

TEMA 1: SISTEMAS DE BASES DE DATOS

1. DE LOS SISTEMAS TRADICIONALES DE FICHEROS A LAS BASES DE DATOS.
2. PROBLEMAS DE LOS FICHEROS.
3. CONCEPTO DE BASE DE DATOS.
4. VENTAJAS DE LAS BASES DE DATOS FRENTE A LOS FICHEROS CLÁSICOS.
5. INCONVENIENTES DE LAS BASES DE DATOS.
6. CARACTERÍSTICAS DESEABLES DE LAS BD.
7. COMPONENTES DE UN SISTEMA DE BASE DE DATOS.
8. NIVELES DE ABSTRACCIÓN EN UNA BASE DE DATOS. ARQUITECTURA ANSI/SPARC.
9. ESTRUCTURA GENERAL DEL SGBD.
10. GESTOR DE LA BD.
11. SISTEMAS GESTORES DE BASES DE DATOS (OBJETIVOS Y FUNCIONES).
12. EL MODELO DE DATOS.

1. DE LOS SISTEMAS TRADICIONALES DE FICHEROS A LAS BASES DE DATOS

Los primeros sistemas de procesamiento de datos estaban pensados para ejecutar las tareas administrativas, para reducir el papeleo. Más recientemente, los sistemas se han extendido hacia la producción y gestión de la información, que se ha convertido en un recurso vital para las empresas. Si los directivos disponen de una buena información, es probable que puedan tomar decisiones acertadas y con una gran rapidez.

Actualmente, la función más importante de los sistemas de bases de datos consiste en proporcionar la materia prima necesaria a los sistemas de información de la empresa para la gestión de la misma. El desarrollo de los sistemas de bases de datos se convirtió en crucial para proporcionar información correcta (sin errores) y oportuna (en el momento que se necesita), a los directivos.

Si analizamos la situación de algunos sistemas de información, encontramos una proliferación de ficheros específicos cada uno de ellos de una determinada aplicación. Los datos se recogen varias veces y se encuentran repetidos en los distintos archivos. Esta redundancia además de malgastar recursos, origina a menudo divergencias en los resultados.

Los sistemas informáticos tradicionales han sido llamados por algunos autores sistemas orientados hacia el 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.

Este planteamiento produce, además de una ocupación inútil de memoria secundaria, 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. Los problemas son aún más acusados cuando se presentan demandas inesperadas de información.

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.

2. PROBLEMAS DE LOS FICHEROS

- Redundancia e inconsistencia de los datos
- Dificultad de acceso a los datos: Existen aplicaciones particulares para cada tipo de acceso a los datos.
- Aislamiento de los datos: Los datos están en archivos con diferentes formatos, por lo tanto resultan difíciles de utilizar en nuevos programas.
- Variedad de usuarios: Si varios usuarios actualizan a la vez se puede llegar a tener información inconsistente
- Problemas de seguridad: Es difícil restringir el acceso a registros de un fichero.
- Problemas de integridad de los datos.

3. CONCEPTO DE BASE DE DATOS

Una base de datos es un conjunto, colección o depósito de datos almacenados en un soporte informático de acceso directo. Los datos deben estar interrelacionados y estructurados.

Dada la importancia que tienen en el mundo real las interrelaciones entre los datos, es imprescindible que la base de datos sea capaz de almacenar éstas interrelaciones, al igual que hace con otros elementos (como las entidades y atributos), siendo ésta una diferencia esencial respecto a los ficheros donde no se almacenan las interrelaciones.

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.

La actualización y recuperación en las bases de datos debe realizarse mediante procesos bien determinados, incluidos en un conjunto de programas que se encargan de la gestión de la base de datos y que se denominan sistemas gestores de bases de datos (S.G.B.D); procedimientos que han de estar diseñados de modo que se mantenga la integridad, seguridad y confidencialidad de la base.

