

INTRODUCCIÓN A LAS BASES DE DATOS

1.1 Introducción a los sistemas de información

Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio.

Cualquier organización, incluye una infraestructura para coordinar los flujos de información necesarios para desarrollar sus actividades de acuerdo con una estrategia. El conjunto de los procesos, los flujos, los datos y la estrategia forman el Sistema de Información.

Los **componentes** de un sistema de Información, son los siguientes:

- **Los procedimientos y las prácticas habituales de trabajo**, de forma que empresas que se dedican a una misma actividad, tienen distinta forma de llevar a cabo las tareas que desarrollan.
- **La información**, que se debe adaptar a las personas que la manejan y a los procedimientos de trabajo utilizados.
- **Las personas o usuarios**, que utilizan la información para realizar sus actividades de acuerdo con los procedimientos establecidos.
- **El equipo o soporte para la comunicación, el procesamiento y el almacenamiento de la información**. Es la parte más visible del Sistema de Información y puede incluir elementos de distintos niveles tecnológicos: papel, lápiz, máquina de escribir, ordenadores...

Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información.

Cuando parte o toda la gestión de un sistema de información se realiza con ordenadores se habla de *Sistema de Información Basado en Ordenadores* o **Sistema Informático**.

Los sistemas informáticos tradicionales han sido llamados por algunos autores **sistemas orientados al proceso**, debido a que en ellos se pone el énfasis en los **tratamientos que reciben los datos**, los cuales se almacenan en ficheros diseñados para una determinada aplicación. Las aplicaciones se analizan e implantan con entera independencia unas de otras, y los datos no se suelen transferir entre ellas, sino que se duplican siempre que los correspondientes trabajos los necesitan.

1.2 Conceptos básicos sobre ficheros

Vamos a ver algunos conceptos relacionados con los ficheros o archivos tradicionales.

Los ficheros son estructuras de información que crean los sistemas operativos de los ordenadores para poder almacenar datos. Suelen tener un nombre y una extensión que determina el formato de la información que contiene.

El formato y tipo de archivo determina la forma de interpretar la información que contiene, pues lo que se almacena realmente es un conjunto de ceros y unos.

Tradicionalmente, los ficheros se han clasificado de muchas formas distintas, según su **contenido** (texto o binario), según su **organización** (secuencial, indexada, directa) o según su **utilidad** (maestros, históricos y de movimientos)

La **utilidad** de un fichero indica qué uso se va a hacer de él, por ejemplo, puede contener datos fundamentales para una organización, como los datos de los clientes, que se almacenan en un **fichero principal llamado maestro**. Si hay variaciones (altas, modificaciones o bajas de clientes) en los ficheros maestros, se almacenan en los llamados **ficheros de movimientos** que posteriormente se enfrentan a los maestros para incorporar las modificaciones. Finalmente, **cuando** existen datos que **ya no son necesarios** para el proceso diario pasan a formar parte de los **ficheros históricos**.

EJERCICIO 1:

Haz una lista de ficheros que se te ocurran para la gestión de toda la información referente al instituto. A continuación, pon un ejemplo de fichero que sea maestro y otro histórico.

La **organización** de un fichero **dicta la forma en que se han de acceder a los datos**. Podemos realizar la siguiente clasificación:

1. Ficheros secuenciales:

- Los registros son almacenados de forma contigua.
- Las operaciones de actualización (inserción, modificación, eliminación) requieren la reescritura de todo el fichero.
- Sólo es posible añadir al final
- La consulta de un registro requiere la lectura de los anteriores.

2. Ficheros indexados:

En este modo de organización, **al fichero le acompaña un fichero de índice que tiene la función de permitir el acceso directo a los registros del fichero de datos**. Estos índices son análogos a los de los libros, si nos interesa leer un capítulo dado se puede recurrir al índice que indica la página de inicio, y se accede al libro por esa página, sin tener que mirar en todas las páginas anteriores para localizarlo.

