



Preparador Informática

www.preparadorinformatica.com

TEMA 34. INFORMÁTICA

TEMA 36. S.A.I.

**SISTEMAS GESTORES DE BASES DE
DATOS. FUNCIONES. COMPONENTES.
ARQUITECTURAS DE REFERENCIA Y
OPERACIONALES. TIPOS DE SISTEMAS.**

TEMA 34 INF/ TEMA 36 SAI: SISTEMAS GESTORES DE BASES DE DATOS. FUNCIONES. COMPONENTES. ARQUITECTURAS DE REFERENCIA Y OPERACIONALES. TIPOS DE SISTEMAS.

1. INTRODUCCIÓN
2. SISTEMAS GESTORES DE BASES DE DATOS
 - 2.1. FUNCIONES
 - 2.2. COMPONENTES
 - 2.2.1. LENGUAJES DE LA BASE DE DATOS
 - 2.2.2. DICCIONARIO DE DATOS
 - 2.2.3. GESTOR DE LA BASE DE DATOS
 - 2.2.4. USUARIOS DE LA BASE DE DATOS
 - 2.2.5. HERRAMIENTAS DE LA BASE DE DATOS
 - 2.3. ARQUITECTURAS DE REFERENCIA Y OPERACIONALES
 - 2.3.1. ARQUITECTURA DE REFERENCIA
 - 2.3.2. ARQUITECTURA OPERACIONAL
 - 2.4. TIPOS DE SISTEMAS
 - 2.4.1. SEGÚN EL MODELO LÓGICO
 - 2.4.2. SEGÚN EL NÚMERO DE USUARIOS
 - 2.4.3. SEGÚN SU UBICACIÓN
3. SGBD COMERCIALES Y LIBRES
4. CONCLUSIÓN
5. BIBLIOGRAFÍA

1. INTRODUCCIÓN

Antes de aparecer los sistemas gestores de bases de datos (SGBD), la información se trataba y se gestionaba utilizando los típicos sistemas de gestión de archivos que iban soportados sobre un sistema operativo. Estos consistían en un conjunto de programas que definían y trabajaban sus propios datos. Los datos se almacenaban en archivos y los programas manejaban esos archivos para obtener la información. Si la estructura de los datos de los archivos cambiaba, todos los programas que los manejaban debían modificarse (*Por ejemplo, un programa que trabajase con un archivo de datos de alumnos, con una estructura o registro ya definido; si se incorporasen elementos o campos a la estructura del archivo, los programas que utilizan ese archivo tendrían que modificarse para tratar esos nuevos elementos.*)

Esto suponía un gran inconveniente a la hora de tratar grandes volúmenes de información. Surgió así la idea de separar los datos contenidos en los archivos de los programas que los manipulan, es decir, que se pueda modificar la estructura de los datos de los archivos sin que por ello se tengan que modificar los programas con los que trabajan. Es decir, estructurar y organizar los datos de forma que se pueda acceder a ellos con independencia de los programas que los gestionan. Y es así como surgen las bases de datos, que son gestionadas por los SGBD.

Hoy en día el mayor activo de las organizaciones son los datos y su gestión eficaz y segura. Por ello, si analizamos la mayoría de los ámbitos de actividad, nos encontramos que la utilización de las bases de datos está ampliamente extendida (al registrarse en una web, al acudir a la consulta médica, al consultar el catálogo de productos de una tienda online, etc.). Las bases de datos y los datos contenidos en ellas, son imprescindibles para llevar a cabo multitud de acciones.

Los SGBD más extendidos son los relacionales. Sin embargo, en los últimos años han aparecido otros tipos, tales como, los SGBDs orientados a objetos (PostgreSQL) o, en el mundo del Big data, sistemas de bases de datos noSQL, como por ejemplo MongoDB y Hadoop.

En el presente tema vamos a empezar definiendo lo que es una base de datos y posteriormente nos dedicaremos al estudio de los sistemas gestores de bases de datos, analizando sus principales funciones, componentes, arquitecturas y tipos de sistemas atendiendo a diversos criterios de clasificación.

2. SISTEMAS GESTORES DE BASES DE DATOS

Podemos definir una **base de datos** como un conjunto, colección o depósito de datos almacenados en un soporte informático no volátil. Los datos están interrelacionados y estructurados.