El concepto de base de datos ha ido cambiando y configurándose a lo largo del tiempo, en la actualidad, y de acuerdo con estas características que acabamos de analizar, podemos definir la base de datos como:

“Colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real; los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de éstas, y su definición y descripción, únicas para cada tipo de datos, han de estar almacenadas junto con los mismos. Los procedimientos de actualización y recuperación comunes y bien determinados, habrán de ser capaces de conservar la integridad, seguridad y confidencialidad del conjunto de los datos”

4. VENTAJAS DE LAS BASES DE DATOS FRENTE A LOS FICHEROS CLÁSICOS

Las bases de datos, surgidas como respuesta al nuevo planteamiento de los sistemas orientados hacia los datos, para mejorar la calidad de las prestaciones de los sistemas informáticos y aumentar su rendimiento, presentan una multitud de ventajas frente a los sistemas clásicos de ficheros, debido, sobre todo, a que se basan en una estructura de datos integrada y centralizada, eliminando así los problemas de redundancia y control de los datos.

Las ventajas de los sistemas de bases de datos son, entre otras, las siguientes:

A) Independencia de los datos respecto a los tratamientos y viceversa:

La mutua independencia de datos y tratamientos lleva a que un cambio de los programas no implican tener que cambiar el diseño lógico y/o físico de la base de datos. Por otra parte, la inclusión de nuevas informaciones, desaparición de otras, cambios en la estructura física o en los caminos de acceso, etc., no deben obligar a alterar los programas. Esta independencia de los tratamientos frente a la estructura de la base de datos, evita el importante esfuerzo que origina la reprogramación de las aplicaciones cuando se producen cambios en los datos.

- Independencia lógica de los datos: Se refiere a que las modificaciones de la representación lógica del problema no afecta a los programas que los manipulan, y viceversa.
- Independencia física de los datos: Se refiere a que la distribución en unidades de almacenamiento es independiente de la estructura lógica general, y viceversa.

B) Coherencia de los resultados:

Debido a que la información de la base de datos se recoge y almacena una sola vez. En todos los programas se utilizan los mismos datos, por lo que los resultados de todos ellos son coherentes y perfectamente comparables.

Además, al no existir (o al menos disminuir en gran medida) la redundancia en los datos, desaparece el problema que se presentaba en el enfoque clásico, de que el cambio de un dato obligaba a actualizar una serie de ficheros. De esta forma se elimina también el inconveniente de las divergencias en los resultados debidas a actualizaciones no simultáneas en todos los ficheros.

C) Mejor disponibilidad de los datos para el conjunto, de los usuarios:

Cuando se aplica la metodología de bases de datos, cada usuario ya no es propietario de los datos, puesto que éstos se comparten entre el conjunto de aplicaciones, existiendo una mejor disponibilidad de los datos para todos los que tienen necesidad de ellos, siempre que estén autorizados para su acceso.

D) Mayor eficiencia en la recogida, validación entrada de los datos al sistema:

Al no existir apenas redundancias, los datos se recogen y validan una sola vez, aumentando así el rendimiento de todo el proceso previo al almacenamiento.

E) Reducción del espacio de almacenamiento:

La desaparición (o disminución) de las redundancias, así como la aplicación de técnicas de compactación, lleva en los sistemas de bases de datos a una menor ocupación de almacenamiento secundario.

5. INCONVENIENTES DE LAS BASES DE DATOS

Las bases de datos no sólo presentan ventajas, sino que también tienen posibles inconvenientes, que es necesario valorar antes de tomar una decisión relativa a un cambio en la orientación del SI. Entre estos inconvenientes es preciso destacar:

A) Instalación costosa:

La implantación de un sistema de bases de datos puede llevar consigo un coste elevado, tanto en equipo físico (nuevas instalaciones o ampliaciones), como en el lógico (sistemas operativos, programas, compiladores, etc... necesarios para su uso).

B) Personal especializado:

Los conocimientos, que resultan imprescindibles para una utilización correcta y eficaz y sobre todo para la administración de las bases de datos, implican una necesidad de personal especializado que resulta difícil de encontrar, y de formar. El problema de la contratación y formación de este tipo de personal es clave a la hora de crear un sistema de base de datos.

C) Implantación larga y difícil:

La implantación de una base de datos puede convertirse en una tarea larga y laboriosa. Las dificultades que van apareciendo a lo largo de su desarrollo llevan en general a que se superen ampliamente los plazos inicialmente previstos.