Las **características** más importantes de un fichero indexado, son las siguientes:

- **El diseño del registro tiene que tener un campo, o combinación de campos, que permita identificar cada registro de forma única**, es decir, que no pueda haber dos registros que tengan la misma información en él. Este campo es el denominado **campo clave** y es el que **va a servir de índice**. Un mismo fichero puede tener más de un campo clave, pero al menos uno de ellos **no permitirá valores duplicados** y se le llama **clave primaria**. **A las restantes se les llama claves alternativas**. Permiten utilizar el modo de acceso secuencial y el modo de acceso directo para leer la información guardada en sus registros.
- **El modo de acceso directo se hace conociendo el contenido del campo clave del registro que queremos localizar**. Con esa información el sistema operativo puede consultar el índice y conocer la posición del registro dentro del fichero.
- **Solamente se puede grabar en un soporte de acceso directo**. Si esto no fuera así no podría emplear el acceso directo.

3. Ficheros de acceso directo:

- La dirección de los registros en el soporte viene dada por un cálculo sobre la clave primaria de los registros. Para ello se utiliza un algoritmo o función.

Existen también variantes de los anteriores que mezclan las mejores características de cada una de ellos.

Hoy en día, con la aparición de las bases de datos, ya no se usan estas clasificaciones. Actualmente un sistema operativo trata un fichero desde dos puntos de vista:

- Según su contenido (texto o datos binarios)
- Según su tipo (imágenes, ejecutables, vídeos...)

Ficheros de texto

Suelen llamarse también ficheros planos o ficheros ascii. El ASCII es un estándar que asigna un valor numérico a cada carácter. La asignación viene dada por la tabla de códigos ASCII. Dicha tabla utiliza un byte por carácter.

Aunque los ficheros de texto no necesitan un formato para ser interpretados, suelen tener extensiones para conocer qué tipo de texto hay dentro del fichero. Por ejemplo los de configuración (.ini, .conf), los de código fuente (.sql, .java), los de páginas web (.xml, .css)

Ficheros binarios

Son todos los que no son de texto y requieren un formato para ser interpretado. A continuación, se muestran algunos tipos de formato de ficheros binarios:

- De imagen: .jpg, .png...
- De vídeo: .avi
- Comprimidos: .zip, .rar
- Ejecutables: .exe, .com
- Procesadores de texto: .doc, .odt

Generalmente los ficheros que componen una base de datos son de tipo binario, puesto que la información que hay almacenada en ellos debe tener una estructura lógica y organizada para que las aplicaciones puedan acceder a ella de manera universal, esto es siguiendo un estándar. Esta organización es muy difícil de expresar mediante ficheros de texto, por tanto, la información de una base de datos se suele guardar en uno o varios ficheros:

- En el caso de MySQL, el gestor guarda su información en tres ficheros de datos binarios con extensión myd, myi y frm

En lo que se refiere al **procesamiento de los datos de los ficheros**, existen dos métodos:

- Modo Secuencial:** Se lee la información de un fichero de registro en registro teniendo que leer todos los que hay antes del que buscamos. Se emplea bien por deseo, bien por imposición del tipo de soporte que estamos usando. El acceso secuencial es recomendable cuando se quiere trabajar con muchos registros del fichero.
- Modo Directo:** Se puede acceder a un registro si tener que leer todos los anteriores (basta con un pequeño número de lecturas). Hay dos maneras:
 - **Cálculo:** Cada región tiene una clave sobre la que se aplica un cálculo que indica el lugar de grabación (**Hashing**).
 - **Índices:** existe un índice independiente o asociado al fichero en el cual se busca el registro y nos indica donde está.

Las **operaciones** que se pueden hacer con los ficheros son las siguientes:

- **Creación:** Consiste en la grabación, por primera vez, sobre un soporte de los registros de un fichero.
- **Borrado:** Consiste en la eliminación de todo el fichero.
- **Ordenación o clasificación:** Consiste en cambiar el orden en que están grabados los registros del fichero en el soporte.
- **Copia:** Crear un nuevo fichero idéntico a uno ya existente. Es muy recomendable como medida de seguridad.
- **Modificación del diseño o estructura**
- **Modificación de registros**
- **Inserción/Eliminación de registros**
- **Recuperación de la información**
- **Mantenimiento, reparación y compactación.**

La compactación elimina en el fichero los huecos producidos por el borrado de determinados registros.

