe tener un profundo conocimiento de los datos de la empresa y también debe conocer sus reglas de negocio. Las reglas de negocio describen las características principales de los datos tal y como los ve la empresa. Para obtener un buen resultado, el diseñador de la base de datos debe implicar en el desarrollo del modelo de datos a todos los usuarios de la base de datos, tan pronto como sea posible. El diseño lógico de la base de datos es independiente del SGBD concreto que se vaya a utilizar, es independiente de los programas de aplicación, de los lenguajes de programación y de cualquier otra consideración física.

➤ Programadores de aplicaciones

- Realizan los programas de aplicación para consultar, insertar, eliminar o actualizar datos.

Estas aplicaciones servirán a los usuarios finales para, de una forma amigable, poder consultar datos, insertarlos, actualizarlos y eliminarlos.

TEMA 1: Generalidades de un S.G.B.D.

➤ Usuarios finales

- No tienen que conocer el funcionamiento de la base de datos ni la organización.
- No tienen que tener conocimientos informáticos.

Trabajan en el nivel externo mediante vistas o porciones de las bases de datos. Son clientes de las bases de datos que hacen uso de ellas sin conocer en absoluto su funcionamiento y organización interna. Son personas con pocos o nulos conocimientos de informática.

6. TIPOS DE SGBD

Tipos de SGBD

Existen numerosos SGBD en el mercado que podemos clasificar según los siguientes criterios:

↪ **Modelo lógico en el que se basan**

- Jerárquico.
- En red.
- Relacional.
- Objeto-relacional.
- Orientado a objetos.
- NoSQL

↪ **Número de sitios**

- Centralizados: en un solo servidor o equipo.
- Distribuidos: en varios equipos que pueden ser homogéneos y heterogéneos.

En un sistema de bases de datos centralizado todos los componentes (software, datos y soportes físicos) residen en un único lugar físico. Los clientes (aplicaciones, funciones, programas cliente, usuarios) acceden al sistema a través de distintas interfaces que se conectan al servidor. Sin embargo, existe otra arquitectura cada vez más extendida en la que se opta por un esquema distribuido en el que los componentes se distribuyen en distintos computadores comunicados a través de una red de cómputo. Dichos sistemas se conocen con el nombre de Sistemas Gestores de Bases de Datos Distribuidas o DDMGS.

↪ **Ámbito de aplicación**

- Propósito General: orientados a toda clase de aplicaciones.
- Propósito Específico: centradas en un tipo específico de aplicaciones.

↪ **Tipos de datos**

- Sistemas relacionales estándar: manejan tipos básicos (int, char, etc.).
- XML: para el caso de bases de datos que trabajan con documentos xml.
- Objeto-relacionales: para bases relacionales que incorporan tipos complejos de datos.
- De objetos: para bases de datos que soportan tipos de objeto con datos y métodos asociados.

↪ **Software libre o de pago**

- SGBD comerciales: Oracle, SQL Server...

TEMA 1: Generalidades de un S.G.B.D.

- SGBD código libre con licenciamiento dual: MySQL (actualmente de Oracle, aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso)
- SGBD libre: PostgreSQL SGBD orientado a objetos y libre

7. SISTEMAS GESTORES DE BASES DE DATOS COMERCIALES Y LIBRES.

Con el crecimiento de Internet, el **software libre** se ha consolidado como alternativa, técnicamente viable y económicamente sostenible frente al **software comercial**, ofreciendo en muchos casos los mismos servicios a un coste cada vez más reducido.

Software libre y gratuito no significan lo mismo.

El software Libre es un asunto de libertad, no de precio:

- Libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades teniendo acceso al código fuente.
- Libertad de distribuir copias, con lo que puedes ayudar a otras personas o entidades interesadas.
- Libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que todos se beneficien.