D) Falta de rentabilidad a corto plazo:

La implantación de un sistema de bases de datos, tanto por su coste en personal y en equipos como por el tiempo que tarda en estar operativo, no resulta rentable a corto plazo. Puede calcularse que para un sistema de dimensiones medias la rentabilidad sólo puede empezar a apreciarse después de bastantes meses de la iniciación de los trabajos; en instalaciones grandes o muy grandes el plazo puede llegar a ser de años.

6. CARACTERÍSTICAS DESEABLES DE LAS BD

- ☐ Versatilidad para representar la información: Ofrecer diferentes visiones de la información que se almacena en la BD.
- ☐ Desempeño: Debe dar respuesta en un tiempo adecuado, permitiendo el acceso simultáneo al mismo o a diferentes datos.
- ☐ Mínima redundancia.
- ☐ Capacidad de acceso: Debe responder en tiempo adecuado a consultas previstas e imprevistas.
- ☐ Simplicidad: Cambios en los requerimientos no deben suponer grandes cambios en el modelo de datos.
- ☐ Seguridad: Capacidad para proteger los datos contra pérdidas totales y/o parciales,
 - Contra destrucción causada por el entorno (fuego, inundación...).
 - Contra destrucción causada por fallos del sistema.
 - Contra accesos no autorizados a la BD.
 - Contra accesos indebidos a los datos.
- ☐ Privacidad: Debe reservar la información de accesos de personas no autorizadas.
- ☐ Afinación: Organización de datos afines para obtener buenos tiempos de respuesta.

- ☐ Integridad: Que los datos sean correctos y se correspondan a los requerimientos del dominio.
 - Integridad frente a fallos Hw o Sw o de acceso concurrente.
 - Integridad asegurando que los datos se ajustan a los requerimientos del problema.

7. COMPONENTES DE UN SISTEMA DE BASE DE DATOS

Un sistema de base de datos es algo más que simples datos o que un conjunto de datos en combinación con unos programas de gestión. Un sistema de base de datos está formado por los siguientes componentes:

Datos

Las características más importantes de la información en estos sistemas es que va a estar integrada y compartida.

Integrada: La Base de datos puede considerarse como una unificación de varios ficheros de datos, que son tratados como uno solo, y en el que se ha eliminado totalmente, o en parte, la redundancia de datos.

Compartida: Los datos pueden compartirse entre varios usuarios distintos. Es posible que varios de estos usuarios accedan al mismo tiempo al mismo elemento de información (acceso concurrente).

Equipo (Hardware)

Conjunto de dispositivos físicos utilizados para almacenar y procesar los datos.

- ☐ Ordenadores, utilizados para procesar los datos de la Base de datos: pueden ser mainframes, miniordenadores u ordenadores personales. Los mainframes y los miniordenadores fueron utilizados tradicionalmente para soportar el acceso de varios usuarios a una base de datos común. Los ordenadores personales eran empleados, inicialmente, para manejar bases de datos autónomas controladas y manipuladas por un usuario único. No obstante, actualmente, también pueden conectarse a una red cliente/servidor, garantizando el acceso de varios usuarios a una base de datos común almacenada en unidades de disco y controladas por un ordenador servidor. El servidor puede ser otro ordenador personal más potente, o bien, un miniordenador o un mainframe.
- ☐ Volúmenes de almacenamiento. Generalmente son unidades de disco que constituyen el mecanismo de almacenamiento principal para las bases de datos.
- ☐ Otros dispositivos, como unidades de cinta, terminales, impresoras, etc.

Programas (Software)

Un sistema de base de datos incluye dos tipos de programas:

- ☐ El software de propósito general, para la gestión de la base de datos, comúnmente llamado Sistema Gestor de Bases de Datos (SGBD, o también DBMS, en inglés). El SGBD maneja todas las solicitudes de acceso a la base de datos formuladas por los usuarios y los programas de aplicación.
- ☐ El software de aplicación, que usa las facilidades del SGBD para manipular la base de datos con el fin de llevar a cabo una función específica en la gestión de la empresa (por ejemplo: la gestión de almacén). Puede estar desarrollado en un lenguaje de programación estándar, tal como COBOL o C, o en un lenguaje propio de los SGBD denominados lenguajes de cuarta generación (4GL).