La reparación comprueba la consistencia de cada uno de los registros y, si se encuentran errores, se procede a su reparación, si es posible.

Las aplicaciones en los **años 70** solían trabajar con ficheros que estaban en cintas magnéticas, por lo que se usaba el **acceso secuencial**.

Cada vez que se le quería añadir una aplicación o programa que requería el uso de algunos de los datos que ya existían y de otros nuevos, se diseñaba un fichero

nuevo con todos los datos necesarios (algo que provocaba redundancia) para evitar que los programas tuviesen que leer muchos ficheros.

Este planteamiento produce, además de una **ocupación inútil de memoria** y un **aumento de los tiempos de proceso**, al repetirse los mismos controles y operaciones en los distintos ficheros. Pero más graves todavía son las **inconsistencias** que a menudo se presentan en estos sistemas, **debido a que la actualización de los mismos datos**, cuando se encuentran en más de un fichero, no se suele realizar de forma simultánea en todos los ficheros.

De este análisis se deduce claramente la necesidad de una gestión más racional del conjunto de datos, surgiendo así un nuevo enfoque que se apoya sobre una base de datos en la cual los datos son recogidos y almacenados una sola vez, con independencia de los tratamientos.

Vemos, por tanto, que la solución de los problemas asociados al tratamiento de los datos en los sistemas tradicionales lleva a un cambio radical en el enfoque del sistema de información, en el cual los datos se organizan y mantienen en un conjunto estructurado que no está diseñado para una aplicación concreta, sino que, por el contrario, tiende a satisfacer las necesidades de información de la organización; necesidades cuya diversidad se ve acentuada con el transcurso del tiempo.

Estos sistemas orientados hacia los datos, van sustituyendo a los sistemas orientados hacia el proceso que, por su poca fiabilidad, falta de adecuación a la realidad y mal asegurada confidencialidad han ido perdiendo de forma progresiva la confianza de los usuarios.

Ejemplo de archivos tradicionales

Se cuenta con dos archivos: CLIENTES y FACTURAS.

El primer archivo tiene los datos básicos de los clientes, mientras que en el segundo se almacenan las ventas realizadas. Al emitir cada factura se ingresan nuevamente los datos num, nombre, domicilio.

CLIENTES

Num	Nombre	Dirección	Teléfono	Fecha Nacimiento	e-mail
1225	Juan García	Guaná 1202	985674863	13/08/1972	jgarcia@adinet.com
1226	Fernando Martínez	Rincón 876	984568643	23/02/1987	fmar@gmail.com
...

FACTURAS

Num	Nombre	Dirección	Producto	Precio
1225	Joaquín García	Guaná 1202	Azulejos	1250
1226	Fernando Martínez	Rincón 876	Pintura	900
...

Los **sistemas de ficheros tradicionales** traían, por tanto, consigo una serie de inconvenientes:

- **Redundancia e inconsistencia de datos:** Puesto que los datos pueden estar duplicados en distintos sitios y tener distintos formatos. Ej: Teléfono de un alumno en varios sitios distintos. Esto aumenta los costes de almacenamiento y acceso y produce inconsistencias puesto que los datos están en varios sitios y pueden no coincidir debido a actualizaciones incorrectas.
- **Aislamiento de los datos:** Los datos están en archivos con diferentes formatos, por lo tanto, resultan difíciles de utilizar en nuevos programas.
- **Dependencia de los datos física-lógica:** Puesto que la estructura física de los datos (definición de ficheros y registros) se encuentra codificada en los programas de aplicación, cualquier cambio en dicha estructura supone modificar también todos los programas que la manejen.
- **Variedad de usuarios:** Si varios usuarios actualizan a la vez se puede llegar a tener información inconsistente
- **Esfuerzo excesivo de programación**
- **Problemas de integridad de datos:** Es decir, los datos tienen que cumplir restricciones de consistencia. Por ejemplo, no se puede poner la nota a un alumno en una asignatura que no se ha creado ni eliminar un curso si antes no se han eliminado los alumnos que se habían incluido en él.