Un **sistema gestor de base de datos o SGBD** es una colección de programas de aplicación que permite a los usuarios la creación y el mantenimiento de una base de datos, facilitando la definición, construcción y manipulación de la información contenida en ésta.

En la actualidad debido al gran volumen de datos que manejamos se requiere de sistemas gestores de bases de datos robustos, que nos permitan acceder y gestionar los datos de un modo eficaz y eficiente.

Las principales **ventajas** del uso de SGBD son:

- Proporcionan al usuario una visión abstracta de los datos, ocultando parte de la complejidad relacionada con cómo se almacenan y mantienen los datos
- Disminuyen la redundancia y la inconsistencia de datos.
- Aseguran la integridad de los datos.
- Aumentan la seguridad y privacidad de los datos.
- Mejoran la eficiencia.
- Permiten compartir datos y accesos concurrentes.
- Incorporan mecanismos de copias de seguridad y recuperación de datos.

2.1. FUNCIONES

Un SGBD desarrolla tres funciones fundamentales como son las funciones de definición, manipulación y control de los datos.

A) FUNCIÓN DE DEFINICIÓN

Esta función debe permitir al diseñador de la BD especificar los elementos de los datos que la integran, sus estructuras, las relaciones entre ellos..., en definitiva permite describir y definir los esquemas de la base de datos.

Esta función se realiza mediante el **lenguaje de definición de datos (DDL)** propio de cada SGBD.

B) FUNCIÓN DE MANIPULACIÓN

Esta función se encarga de todas las operaciones de gestión de los datos de la base de datos. Se realiza mediante el **lenguaje de manipulación de datos (DML)**, permitiendo a los usuarios de la base de datos consultar o actualizar los datos de la misma, de acuerdo con las especificaciones y las normas de seguridad dictadas por el administrador.

Para recuperar la información que necesitamos se realizan consultas a la BD que pueden ser consultas a la totalidad de los datos o consultas selectivas con un cierto criterio de selección.

Para actualizar nuestra base de datos podemos realizar tres tipos de operaciones según sean nuestras necesidades que son inserción de datos, modificación o borrado de datos.

C) FUNCIÓN DE CONTROL

Esta función se realiza mediante el **lenguaje de control de datos (DCL)**, permitiendo al administrador de la base de datos establecer mecanismos de protección de las diferentes visiones de los datos asociadas a cada usuario, proporcionando elementos de creación y modificación de dichos usuarios. Adicionalmente, incorpora sistemas para la creación de copias de seguridad, carga de ficheros, auditoria, protección de ataques, configuración del sistema...

2.2. COMPONENTES

Los componentes en los que se divide todo SGBD son: **lenguajes** de la base de datos, **diccionario de datos**, **gestor** de la base de datos, **usuarios** de la base de datos y **herramientas** de la base de datos.

2.2.1. LENGUAJES DE LA BASE DE DATOS

A través de los lenguajes se pueden especificar los datos que componen la base de datos, su estructura, relaciones, reglas de integridad, características físicas y vistas externas de los usuarios.

Los lenguajes del SGBD son:

- Lenguaje de Definición de los Datos (**DDL**)

Habitualmente, los DDL suelen ser lenguajes muy simples con una gramática muy sencilla que permiten describir los datos con facilidad y precisión y sin necesidad de apoyarse en ningún lenguaje de programación. La representación de los datos obtenida en este proceso de compilación es almacenada en otro componente del SGBD denominado Diccionario de Datos.

- Lenguaje de Control de Datos (**DCL**)

El lenguaje de Control de Datos (DCL) es un sublenguaje del DDL encargado del control y seguridad de los datos y permite el control del acceso a la información almacenada en el diccionario de datos, así como el control de seguridad de los datos.

- Lenguaje de Manipulación de Datos (**DML**)

Es el encargado de gestionar la información de la BD. Gracias a él podemos recuperar información de la BD (consultas) o manipular datos (añadir, modificar y eliminar registros). Se trata también de un lenguaje basado en una gramática sencilla y fácil de entender. Además, los SGBD son capaces de procesar peticiones de DML que se han formulado desde programas escritos en otros lenguajes de programación.