Personal

En un sistema de base de datos intervienen un número importante de usuarios, que podemos clasificar en tres grupos:

- ☐ Administrador de la base de datos (A.B.D.). Son los encargados de diseñar la estructura de la base de datos y los responsables de que el sistema funcione correctamente. El A.B.D. se encarga de autorizar el acceso a la base de datos, de coordinar y vigilar su utilización y de adquirir los recursos necesarios de software y hardware. El A.B.D. es el responsable cuando surgen problemas como violaciones de seguridad o una respuesta lenta del sistema. El A.B.D. tiene, entre otras, las siguientes funciones:
 - Definición del esquema: Decidir el contenido de la base de datos, eligiendo cuales son los datos que interesa tener almacenados y organizarlos de la mejor forma posible, creando el esquema conceptual, que se escribirá mediante un lenguaje de definición de datos (DDL).
 - Definición de las estructuras de almacenamiento y método de acceso: Debe decidir sobre la forma en que se van a almacenar los datos sobre los soportes físicos en los que se grabará la base de datos y la correspondencia entre esta estructura de almacenamiento y el esquema conceptual.
 - Modificación del esquema y de la organización física si los requerimientos cambian.
 - Decidir los controles de autorizaciones para el acceso a los datos: Es el que concede diferentes tipos de autorizaciones al resto de los usuarios de la base de datos.
 - Especificar las restricciones de integridad: Debe definir los procedimientos de validación que habrán de ejecutarse cada vez que se actualiza la base de datos. Estas restricciones son consultadas por el SGBD cada vez que se realiza una actualización de los datos.
- ☐ Programadores de aplicaciones, que se encargan de desarrollar las aplicaciones que manejan datos de la base de datos. Estas aplicaciones contendrán solicitudes de datos al S.G.B.D que luego serán procesados por los programas de la aplicación que tendrán como finalidad resolver problemas específicos de la empresa.

- Usuarios finales, que son personas que no tienen por que tener conocimientos informáticos y que pueden manipular los datos (examinarlos y actualizarlos) con la ayuda de las aplicaciones, o bien de lenguajes de consulta no procedimentales (no es necesario indicar el algoritmo de acceso a los datos), tipo SQL, o bien, mediante herramientas basadas en sistemas de menús. Se distinguen tres tipos de usuarios finales:

- Usuarios especializados: Aquellos que son capaces de escribir ciertas aplicaciones para la BD, para su uso propio.
- Usuarios casuales: Aquellos que realizan consultas a través de un procesador de consultas. Esas consultas pueden ser creadas por ellos mismos o por otras personas.
- Usuarios ingenuos: Aquellos que solo acceden a través de aplicaciones previamente escritas por otros usuarios.

8. NIVELES DE ABSTRACCIÓN EN UNA BASE DE DATOS. ARQUITECTURA ANSI/SPARC

Antes de analizar el concepto de S.G.B.D, es preciso exponer, siquiera globalmente y sin entrar en detalles, los distintos niveles de abstracción de una base de datos. Esto nos servirá, mas adelante, para identificar las diferentes funciones que han de cumplir estos sistemas.

Se puede observar en los SI la existencia de dos estructuras distintas, la lógica (vista del usuario) y la física (forma en que se encuentran los datos en el almacenamiento). En las bases de datos aparece un nuevo nivel de abstracción que se ha denominado de diversas maneras: nivel conceptual, estructura lógico global, esquema, etc. Esta estructura intermedia pretende una representación global de los datos que se interponga entre las estructuras lógica y física y que sea independiente, tanto del equipo como de cada usuario en particular.

ANSI/SPARC es un grupo de normalización creado en 1969 para estudiar el impacto de los S.G.B.D. en los sistemas de información y cuyos resultados, publicados en 1975 propusieron el uso de tres niveles de descripción de datos:

1. Nivel interno o físico. Se refiere al almacenamiento físico en el se describe cómo se almacenan realmente los datos en memorias secundarias, en qué archivos, su nombre y dirección. También estarán los registros, longitud, campos, índices y las rutas de acceso a esos archivos.
2. Nivel Conceptual. En el 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 que información se guarda en la BD. Este nivel 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).
3. Nivel externo o vistas. 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. A la hora de construir un esquema externo:
- Se pueden omitir una o más entidades del sistema.
 - Se pueden omitir uno o más atributos de una entidad.
 - Se pueden omitir una o más relaciones entre los datos.
 - Se pueden cambiar el orden de los atributos.

