

## **INTRODUCCIÓN A LAS BASES DE DATOS (BD)**

### **1. Base de datos (BD)**

Una **Base de Datos (BD)** es un conjunto de datos relacionados entre sí, organizados y estructurados con información referente a algo.

Un **Sistema Gestor de Bases de Datos (SGBD)** es un conjunto de programas que permite a los usuarios crear y mantener una base de datos (BD). Además de almacenar la información, el SGBD permite realizar consultas sobre esos datos, obtener listados, generar pequeños programas de mantenimiento de la BD (como formularios de entrada de datos) o ser utilizado como servidor de datos para programas más complejos realizados en cualquier lenguaje de programación. Además ofrece otras herramientas más propias de la gestión de bases de datos como sistemas de permisos para la autorización de accesos, volcados de seguridad, transferencia de ficheros, recuperación de la información dañada, indexación, etc.

Antes de la existencia de los SGBD, las empresas gestionaban la información a través de **sistemas de procesamiento de archivos**. Los registros permanentes eran almacenados en varios archivos y se escribían diferentes programas de aplicación para extraer y añadir registros a los archivos adecuadas. Sin embargo, mantener la información mediante estos sistemas tiene una serie de **inconvenientes** importantes:

- **Redundancia e inconsistencia de los datos:** la redundancia consiste en la repetición innecesaria de los datos y que muchas veces puede generar la inconsistencia de los datos, es decir, que las diversas copias de los mismos datos no coincidan.
- **Dificultad de acceso a los datos:** no permite que los datos sean obtenidos de una forma práctica y eficiente.
- **Aislamiento de datos:** ocurre ante la dificultad de ampliar los requerimientos de las aplicaciones informáticas ya creadas para acceder a los datos.
- **Problemas de integridad:** los valores de los datos almacenados en la base de datos deben satisfacer ciertos tipos de restricciones de consistencia, que son difíciles de mantener en estos sistemas de procesamiento de archivos.
- **Problemas de atomicidad:** cuando se produce un fallo en una aplicación y se crean datos inconsistentes el sistema debe asegurar que los datos se restauren al estado de consistencia que había antes del fallo. Para ello, es necesario que las operaciones sean atómicas, o que se realicen enteras o que no se realicen, esto es difícil de asegurar en estos sistemas.
- **Anomalías en el acceso concurrente:** cuando varios usuarios o programas acceden a un sistema para actualizar datos se debe garantizar que los procesos de actualización se efectúen de forma que no se creen datos inconsistentes.
- **Problemas de seguridad:** no todos los usuarios deberían poder acceder a todos los datos. Estas restricciones de seguridad son difíciles de asegurar en los sistemas de archivos.

Estos inconvenientes, entre otros, motivaron el desarrollo de los sistemas de bases de datos. Por tanto, **los objetivos de un SGBD** son:

- Reducir o eliminar: redundancia, inconsistencia, aislamiento y errores de acceso concurrente.
- Aumentar la seguridad y la integridad de los datos.
- Conseguir una administración centralizada de los datos.

## 2. Arquitectura de un SGBD

La mayoría de los SGBD actuales están inspirados en una arquitectura llamada ANSI/X3/SPARC que divide la base de datos en tres niveles:

1. **Nivel Externo:** es la representación de los datos, tal y como los ve el usuario. Cada usuario tendrá una visión distinta de los datos según sus permisos y las herramientas que use.
2. **Nivel conceptual:** es la representación abstracta del contenido de la base de datos. Contiene la definición de los datos existentes, restricciones de seguridad, controles de integridad, etc.
3. **Nivel interno:** es una representación a bajo nivel de la BD, en la que se definen la forma en que los datos se almacenan físicamente en la máquina, espacio reservado, estrategias de acceso etc.

Gracias a la arquitectura a tres niveles se introduce el concepto de independencia de datos. Se definen dos tipos de independencia:

- **Independencia lógica:** capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación.
- **Independencia física:** capacidad de modificar el esquema interno sin tener que modificar el esquema conceptual ni el externo.