2.2.2. DICCIONARIO DE DATOS

Es el lugar donde se almacena la información sobre la totalidad de los datos que forman la base de datos. Se trata de una metabase de datos; es decir, una base de datos que contiene información sobre otra base de datos. Contiene las características lógicas de las estructuras que almacenan los datos, su nombre, descripción, contenido y organización. Además, en él están contenidas, entre otras cosas, las restricciones de privacidad y acceso a los datos que han sido definidas en DDL y/o DCL.

2.2.3. GESTOR DE LA BASE DE DATOS

Este componente es uno de los componentes más complejos del SGBD. Es el encargado de proporcionar una interfaz entre los datos almacenados y los programas de aplicación que los manejan. Puede considerarse un intérprete entre el usuario y los datos.

El gestor de la base de datos se encargará de:

- Garantizar la privacidad, seguridad e integridad de los datos.
- Controlar los accesos concurrentes a la BD.
- Interactuar con el sistema operativo.

2.2.4. USUARIOS DE LA BASE DE DATOS

En los SGBD existen diferentes perfiles de usuario. Generalmente existen:

- **Administrador de la base de datos:** encargado de la función de administración de la BD. Va a tener las responsabilidades de definición, administración, seguridad, privacidad e integridad de la información utilizada.
- **Usuarios de la base de datos:** pueden existir diferentes usuarios con diferentes accesos y privilegios sobre los datos:
 - o Usuarios **técnicos** (diseñadores, operadores y personal de mantenimiento, analistas y programadores de aplicación) que son profesionales informáticos cuyo objetivo es desarrollar los programas de aplicación que posteriormente utilizarán los usuarios finales de la BD.
 - o Usuarios **finales** que son los que a través de programas de aplicación interactúan con la BD. Estos suelen ser usuarios no especializados.

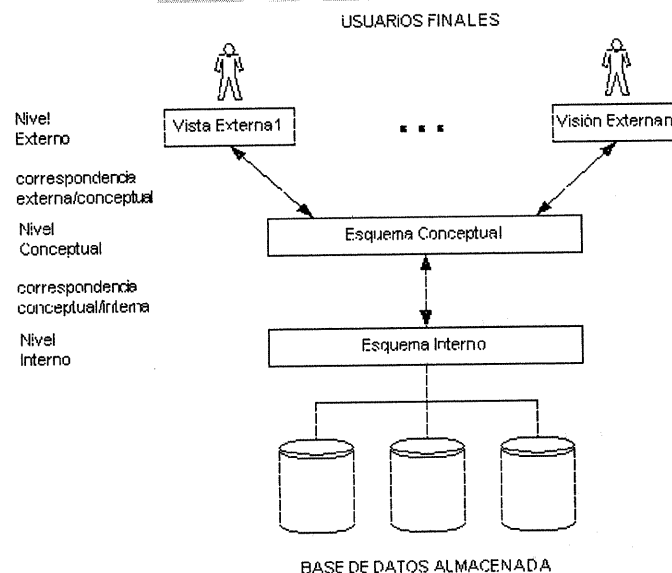
2.2.5. HERRAMIENTAS DE LA BASE DE DATOS

Las herramientas de la base de datos son un conjunto de aplicaciones que permiten a los administradores la gestión de la base de datos, controlar la privacidad de los datos (gestión de usuarios y permisos), generadores de formularios, informes, interfaces gráficas, etc. Facilitan el trabajo a los usuarios ya que suelen tener un interfaz fácil de entender y suelen basarse en menús que guían al usuario para conseguir el objetivo final.

2.3. ARQUITECTURAS DE REFERENCIA Y OPERACIONALES

2.3.1. ARQUITECTURA DE REFERENCIA

En 1975, el comité ANSI-SPARC propuso una arquitectura de tres niveles de abstracción para los SGBD cuyo objetivo principal era separar los programas de aplicación de la base de datos física: nivel interno o físico, nivel lógico o conceptual y nivel externo o de visión del usuario.



a) Nivel interno o físico

Este es el nivel más bajo de abstracción y asociado a él se encuentra el esquema interno. En él se describe la estructura física de la base de datos a través de un esquema que detalla el sistema de almacenamiento de la base de datos y sus métodos de acceso. Es el nivel más cercano al almacenamiento físico. Los usuarios que trabajan a este nivel son los **diseñadores de la base de datos** o **los Administradores**.