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

9. ESTRUCTURA GENERAL DEL SGBD

1. Gestor de ficheros: Se encarga de las estructuras de datos que se emplean para representar la información almacenada en disco. Además es el encargado de asignar espacio en disco.
2. Gestor de BD: Es el interface entre los datos de bajo nivel almacenados en la BD y los programas de aplicaciones y consultas de datos.
3. Procesador de consultas: traduce las proposiciones del lenguaje de consultas a instrucciones de bajo nivel para que las entienda el gestor de BD. Además el procesador de consultas trata de traducir una petición del usuario a otra equivalente pero más eficiente.
4. Compilador DDL: Convierte las proposiciones DDL en un conjunto de tablas que contienen metadatos.
5. Ficheros de datos: Almacenan la información (los datos).
6. Diccionario de datos: Almacena información referida a la estructura de la BD. Se usa continuamente. Se debe poner un gran énfasis en un buen diseño y una implementación adecuada del diccionario de datos.

10. GESTOR DE LA BD

Las funciones del Gestor de BD son:

- ☐ Interactuar con el gestor de ficheros: Traduce proposiciones DML a comandos del sistema de archivos. El gestor de ficheros asigna espacio en disco, localiza el fichero, lleva el control de los ficheros, ...
- ☐ Es responsable de conservar la integridad: Comprueba si los datos almacenados en la BD satisfacen las restricciones de integridad (consistencia).
- ☐ Es responsable de garantizar la seguridad: Evita accesos indebidos.
- ☐ Es responsable de respaldo y recuperación: Para evitar pérdidas de información el gestor BD debe detectar fallos y restaurar la BD al estado que tenía antes de que ocurriera el fallo.
- ☐ Control de concurrencia: Cuando varios usuarios actualizan la BD de forma concurrente (a la vez) puede que no se conserve la consistencia de los datos, con lo cual el sistema debe controlar la interacción entre los usuarios.

11. SISTEMAS GESTORES DE BASES DE DATOS (OBJETIVOS Y FUNCIONES)

Si disponemos de una base de datos, necesitamos un conjunto de herramientas para poder acceder a los datos. La forma de acceso va a depender de la tarea a realizar, así:

- El administrador añadirá usuarios, modos de acceso, reorganizará la BD cuando sea necesario,...
- Los analistas y programadores deberán conocer la estructura de la BD para poder hacer aplicaciones y programas.
- Los usuarios finales utilizarán la BD a través de aplicaciones o lenguajes de consulta interactivos.

Para poder realizar todo esto necesitamos un sistema gestor de base de datos (SBGD) que se puede definir como el "conjunto de herramientas que suministra a todos (administrador, analistas, programadores, usuarios) los medios necesarios para describir, recuperar y manipular los datos almacenados en la BD, manteniendo la seguridad, integridad y confidencialidad de los mismos".

Objetivos de los S.G.B.D.

En un ambiente multiusuario el S.G.B.D ofrece a la empresa un control centralizado de su información. Los objetivos que se plantean estos sistemas están relacionados con la intención de evitar los problemas que existían en los sistemas de información orientados a los procesos. Los principales objetivos son:

- Evitar la redundancia de los datos, eliminando así la inconsistencia de los mismos.
- Mejorar los mecanismos de seguridad de los datos y la privacidad. Podemos distinguir cuatro tipos de contextos para usar mecanismos de seguridad: seguridad contra accesos indebidos a los datos, seguridad contra accesos no autorizados a la BD, seguridad contra destrucción causada por el entorno (fuego, inundación, robo, ...), seguridad contra fallos del propio sistema (fallos del hardware, del software, ...).
- Asegurar la independencia de los programas y los datos, es decir, la posibilidad de modificar la estructura de la base de datos (esquema) sin necesidad de modificar los programas de las aplicaciones que manejan esos datos.
- Mantener la integridad de los datos realizando las validaciones necesarias cuando se realicen modificaciones en la base de datos.
- Mejorar la eficacia de acceso a los datos, en especial en el caso de consultas imprevistas.

Funciones del S.G.B.D.

