**SISTEMAS GESTORES DE BASES DE DATOS**

2.1.-Sistemas Gestores de Bases de Datos

La base de datos, como depósito único de los datos de toda la organización, debe ser capaz de atender las necesidades de los distintos tipos de usuarios que interactúan con ella. Puede definirse un **Sistema Gestor de la Base de Datos (SGBD)** como: ***“Un conjunto coordinado de programas, procedimientos, lenguajes, etc., que suministra, tanto a los usuarios no informáticos como a los analistas, programadores o al administrador, los medios necesarios para describir, recuperar y manipular los datos almacenados en la base, manteniendo su seguridad”.***

Debido a la diversidad usuarios con necesidades cambiantes a lo largo del tiempo, es imprescindible dotar al sistema de la adecuada flexibilidad para que pueda atender las exigencias de todos los usuarios y para que sea capaz de responder a los cambios a un coste no excesivo, es decir, el SGBD debe ser diseñado de forma tal que optimice las ventajas que se han indicado para una base de datos.

2.1.1- Funciones del Sistema Gestor de Bases de Datos.-

Las funciones esenciales de un SGBDson:

**Función de descripción o de definición**. Esta función debe permitir al administrador de la base especificar los elementos de datos que la integran , su estructura, las relaciones que existen entre ellos, las reglas de integridad semántica, los controles a efectuar antes de autorizar el acceso a la base, etc. Esta función se lleva a cabo mediante el Lenguaje de Descripción o de Definición de Datos (LDD) propio de cada SGBD y debe suministrar los medios para definir las tres estructuras de datos – externa, lógica global e interna -, especificando las características de los datos a cada uno de estos niveles.

**Función de manipulación.** Permite a los usuarios de la Base buscar, añadir, suprimir o modificar los datos de la misma, siempre de acuerdo con las especificaciones y las normas de seguridad establecidas por el administrador. Esta función se realiza mediante el Lenguaje de Manipulación de Datos (LMD) que facilita los instrumentos necesarios para la realización de estas tareas.

**Función de utilización.** Reúne todas las interfaces que necesitan los diferentes usuarios para comunicarse con la base y proporciona un conjunto de procedimientos para el administrador entre los que se encuentra el Lenguaje de Control de Datos (LCD). Además, en la mayoría de los SGBD existentes en el mercado existen funciones de servicio, como cambiar la capacidad de los ficheros, obtener estadísticas de utilización, cargar archivos, etc., y otras relacionadas con la seguridad física - copias de seguridad, rearranque en caso de caída del sistema, etc. – y protección frente a accesos no autorizados.

En resumen, en el cuadro adjunto se presentan las funciones esenciales de un Sistema Gestor de la Base de Datos (SGBD):

**DESCRIPCIÓN**

***Permite describir:***

Los elementos de la base de datos con

1. Su estructura
2. Sus Interrelaciones
3. Sus Validaciones

***A tres niveles***

1. Externo
2. Lógico Global
3. Interno

***Mediante un Lenguaje de Definición de Datos (LDD)***

**MANIPULACIÓN**

***Permite***

1. Buscar
2. Añadir
3. Suprimir
4. Modificar

***Datos de la base***

***Mediante un Lenguaje de Manipulación de Datos (LMD)***

***Lo cual supone***

1. Definir un criterio de selección (responsabilidad del usuario)
2. Definir la estructura externa a recuperar (responsabilidad del usuario)
3. Acceder a la estructura física (responsabilidad del sistema)

**CONTROL**

1. Reúne las Interfaces de los Usuarios
2. Suministra procedimientos para el Administrador

***Mediante un Lenguaje de Control de Datos (LCD)***

Las interrelaciones existentes entre estas funciones, el conjunto de los datos y los usuarios se encuentran representados en la figura adjunta.

Usuarios Remotos

Monitor de Teleproceso

Función de Utilización

Función de Descripción

Función de Manipulación

**SGBD**

Base de Datos

Usuarios Locales



2.2.- Arquitectura de los SGBD. Estandarización.-

Desde comienzos de los años setenta diversos grupos informáticos se han ocupado del tema de la estandarización de las bases de datos (ISO, INRIA, GESC, BSI, Codasyl, ANSI, etc) con el fin de conseguir que, una vez desarrollado un sistema e instrumentado en un determinado SGBD, el cambio de éste a otro producto comercial no implique tener que diseñar de nuevo la base de datos, ni tampoco que los programas que acceden a dicha base de datos tengan que ser reescritos. La estandarización ha de ofrecer también la oportunidad de adquirir distintos componentes de un SGBD (lenguajes, diccionarios, etc.) a diferentes suministradores.