EJEMPLOS SGBD nos encontramos con:

SGBD para	LIBRE	COMERCIAL
Alternativas ofimáticas BD Ofimáticas	Openoffice (BASE)	Microsoft Office (ACCESS).
Más avanzados y de propósito general, para cierta cantidad de usuarios diversos con control de integridad y otras funcionalidades	MySQL	SQL Server
Nivel superior en cuanto a potencialidad, para grandes corporaciones que requieran requieran herramientas avanzadas para grandes bases de datos. BD Corporativas	PostgreSQL	Oracle

A menudo los SGBD comerciales ofrecen versiones gratuitas (que no libres) de sus productos que permiten probarlos y aprender sus interioridades. Estas versiones se suelen denominar de **tipo express** y están limitadas en la funcionalidad,

Factores importantes a tener en cuenta a la hora de decidir serán:

- Si se dispone de personal cualificado: usar software libre a menudo implica un mayor conocimiento del producto y, por tanto, personal más cualificado. Esto permite un mayor control sobre el mismo y sobre las aplicaciones a desarrollar, además de una gran independencia con respecto al proveedor del mismo, y ahorro en cuanto a licencias y costes de mantenimiento y soporte. Los sistemas comerciales, por el contrario, suelen ser más cerrados y rígidos pero cuentan con servicios de soporte de gran calidad.
- Cuestiones de seguridad y control de acceso a los recursos.
- Complejidad en la migración de datos.
- Volúmenes de información que soportará y número de accesos esperable

No se trata tanto de defender “qué es mejor en sí mismo”, sino de tener la información y poder elegir “qué es mejor según mis circunstancias”.

Entre los SGBD comerciales y libres hay un tercer grupo:

- **SGBD código libre con licenciamiento dual:** MySQL (actualmente de Oracle, aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso)

8. SGBD RELACIONALES (SQL) FRENTE A NO RELACIONALES (NOSQL)

Base de datos relacional

Véase también: Base de datos objeto-relacional

La **base de datos relacional** (BDR) es un tipo de **base de datos** (BD) que cumple con el **modelo relacional** (el modelo más utilizado actualmente para implementar las BD ya planificadas). Tras ser postuladas sus bases en 1970 por Edgar Frank Codd,¹ de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos.²

Un sistema de software utilizado para mantener las bases de datos relacionales es un *relational database management system* (RDBMS) o sistema de gestión de bases de datos relacionales. Virtualmente, todos los sistemas de bases de datos relacionales utilizan **SQL** (Structured Query Language) para consultar y mantener la base de datos.

NoSQL

En informática, **NoSQL** (a veces llamado “**no solo SQL**”) es una amplia clase de **sistemas de gestión de bases de datos** que difieren del modelo clásico de **SGBDR** (Sistema de Gestión de Bases de Datos Relacionales) en aspectos importantes, siendo el más destacado que no usan **SQL** como **lenguaje principal de consultas**. Los datos almacenados no requieren estructuras fijas como **tablas**, normalmente no soportan operaciones JOIN, ni garantizan completamente ACID (atomicidad, consistencia, aislamiento y durabilidad) y habitualmente escalan bien horizontalmente. Los sistemas NoSQL se denominan a veces “no solo SQL” para subrayar el hecho de que también pueden soportar lenguajes de consulta de tipo SQL.



Por lo general, los investigadores académicos se refieren a este tipo de **bases de datos** como **almacenamiento estructurado**, término que abarca también las bases de datos relacionales clásicas. A menudo, las bases de datos NoSQL se clasifican según su forma de almacenar los datos, y comprenden categorías como **clave-valor**, las implementaciones de **BigTable**, bases de datos documentales, y **bases de datos orientadas a grafos**.

Los sistemas de bases de datos NoSQL crecieron con las principales redes sociales, como Google, Amazon, Twitter y Facebook. Estas tenían que enfrentarse a desafíos con el tratamiento de datos que las tradicionales SGBDR no solucionaban (ver artículo³). Con el crecimiento de la web en tiempo real existía una necesidad de proporcionar información procesada a partir de grandes volúmenes de datos que tenían unas estructuras horizontales más o menos similares. Estas compañías se dieron cuenta de que el rendimiento y sus propiedades de tiempo real eran más importantes que la coherencia, en la que las bases de datos relacionales tradicionales dedicaban una gran cantidad de tiempo de proceso^[cita requerida].

En ese sentido, a menudo, las bases de datos NoSQL están altamente optimizadas para las operaciones recuperar y agregar, y normalmente no ofrecen mucho más que la funcionalidad de almacenar los registros (p.ej. **almacenamiento clave-valor**). La pérdida de flexibilidad en tiempo de ejecución, comparado con los sistemas SQL clásicos, se ve compensada por ganancias significativas en escalabilidad y rendimiento cuando se trata con ciertos modelos de datos^[cita requerida].