La **integridad referencial** es una propiedad deseable en las bases de datos. Gracias a la integridad referencial se garantiza que una entidad siempre se relacione con otras entidades válidas, es decir, que existan en la base de datos. Implica que en todo momento dichos datos sean correctos, sin repeticiones innecesarias, datos perdidos y relaciones mal resueltas.

1.3 Definición de base de datos

Para solucionar todos estos problemas se desarrollan las bases de datos.

Una base de datos es un **conjunto estructurado de datos relacionados entre sí, con características de integridad máxima, redundancia mínima y posibilidad de compartir el contenido entre usuarios.**

La redundancia de los datos debe ser controlada, de forma que no existan duplicidades perjudiciales ni innecesarias, y que las redundancias físicas, convenientes muchas veces a fin de responder a objetivos de eficiencia, sean tratadas por el mismo sistema, de modo que no puedan producirse incoherencias. Por tanto, un dato se actualizará lógicamente por el usuario de forma única, y el sistema se preocupará de cambiar físicamente todos aquellos campos en los que el dato estuviese repetido, en caso de existir redundancia física.

Dos características inherentes a los sistemas de bases de datos son la separación entre los programas de aplicación y los datos y el manejo de múltiples vistas por parte de los usuarios.

1.4 Términos relacionados con las bases de datos

Existen un montón de términos asociados con las bases de datos y que iremos viendo en posteriores temas. Algunos de estos conceptos son:

- **Dato:** Es un trozo de información concreta sobre algún concepto. Ej: 1992 es un número que representa un año de nacimiento de una persona. Los datos se caracterizan por pertenecer a un tipo.
- **Tipo de dato:** Indica la naturaleza del campo. Ej: numérico, alfanumérico...
- **Campo:** Es el identificador de toda una familia de datos. Cada campo pertenece a un tipo de datos. Ej: La fecha de nacimiento pertenece al tipo de datos Fecha. También se le llama columna.
- **Registro:** Es una recolección de datos referentes a un mismo concepto. Ej: Los datos de una persona pueden ser su NIF, año de nacimiento, nombre, dirección... A los registros también se les llama tuplas o filas.
- **Campo clave:** Es un campo especial que identifica de forma única a cada registro. Ej: el NIF.
- **Tabla:** Es un conjunto de registros bajo un mismo nombre que representa el conjunto de todos ellos. Ej: los clientes de una empresa se almacenan en una tabla cuyo nombre es Clientes.
- **Consulta:** Es una instrucción para hacer peticiones a una base de datos. Veremos que existen consultas de búsqueda, pero también de eliminación, actualización o inserción de registros.
- **Índice:** Es una estructura que almacena los campos clave de una tabla, organizándolos para hacer más fácil encontrar y ordenar los registros de esa tabla. Su funcionamiento es similar al de un libro y guarda parejas de elementos: el elemento que se desea indexar y su posición en la base de datos.
- **Vista:** Es una transformación que se hace a una o más tablas para obtener una nueva tabla. Esta tabla será virtual, es decir, no está almacenada en los dispositivos de almacenamiento del ordenador, aunque sí se almacena su definición.
- **Informe:** Es un listado ordenado de los campos y registros seleccionados en un formato fácil de leer. Ej: Informe de las facturas impagadas del mes de enero ordenado por nombre de cliente.
- **Guiones:** O scripts. Son un conjunto de instrucciones que, ejecutadas de forma ordenada, realizan operaciones avanzadas de mantenimiento de los datos almacenados en la base de datos.