Las principales funciones que debe realizar un S.G.B.D. son:

- La definición de los datos.
- La manipulación de los datos.
- Garantizar la seguridad e integridad de los datos.
- La gestión de las transacciones y el acceso concurrente.

Definición de datos

Mediante el Lenguaje de Definición de datos (L.D.D.) el S.G.B.D. permite describir y definir los esquemas de la base de datos. Este lenguaje debe permitir:

- La creación de objetos conceptuales
- La descripción de estructuras físicas
- La definición de vistas

Para una base de datos existente, el L.D.D. debe permitir también:

- La modificación de la descripción de objetos conceptuales
- El borrado de objetos conceptuales
- La modificación y borrado de caminos de acceso
- La modificación de las vistas

Un L.D.D. está compuesto por un conjunto de comandos que actúan sobre los objetos conceptuales y sobre los objetos físicos. El conjunto de las descripciones de objetos de una base de datos se conoce con el nombre de diccionario de datos.

Manipulación de datos

La función de manipulación de datos se encarga de todas las operaciones de intercambio de datos entre los usuarios y la base de datos. Esta función se hace con la ayuda del Lenguaje de Manipulación de datos (L.M.D.), que está compuesto por un conjunto de comandos que nos permiten la consulta o puesta al día (inserción, modificación y borrado) de los datos de una base de datos.

Los L.M.D. funciona de forma diferente, según el modelo de datos:

- En los modelos en red y jerárquicos son, lenguajes procedimentales (se indica que datos se quieren y como se van a obtener esos datos), es decir, los programadores deben indicar el camino a seguir para acceder a los datos solicitados.
- En el modelo relacional son lenguajes declarativos o no procedimentales (se indica que datos se quieren pero no la forma de obtenerlos), y no es necesario indicar cómo se tiene que acceder a los datos, es suficiente indicar qué es lo que se desea obtener.

Seguridad e integridad de los datos

Se trata de garantizar la coherencia de los datos, comprobando que sólo los usuarios autorizados puedan efectuar las operaciones correctas sobre la base de datos. Esto se consigue mediante:

- Un control sobre los usuarios que acceden a la base de datos y los tipos de operaciones que están autorizados a realizar. Este control se llama gestión de autorizaciones, y permite crear o borrar usuarios y conceder o retirar derechos a efectuar determinados tipos de operaciones (por ejemplo: crear objetos, borrar objetos, modificar datos, etc.).
- La validación de las operaciones realizadas con los datos. Este control se hace mediante un conjunto de reglas llamadas restricciones de integridad. Existen varios tipos de restricciones de integridad, como por ejemplo, las restricciones de integridad referencial, que imponen que las modificaciones realizadas sobre algunos datos, obliguen a realizar modificaciones de otros datos con los que están enlazados (por ejemplo, si se modifica el código de un artículo, se debería modificar ese código en todos los pedidos que soliciten el artículo).
- Una protección de los datos contra los accesos malintencionados y los fallos. Los accesos malintencionados se suelen evitar con la asignación de palabras de paso (password) a los usuarios, la definición de vistas, protección física de los datos (encriptado de los datos). Con respecto a los fallos causados por manipulaciones incorrectas, o accidentes lógicos o físicos, los S.G.B.D. suelen disponer de utilidades de recuperación de los datos después de un fallo.

La correcta utilización de todas estas operaciones de seguridad e integridad constituye una tarea esencial del Administrador de la base de datos (gestión de usuarios y sus derechos, gestión de vistas y recuperación después de un fallo).

Gestión de transacciones y acceso concurrente

El principal objetivo de la implantación de una base de datos en un sistema de información de una organización, es poner a disposición de un gran número de usuarios un conjunto integrado de datos. El S.G.B.D. debe proteger los datos cuando estos sean manipulados simultáneamente por diferentes usuarios.

Una transacción se define como una unidad lógica de tratamiento (conjunto de órdenes) que aplicada a un estado coherente de la base de datos la deja, de nuevo, en un estado coherente, después de hacer las modificaciones. Una transacción solo se puede ejecutar completamente o ser anulada.