### 3. Componentes de un SGBD

Los componentes principales de un SGBD son los siguientes:

- **Herramientas de gestión.** Todos los SGBD disponen de herramientas de gestión para poder crear las bases de datos, manipularlas, modificar su diseño, crear usuarios, asignar permisos, etc. Estas herramientas suelen estar solo disponibles para los administradores de la base de datos, es decir, aquellos usuarios especializados en mantener la base de datos en perfecto estado.
- **Herramientas de programación.** Muchos SGBD ofrecen la posibilidad de crear las aplicaciones que utilizarán los usuarios para acceder a la base de datos cuando estos usuarios no pueden (o no deben) trabajar directamente con el SGBD. Un ejemplo, es Access.
- **Lenguajes.** Los SGBD proporcionan unos lenguajes de trabajo para poder acceder a los datos, y también para poder efectuar algunas tareas de gestión. Estos lenguajes pueden clasificarse en:
  - **Lenguajes de definición (LDD):** son aquellos que se usan para crear la base de datos o modificar su estructura una vez creada, para crear las vistas de usuarios, las estructuras de almacenamiento etc.
  - **Lenguajes de manipulación (LMD):** son los que permiten trabajar con los datos, haciendo consultas, insertando datos, eliminando datos, realizando consultas, etc.
- **Diccionario de datos.** es el lugar donde se deposita la información acerca de todos los elementos que forman la BD. Estos datos se consultan antes de leer o modificar los datos de un sistema de base de datos. En el caso de una BD relacional, el diccionario de datos almacena información sobre la estructura lógica y física de la BD, definiciones de todos los objetos de la BD (tablas, vistas, procedimientos etc.), espacio asignado y utilizado por cada objeto, permisos de usuarios, restricciones de integridad, etc.
- **Usuarios.** En los SGBD existen diferentes categorías de usuarios:
  - **Administradores de la BD:** son los usuarios que poseen el control central sobre el sistema. Entre sus funciones se encuentran: el diseño y modificación de la base de datos, la definición de la estructura de almacenamiento y el método de acceso a los datos, la autorización para el acceso a los datos y la especificación de las restricciones de integridad.
  - **Programadores de aplicaciones:** se encargan de diseñar y programar las aplicaciones necesarias para la utilización de la BD, realizando llamadas al SGBD.
  - **Usuarios finales:** donde tenemos desde usuarios que interactúan con los datos a través de los programas de aplicación, a usuarios que interactúan directamente con la BD mediante lenguajes de consulta de bases de datos.

## 4. Arquitectura Cliente/Servidor

La arquitectura cliente/servidor separa las funciones de una aplicación en componentes que establecen diálogos entre sí, para intercambiar información, servicios o recursos con el objeto de realizar una tarea común. Cada componente puede estar en un ordenador diferente. El proceso que inicia el diálogo o solicita recursos se denomina cliente y suele ser la aplicación que el usuario está ejecutando. El proceso que responde a las solicitudes se le denomina servidor.

Esta arquitectura, por tanto, se basa en varias plataformas interconectadas, una de las cuáles actúa como servidor de la base de datos, en las que los datos están físicamente localizados y centraliza las funciones de administración. Las plataformas denominadas “clientes” realizan funciones de manejo de las interfaces de usuario, lógica de aplicación, etc.

La arquitectura cliente/servidor permite las siguientes configuraciones:

- **Basada en anfitrión:** cuando la máquina cliente y servidor es la misma.
- **Cliente/Servidor:** la base de datos reside en un servidor y los usuarios acceden a la base de datos del servidor a través de la red.
- **Procesamiento distribuido:** la base de datos está repartida en más de una máquina servidora.

## 5. Modelo de datos

Un **modelo de datos** es una serie de conceptos que puede utilizarse para describir un conjunto de datos y las operaciones para manipularlos. Hay dos tipos de modelos de datos: los modelos conceptuales y los modelos lógicos.

Los **modelos conceptuales** se utilizan para representar la realidad a un alto nivel de abstracción. Mediante los modelos conceptuales se puede construir una descripción de la realidad fácil de entender.

En los **modelos lógicos**, las descripciones de los datos tienen una correspondencia sencilla con la estructura física de la base de datos.

En el diseño de bases de datos se usan primero los modelos conceptuales para lograr una descripción de alto nivel de la realidad, y luego se transforma el esquema conceptual en un esquema lógico. El motivo de realizar estas dos etapas es la dificultad de abstraer la estructura de una base de datos que presente cierta complejidad. Un esquema es un conjunto de representaciones lingüísticas o gráficas que describen la estructura de los datos de interés.

Los modelos conceptuales deben ser buenas herramientas para representar la realidad, por lo que deben poseer las siguientes cualidades:

- **Expresividad:** deben tener suficientes conceptos para expresar perfectamente la realidad.
- **Simplicidad:** deben ser simples para que los esquemas sean fáciles de entender.
- **Minimalidad:** cada concepto debe tener un significado distinto.
- **Formalidad:** todos los conceptos deben tener una interpretación única, precisa y bien definida.

Nosotros vamos a estudiar dos modelos concretos, el modelo entidad-relación extendido (EER), que es un modelo conceptual, y el modelo relacional como modelo lógico.