1.5 Arquitectura de las bases de datos

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. Este comité propuso una arquitectura general basada en tres niveles o esquemas: el nivel físico, o de máquina, el nivel externo, o de usuario, y el nivel conceptual. Así mismo describió las interacciones entre estos tres niveles y todos los elementos que conforman cada uno de ellos.

Estos niveles son:

- **Nivel interno o físico:** Se relaciona directamente con el almacenamiento físico de los datos. En él se describe cómo se almacenan realmente los datos, en qué archivos, su nombre y dirección. También estarán los registros, longitud, campos, índices y las rutas de acceso a esos archivos. Se definen también los métodos de acceso.

- **Nivel conceptual o lógico:** Es el nivel donde se lleva a cabo el diseño global de la BD, estableciendo las entidades, relaciones, reglas de integridad, autorizaciones y restricciones a usuarios... En él se describen cuáles son los datos reales almacenados en la BD y que relaciones existen entre ellas. Este nivel lo definen los administradores de la BD, que son los que deciden qué información se guarda en la BD, y corresponde a la estructura organizacional de los datos obtenida al reunir los requerimientos de todos los usuarios, sin preocuparse de su organización física ni de las vías de acceso.

Podría contener:

- Entidades del mundo real (clientes, artículos, pedidos, ...)
- Atributos de las entidades (nombre_cliente, NIF, ...)
- Asociaciones entre entidades (compra de artículos)
- Restricciones de integridad (son las normas que deben cumplir los datos).

- **Nivel externo o nivel de vistas de usuario:** Es el nivel más cercano al usuario y representa la percepción individual de cada usuario. Si los niveles interno y conceptual describen toda la BD, este nivel describe únicamente la parte de datos para un usuario o grupo de usuarios. Habrá usuarios que podrán acceder a más de un esquema externo y uno de éstos puede ser compartido por varios usuarios. Se protege así el acceso a los datos por parte de personas no autorizadas. Habrá tantos niveles como usuarios, pues cada uno podrá realizar sólo ciertas funciones, salvo el administrador que tendrá acceso a todas.

Para una BD habrá un único esquema interno, un único esquema conceptual, pero puede haber varios esquemas externos.

Hay que destacar que los tres esquemas no son más que descripciones de los mismos datos, pero con distintos niveles de abstracción. Los únicos datos existentes son los que están a nivel físico. El SGBD debe transformar cualquier petición expresada en términos de un esquema externo a una petición expresada en términos del nivel conceptual y luego a una en el esquema interno, que se procesará sobre la base de datos almacenada. El proceso de transformar peticiones y resultados de un nivel a otro se denomina correspondencia o transformación.

La arquitectura de tres niveles es útil para explicar el concepto de **independencia de datos**, que podemos definir como la *capacidad para modificar el esquema en un nivel del sistema sin tener que modificar el esquema del nivel inmediato superior*. Se pueden definir dos tipos de independencia de datos:

La independencia lógica: capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación. Se puede modificar el esquema conceptual para ampliar la base de datos o para reducirla. Si, por ejemplo, reduzco la base de datos eliminando una entidad, los esquemas externos que no se refieran a ella no deberán verse afectados.

La independencia física: capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual (o los externos). Por ejemplo, puede ser necesario reorganizar ciertos ficheros físicos para mejorar el rendimiento de las operaciones de consulta o actualización.

Además de la independencia física y lógica de datos asociada a la arquitectura de tres niveles existen una serie de **características asociadas a dicho estándar**. Estas son:

- **Facilidad de uso:** las personas que no sepan nada sobre bases de datos deben poder hacer consultas sin hacer referencia a los componentes técnicos de la base de datos.
- **Administración centralizada:** El administrador manipulará datos y verificará la integridad de la base de datos de forma centralizada.
- **Redundancia centralizada:** El SGBD debe poder evitar la redundancia de datos siempre que sea posible, para minimizar errores y para ahorrar memoria.
- **Verificación de integridad:** Los datos deben ser internamente coherentes y, cuando algunos elementos referencien a otros, estos últimos tienen que estar presentes.
- **Uso compartido de datos:** El SGBD debe permitir que múltiples usuarios accedan simultáneamente a la base de datos.
- **Seguridad de los datos:** El SGBD debe poder administrar los derechos de acceso a los datos de cada usuario.