Para controlar la ejecución simultánea de transacciones es necesario una buena gestión de acceso concurrente que controle que la ejecución paralela de transacciones da el mismo resultado que una ejecución secuencial. Para controlar la ejecución paralela, la solución clásica consiste en bloquear momentáneamente los datos utilizados por una transacción hasta que finalice la actualización y las demás transacciones que demandan estos datos se ponen en espera hasta que quedan desbloqueados.

Un parámetro importante que afecta al rendimiento es el nivel de bloqueo de datos, llamado granularidad del bloque. Esta granularidad puede ser lógica (toda la base, una tabla, una línea) o física (segmento, fichero, página).

La consecuencia de esta técnica de bloqueos es el riesgo del interbloqueo (llamado bloqueo mortal), que sucede cuando dos o más transacciones se encuentran a la espera de que queden liberados datos que tiene bloqueados otra transacción que espera por los datos que están bloqueados por las transacciones anteriores. El S.G.B.D. debe estar capacitado para detectar esta situación y desbloquearla.

Otras funciones que deben realizar los S.G.B.D. son:

- Capacidad de auditoria: Para el administrador de la base de datos es muy importante conocer quien accede a la base de datos y que operaciones realiza. Un buen S.G.B.D debe poder proporcionarle esa información.
- Garantizar un tiempo de respuesta idóneo para el diálogo entre los usuarios y la base de datos en los procesos en línea.

12. EL MODELO DE DATOS

Una base de datos está definida generalmente por un conjunto de datos que representan mediante un modelo determinado un universo dado. Un modelo de datos no es más que un método conceptual para estructurar los datos. Existen tres modelos de datos fundamentales:

Modelo jerárquico

Consiste en que todas las interrelaciones de los datos se basan en jerarquías. Los archivos se conectan entre sí mediante punteros físicos (dirección física que indica donde puede encontrarse un registro sobre el disco) o campos añadidos a los registros individuales.

En una jerarquía un padre (registro propietario) puede tener muchos hijos (registro subordinado) pero un hijo sólo puede tener un padre. Por ejemplo si tenemos un fichero de facturas (FACTURA) y otro de líneas de factura (LINEA_F), el padre (registro propietario) sería los datos de la factura y los hijos (registros subordinados) las líneas de la factura.

Este modelo tenía algunas limitaciones ya que no todas las interrelaciones se pueden representar en una estructura jerárquica. Para intentar solucionar estas limitaciones se desarrollan los sistemas de base de datos en red. Un ejemplo de bases de datos jerárquica es IMS de IBM desarrollado a finales de los 60.

Modelo en red

Los sistemas de base de datos en red al igual que los jerárquicos utilizan punteros físicos. En este caso, un padre puede tener muchos hijos y un hijo puede tener muchos padres, es decir, un registro puede estar subordinado a registros de más de un archivo.

A principios de los 70 se desarrollaron y se comercializaron varios SGBD en red y este modelo de datos se normalizó como el modelo CODASYL. Ejemplos de bases de datos en red son ADABAS, TOTAL, IMAGE,...

Modelo relacional

El uso de punteros era simultáneamente una fortaleza y una debilidad de los sistemas de bases de datos jerárquicos y en red. Los punteros permitían una rápida recuperación de los datos, pero las interrelaciones tenían que definirse antes de que el sistema se pusiera en explotación. Era muy difícil recuperar datos basados en otras interrelaciones.

En 1970, E.F. Codd publica un artículo en el que argumenta que los datos deberían relacionarse mediante interrelaciones naturales, lógicas e inherentes a los datos, más que mediante punteros físicos. Codd propone así, un modelo simple de datos en el que todos ellos se representarían en tablas constituidas por filas y columnas. A estas tablas se les dio el nombre de relaciones y por eso se denominó al modelo relacional. Codd también propuso dos lenguajes para manipular los datos en las tablas: el álgebra relacional y el

cálculo relacional. La manipulación lógica de los datos también hace factible la creación de lenguajes de interrogación más accesibles para un usuario no especialista en programación. Actualmente los sistemas relacionales son un estándar en el mercado, especialmente en operaciones comerciales. Ejemplos de sistemas de bases de datos relacionales son: INFORMIX, DB2, ORACLE, SYBASE, DBASE, FOXPRO, INGRES, MYSQL, MS SQL SERVER, ...