b) Nivel lógico o conceptual

Se describe con el esquema conceptual o esquema de la BD. Se obtiene a partir de los requerimientos de los usuarios potenciales del sistema de BD a implantar así que en este nivel se describe la estructura completa de la base de datos a través de un esquema que detalla las entidades, atributos, relaciones, operaciones de los usuarios y restricciones. A este nivel los usuarios que intervienen son los **programadores**, encargados de crear las estructuras lógicas necesarias para guardar la información.

c) Nivel externo o de visión del usuario

En este nivel se describen las diferentes vistas que los usuarios percibirán de la base de datos. Cada tipo de usuario o grupo de ellos verá sólo la parte de la base de datos que le interesa, ocultando el resto.

2.3.2. ARQUITECTURA OPERACIONAL

Existen diversas arquitecturas, que varían notablemente en base al tipo de SGBD utilizado, tanto en función del número de usuarios del sistema, como de la localización de la información y del modelo de SGBD:

a) Arquitectura centralizada:

Un sistema de BD centralizado es aquel que se ejecuta en un único sistema computacional sin tener que interactuar con otros computadores. Pueden ser desde sistemas monousuario hasta sistemas de BD que se ejecutan en sistemas de alto rendimiento.

Los sistemas monousuario tienen grandes déficits en comparación con los multiusuario, por ejemplo no tienen control de recurrencia y no suelen tener sistemas de recuperación.

b) Arquitectura Cliente-Servidor:

En esta arquitectura tenemos el lado cliente cuya funcionalidad será la parte visible al usuario de la base de datos como son formularios, gestión de informes, etc y, por otro lado, el lado servidor donde tendremos lo relacionado con las estructuras de datos, consultas, control de concurrencia y recuperación.

Hay dos tipos de sistemas servidores, los servidores transaccionales y los servidores de datos siendo éstos últimos los que ofrecen muchas más funcionalidades.

c) Arquitectura paralela:

Estos sistemas constan de varios procesadores y varios discos conectados a través de una red de interconexión de alta velocidad. Este tipo de SGBD tiene beneficios en velocidad (menos tiempo de ejecución para una tarea dada) y en ampliabilidad (capacidad de procesar tareas más largas en el mismo tiempo). Los modelos de este tipo de arquitectura más utilizados son el de memoria compartida, disco compartido, sin compartimiento y jerárquico.

d) Arquitectura distribuida:

La BD se almacena en varios computadores que se pueden comunicar por distintos medios de comunicación. No comparten memoria ni discos y sus tamaños y funciones pueden variar dependiendo del sistema que abarquen. Cada computador de un sistema distribuido se denomina nodo y normalmente se encuentran en lugares geográficos distintos.

2.4. TIPOS DE SISTEMAS

Los SGBD se pueden clasificar atendiendo a diversos criterios como son:

- Según el modelo lógico en el que se basan.
- Según el número de usuarios a los que da servicio.
- Según el número de sitios en el que está distribuida.

2.4.1. SEGÚN EL MODELO LÓGICO

A. MODELO JERÁRQUICO

El modelo jerárquico organiza la información en una jerarquía en la que la relación entre las entidades de este modelo siempre es del tipo padre/hijo.

De tal manera que existen nodos que contienen atributos y que se relacionarán con sus nodos hijos, pudiendo tener cada nodo más de un hijo, pero un nodo siempre tendrá un sólo padre.

La forma visual de este modelo es de árbol invertido, en la parte superior están los padres y en la inferior los hijos. Hoy en día el modelo jerárquico está en desuso.



B. MODELO EN RED

El modelo en red organiza la información en registro (se representan por nodos) y enlaces (se representan por arcos o lazos). En los registros se almacenan los datos, mientras que los enlaces permiten relacionar estos datos. El modelo en red aparece a mediados de los sesenta como respuesta a limitaciones del modelo jerárquico en cuanto a representación de relaciones más complejas. Una red es una estructura más general que una jerarquía, porque una ocurrencia de registro específica puede tener cualquier número de superiores inmediatos, es decir, no está limitada a un máximo de uno, como ocurre en el modelo jerárquico.