1.6 Clasificación de las bases de datos

Las bases de datos pueden clasificarse en virtud de diferentes criterios:

1. Según el modelo utilizado

- Modelo de datos jerárquico**
- Modelo de datos en red**
- Modelo de datos relacional**

Es el más usado actualmente.

El modelo relacional se basa en el concepto matemático de *relación*, que gráficamente se representa mediante una tabla. Una *relación* es una tabla con columnas y filas, mediante las que vamos a representar las entidades y las interrelaciones entre ellas, así como los atributos o características de esas entidades. Trataremos más adelante de todas sus características, pues es el modelo en el que nos vamos a centrar.

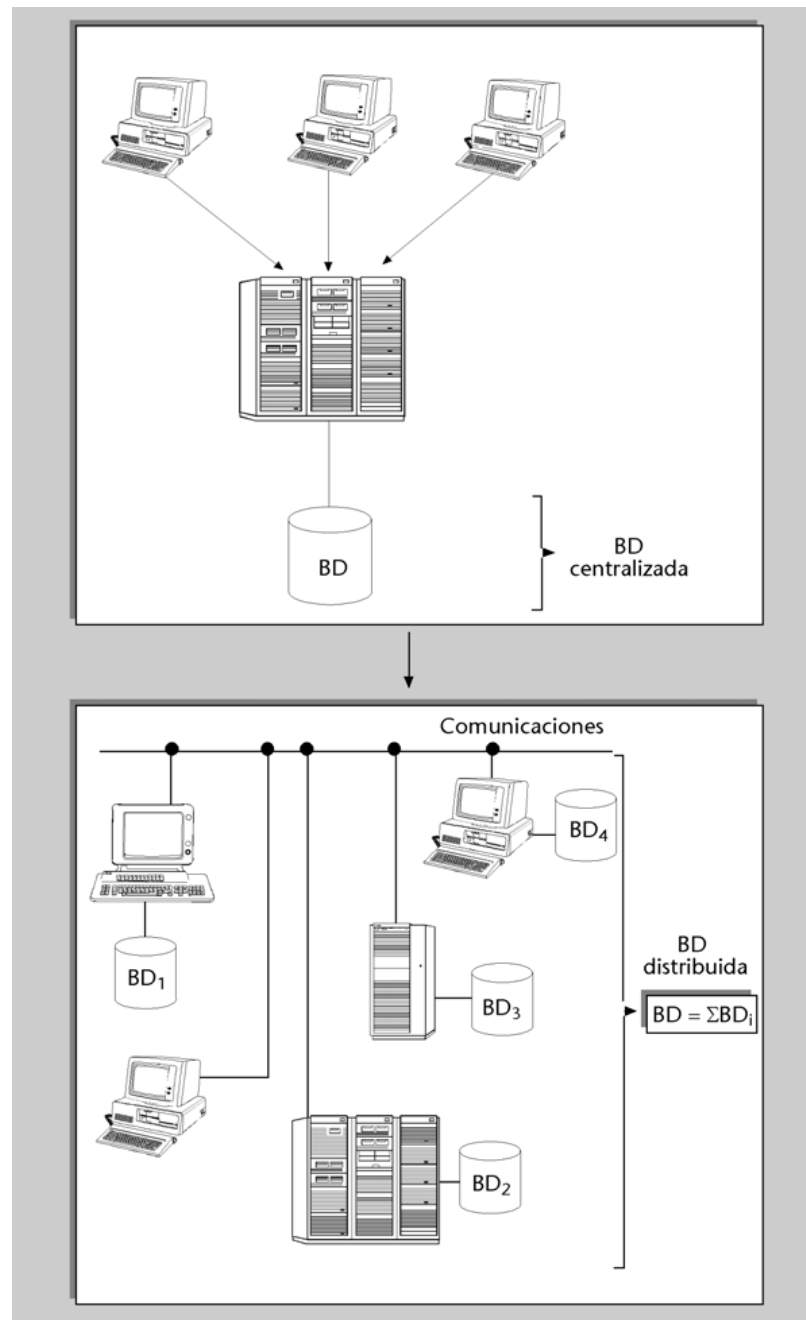
2. Según la distribución de los datos

a. Modelo de datos centralizado

Una base de datos centralizada es una **base** de datos que está físicamente situada en un único lugar, controlado por un solo servidor u ordenador.