La arquitectura a tres niveles (interno, conceptual y externo) definidos anteriormente, establecidos por el grupo ANSI/X3/SPARC marca la línea de investigación fundamental para la normalización y estandarización de los SGBD.

Esta arquitectura triesquemática de ANSI está parcialmente basada en el concepto de máquinas anidadas (llamadas de ***tipo cebolla***). El flujo de datos pasa a través de las distintas capas que están separadas por interfaces que tienden a aislar los diversos componentes del sistema para conseguir el objetivo de independencia.

En las arquitecturas se distinguen dos partes:

**\*.- Definición de la Base de Datos.**- La parte de definición se facilita por medio de una serie de funciones de programa e interfaces, que dan lugar a un conjunto de datos llamados **metadatos** (datos acerca de los datos) que se almacenan en el **diccionario** (*catálogo* o *metabase* en los sistemas relacionales), que es el eje principal de la arquitectura alrededor del cual giran los demás elementos.

Una base de datos se define especificando primeramente el **esquema conceptual** mediante un *lenguaje de definición del esquema conceptual* o una herramienta CASE integrada. Este esquema conceptual es compilado por el procesador del esquema conceptual y se almacena en el **catálogo de datos.**

**El procesador del esquema conceptual** muestra, por ejemplo, mediante **menús**, la información del esquema conceptual, mediante dicha información pueden definirse los **esquemas externo e interno** a través de distintas interfaces. Estos esquemas, claramente diferenciado, llevan a considerar la existencia de tres tipos de administraciones: *Administración por parte de la Empresa, Administración de la Base de Datos y Administración de Aplicaciones*

**\*.- Manipulación de la Base de Datos.-** El usuario puede manipular (insertar, borrar, modificar y recuperar) los datos utilizando un interfaz que puede ser un *lenguaje de manipulación de datos*, por ejemplo SQL. Una petición de datos por parte del usuario se ejecuta por los **transformadores externo / conceptual, conceptual / interno e interno / almacenado** que utilizan los metadatos devolviendo el resultado al usuario.

Estos transformadores constituyen la **función de vinculación** entre los distintos niveles (conceptual, interno y externo).

La arquitectura a tres niveles de ANSI responde por tanto positivamente a las exigencias de independencia, flexibilidad y capacidad de evolución propuestas en la estandarización.

La consecución de trabajos del grupo ANSI culminó en marzo de 1986 al presentar el **Modelo de referencia para la estandarización de los SGBD.**

Se define como **Modelo de Referencia** a una estructura conceptual para la estandarización de los SGBD, en la que se identifican una serie de componentes y se establece cómo interrelacionan entre ellos. Los objetivos a cumplir por este Modelo de Referencia son:

Establecer un ***Marco común*** para la descripción de los SGBD

Lograr una ***Estandarización*** para impulsar la compatibilidad de los distintos componentes de los SGBD facilitando su comparación y evaluación

El modelo de referencia no es en sí un estándar pero sienta las bases para futuras estandarizaciones ya que se contempla desde tres puntos de vista distintos:

El de los **componentes** que integran un SGBD

El de las **funciones** que se deben especificar

El de los **datos que se deben** describir y utilizar

El **Modelo de Referencia** está basado en la arquitectura ANSI revisado, en aras de una simplificación, ocupándose del qué, por qué y para qué peno no del cómo. Es decir, su objetivo es describir las interrelaciones del SGBD pero no indicar nada de su instrumentación.

En particular, recomienda que todos los datos relacionados con el control centralizado de la Base de Datos (reglas de integridad y de seguridad) se encuentren en la metabase (***catálogo***) y no se dejen en manos de los usuarios ya sean éstos finales o programadores.

Recomienda así mismo la existencia de ***Herramientas de Gestión de Datos*** o componentes software, como lenguajes de cuarta generación, soporte de ayuda a la decisión, facilidades para realizar el ajuste (*tuning*), utilidades para el volcado de ficheros, sistemas de diccionario de datos, etc.

En el Modelo de Referencia se distingue un **Sistema de Control de Transformación de Datos,**  que es el *núcleo o kernel* del SGBD, que provee de operadores para la descripción y manipulación de los datos, y dos tipos de interfaces:

**Interfaz de Lenguaje de Datos** que permite a los usuarios y a los procesadores especificar sus peticiones para la recuperación y actualización de los datos por parte del SGBD.