C. MODELO RELACIONAL

Este modelo es posterior a los dos anteriores y fue desarrollado por Codd en la década de los 70. Hoy en día este modelo es el más utilizado. El modelo relacional utiliza tablas bidimensionales (relaciones) para la representación lógica de los datos y las relaciones entre ellos. Cada tabla (relación) posee un nombre que es único y contiene un conjunto de columnas. Se llama registro o tupla a cada fila de una tabla, y campo o atributo a cada columna de la tabla. Las tablas deben cumplir las siguientes condiciones:

- Todos los registros de una tabla son del mismo tipo.
- En ninguna tabla aparecen campos repetidos ni registros duplicados.
- El orden de los registros en la tabla es indiferente.
- En cada tabla hay una llave (clave) formada por uno o varios campos.

D. MODELO ORIENTADO A OBJETOS

Con el auge de la programación orientada a objetos aparece el modelo orientado a objetos, que permite establecer relaciones entre objetos y atributos, en lugar de hacerlo entre campos individuales.

El objetivo de este modelo es cubrir las limitaciones del modelo relacional. Con este modelo se incorporan mejoras como la herencia entre tablas, los tipos definidos por el usuario, etc.

Aunque no han reemplazado a las bases de datos relacionales, si son el tipo de bases de datos que más está creciendo en los últimos años.

E. MODELO OBJETO-RELACIONAL

Algunos sistemas existentes en el mercado, basados en el modelo relacional, han sufrido evoluciones incorporando conceptos orientados a objetos. A estos modelos se les conoce como sistemas objeto-relacionales. La gran mayoría de las bases de datos relacionales clásicas de gran tamaño como Oracle, SQL Server, etc. son objeto-relacionales.

2.4.2. SEGÚN EL NÚMERO DE USUARIOS

- **Monousuario:** sólo atienden a un usuario a la vez
- **Multiusuario:** atienden a varios usuarios al mismo tiempo, la mayor parte de los SGBD son de este tipo.

2.4.3. SEGÚN SU UBICACIÓN

- **Centralizados:** el SGBD y la base de datos en sí residen por completo en un solo ordenador.
- **Distribuidos:** el SGBD y la base de datos pueden estar distribuidos en varios equipos conectados por una red.

Preparador Informática

3. SGBD COMERCIALES Y LIBRES

A continuación, se listan algunos de los SGBD comerciales y libres más utilizados:

SGBD comerciales:

Oracle, MySQL, Microsoft SQL Server, SYBASE, DB2, dBASE, Informix.

SGBD libres:

MySQL, PostgreSQL, Apache Derby, SQLite, Firebird, MariaDB

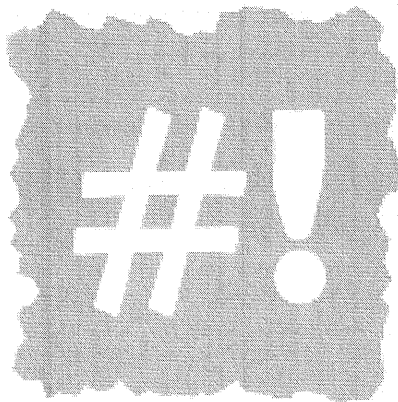
4. CONCLUSIÓN

La elección de un SGBD es una decisión muy importante a la hora de desarrollar proyectos. A veces, el sistema más avanzado o mejor según los expertos, puede no serlo para el tipo de proyecto que estemos desarrollando. Para ello, hay que tener en cuenta factores tales como el volumen de carga que debe soportar la base de datos, qué sistema operativo se utiliza como soporte, presupuesto del que se dispone, plataforma a la que va dirigido, etc.

5. BIBLIOGRAFÍA

- Date D.J.: **Introducción a los sistemas de bases de datos**. Editorial Addison-Wesley
- De Miguel A,y Piattini M:**Fundamentos y modelos de BBDD**. Edit. Ra-Ma
- Korth H. y Silberschatz: **Fundamentos de bases de datos**. Editorial McGraw-Hill
- <https://www.oracle.com/>
- <https://www.mysql.com/>
- <https://www.mongodb.com/>
- Etc.

Preparador Informática



Preparador Informática