b. Modelo de datos distribuido

Un ordenador o programa puede trabajar con diferentes BD como si se tratase de una sola.



1.7 El SGBD (Sistema Gestor de Bases de Datos)

Además de los datos, en la base de datos se guardarán los **metadatos**, que son la estructura donde se van a almacenar los datos.

El SGBD es una aplicación que **permite a los usuarios definir, crear y mantener la BD, y proporciona un acceso controlado a la misma**. Debe prestar los siguientes servicios:

Creación y definición de la BD. El SGBD **permite la especificación de la estructura, el tipo de los datos, las restricciones y las relaciones entre ellos mediante "lenguajes de definición de datos"**. Toda esta información se almacena en el **diccionario de datos**.

- **Manipulación de los datos.** El SGBD permite realizar consultas sobre los datos almacenados, inserciones de nuevos datos, y actualizaciones sobre los datos ya existentes, utilizando “*lenguajes de manipulación de datos*”.
- **Acceso controlado a los datos de la BD.** El SGBD permite mantener la seguridad para controlar el acceso no autorizado a los datos por usuarios sin permisos establecidos.
- **Mantenimiento de la integridad y consistencia.** El SGBD permite mantener la coherencia de los datos a través de mecanismos correctores que controlan los cambios no autorizados en los datos.
- **Independizan los datos de la aplicación o usuario,** haciendo más fácil su migración a otras plataformas.
- **Proporcionan un diccionario de metadatos,** que contiene el esquema de la base de datos en lo que se refiere a tablas, registros, campos, relaciones entre los datos, usuarios, permisos...
- **Acceso compartido a la BD.** El SGBD permite la interacción entre usuarios concurrentes. Se ocupará de:
 - **Gestión de los permisos:** Se necesita un método para asegurar que cada usuario sólo pueda acceder a la información para la que está autorizado y realizar sólo las operaciones que le estén permitidas.
 - **Gestión de la concurrencia:** Cuando varios usuarios quieran realizar operaciones que afecten a los mismos datos u operaciones excluyentes entre sí, ha de existir un mecanismo por el que se impida que la incorrecta realización de las operaciones cause pérdida o inconsistencia de datos. Uno de los mecanismos usados se llama BLOQUEO que consiste en que cuando dos usuarios intentan realizar operaciones incompatibles, el sistema bloquea a uno de ellos hasta que el otro termina, para que luego el usuario bloqueado lleve a cabo su tarea.
- **Mecanismos de copias de seguridad y recuperación de los datos.** El SGBD permite, tras un fallo del sistema que ha provocado errores en los datos, recuperar los datos desde algún sistema de almacenamiento externo donde se realizó una copia de seguridad recientemente.

1.8 COMPONENTES DE LOS SGBD

- **Lenguajes de los SGBD**

Todos los SGBD ofrecen lenguajes e interfaces apropiadas para cada tipo de usuario: administradores, diseñadores, programadores de aplicaciones que acceden a la BD y usuarios finales. Dichos lenguajes se clasifican en:

- **Lenguaje de Definición de Datos (LDD o DDL):** Este lenguaje permite crear toda la estructura de una base de datos (desde tablas hasta usuarios). Sus cláusulas son del tipo DROP (eliminar objetos) y CREATE (crear objetos)
- **Lenguaje de Manipulación de Datos (LMD o DML):** Este lenguaje permite con 4 sentencias sencillas seleccionar determinados datos (SELECT), insertar datos (INSERT), modificarlos (UPDATE) e incluso borrarlos (DELETE).

- **Lenguaje de control de datos (LCD o DCL):** Incluye comandos (GRANT y REVOKE) que permiten al administrador **gestionar el acceso a los datos contenidos en la base de datos**. Permiten conceder o suprimir privilegios a los usuarios, es decir, realiza el control del acceso a los datos. Con este lenguaje se establecen las vistas de los usuarios, así a cada usuario se le permite manipular únicamente el conjunto de datos que le interesan, y se le deniega el acceso a los datos que no necesita.
- **Lenguaje TCL: o lenguaje de control de transacciones.** El propósito de este lenguaje es **permitir ejecutar varios comandos de forma simultánea** como si fuera un comando indivisible. **Si es posible** ejecutar todos los comandos se aplica la transacción (COMMIT) y, si en algún paso de la ejecución **sucede algo inesperado**, se pueden deshacer todos los pasos dados (ROLLBACK).

- **El diccionario de datos**

Es el **lugar donde se guarda información acerca de todos los datos que forman la BD: su descripción y la de los objetos que la forman.** En una BD relacional, el diccionario de datos proporciona información acerca de:

- La estructura lógica y física de la BD. Esquemas externo, conceptual e interno, y correspondencia entre los esquemas.
- Las definiciones de todos los objetos de la BD: tablas, vistas, índices, disparadores, procedimientos, funciones, etc.
- El espacio asignado y utilizado por los objetos.
- Los valores por defecto de las columnas de las tablas.
- Información acerca de las restricciones de integridad.
- Los privilegios y roles otorgados a los usuarios.
- Estadísticas de utilización, tales como la frecuencia de las transacciones y el número de accesos realizados a los objetos de la base de datos.
- Se puede tener un historial de los cambios realizados sobre la base de datos.

- **Seguridad e integridad de los datos**

Un SGBD **proporciona los siguientes mecanismos** para garantizar la seguridad e integridad de los datos:

- Debe garantizar la protección de los datos contra accesos no autorizados, tanto intencionados como accidentales.
- Debe implantar restricciones de integridad que protegerán la BD contra daños accidentales, pues los valores de los datos que se quieren almacenar deberán satisfacer ciertos tipos de restricciones de consistencia y reglas de integridad, especificadas por el administrador de la BD. El SGBD puede determinar si se produce una violación de la restricción impuesta.
- Debe proporcionar herramientas y mecanismos para la planificación y realización de copias de seguridad, y su posible posterior restauración tras un fallo del sistema.
- Debe ser capaz de recuperar la BD llevándola a un estado consistente en caso de ocurrir algún suceso que la dañe.

- Debe asegurar el acceso concurrente y ofrecer mecanismos para conservar la consistencia de los datos en el caso de que varios usuarios actualicen la BD de forma concurrente.

1.9 USUARIOS DE LOS SGBD

Generalmente distinguimos **cuatro grupos de usuarios** de sistemas gestores de bases de datos: los usuarios administradores, los diseñadores de la base de datos, los programadores y los usuarios de aplicaciones que interactúan con las bases de datos.

- **Administrador de la base de datos:** Se encarga del **diseño físico de la base de datos y de su implementación, realiza el control de la seguridad y de la concurrencia, mantiene el sistema para que siempre esté operativo y se encarga de que los usuarios y las aplicaciones obtengan buenas prestaciones.** El administrador debe conocer bien el SGBD y el equipo informático sobre el que esté funcionando.
- **Diseñadores de 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 debe conocer muy bien los datos de la empresa y 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 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 usar, es independiente de los programas de aplicación, de los lenguajes de programación y de otras consideraciones físicas.
- **Programadores de aplicaciones:** Se encargan de **implementar los programas de aplicación que servirán a los usuarios finales.** Estos programas de aplicación son los que permiten consultar datos, insertarlos, actualizarlos y eliminarlos.
- **Usuarios finales:** Clientes de la base de datos que hacen uso de ella sin conocer su funcionamiento y organización.