**Interfaz de Lenguaje de Datos Interno** que permite el uso de los procesadores que soportan el funcionamiento de los SGBD, en particular los del SO.

2.3.- Componentes de un Sistema Gestor de Base de Datos.-

Puede presentarse de forma gráfica la estructura general de un SGBD donde, además del **núcleo del sistema**, existe un conjunto de herramientas y facilidades.

**Herramientas y facilidades**

**Acceso a Datos**

**Diccionario de Datos**

**Núcleo**

**Sistema Operativo**

**Datos**

El **núcleo** del SGBD esta en mayor o menor medida soportado por el **Sistema Operativo**; le corresponde transformar las instrucciones de petición de datos que le llegan en órdenes que el Sistema Operativo sea capaz de de entender y gestionar. Los servidores de bases de datos poseen módulos, facilidades de usuario que realizan llamadas o funciones de Sistema Operativo, de modo que no haya que cerrar las aplicaciones para realizar las acciones típicas del S.O.

Sobre el núcleo se sitúa el **diccionario de datos** (también llamado **catálogo** o **metabase**) ya que a este nivel se produce la transformación de las solicitudes de losclientes en instrucciones inteligibles para el servidor.

El conjunto de herramientas y facilidades permiten el acceso a los datos, ya sea directamente (**facilidades de usuario**) o mediante aplicaciones desarrolladas por informáticos con la ayuda de precompiladores, generadores de aplicaciones, etc.

Otros componentes (como las **utilidades** y el **exportador/importador**) facilitan la tarea del administrador o ayudan a realizar el diseño de la base de datos (herramientas CASE).

Existen productos comerciales que ofrecen una gran portabilidad, al disponer de interfaces para la mayoría de los SGBD con penetración en el mercado.

**FACILIDADES DE USUARIO**

**GENERADOR DE APLICACIONES (L4G)**

**EXPORTADOR/**

**IMPORTADOR**

**UTILIDADES**

**DEL SGBD**

**GENERADORES DE INFORMES**

**PRECOMPILADOR**

**LENGUAJES**

**AYUDAS AL DISEÑO (CASE)**

**APLICACIONES DESARROLLADAS**

**DICCIONARIO DE RECURSOS DE LA INFORMACIÓN**

**NÚCLEO DEL SGBD**

**LENGUAJE DEL SGBD**

**DATOS**

**SISTEMA OPERATIVO**

**(Catálogo)**

2.3.1.- Diccionario de datos: Concepto y estructura

Para el control de los objetos de las bases de datos se definen en éstas dos elementos que pueden aparecer asociados o no: el ***diccionario de datos*** y el ***directorio de datos*** En el uso coloquial de estos términos se ha introducido el concepto de ***catálogo*** de forma que suelen considerarse los tres términos como sinónimos aunque se refieren a conceptos muy diferentes.

El ***diccionario de datos*** contiene información los datos almacenados. Es lo que se llama una *metabase* ya que contiene información sobre los elementos que conforman cada una de las bases de datos, información para la manipulación: tablas, registros, campos, relaciones y sus estructuras, contenidos, descripciones, significado, etc. de modo que siendo esta información legible por los usuarios, puedan éstos hacerse una buena idea sobre la estructuración de los contenidos de información almacenados en cada base de datos. Posee información sobre posniveles de seguridad implementados para la definición de datos.

A los contenidos del diccionario de datos se les denomina ***metadatos.*** Corresponde al administrador de la base de datos la responsabilidad de la creación y el mantenimiento del diccionario de datos.

Los contenidos del diccionario de datos son:

\*.- Los ***esquemas externo y conceptual de la base de datos.*** Esto es, cada uno de los elementos que los forman

\*.- La ***información sobre tablas, consultas, campos, registros y referencias cruzadas entre registros de varias tablas.*** Se guarda información sobre las propiedades de esos elementos: nombre, tipo, longitud, representación, descripción de los registros, reglas de validación, etc. Es decir, la descripción de los datos y las estructuras.

\*.- Los ***niveles de autorización y derechos de acceso***  de cada uno de los usuarios de la base de datos.

\*.- Los ***controles de seguridad de acceso a datos*** - gestión de usuarios -e ***integridad.***

**Directorio de Datos**

**Diccionario de Datos**

**Repositorio de Datos**

\*.- ***Sinónimos*** y palabras polisémicas con sus descripciones.

\*.- ***Aplicaciones*** que se relacionan con los datos de cada base de datos

2.3.2.- El Directorio de datos

El **Directorio de datos** contiene información sobre el esquema interno, ya que se encarga de que el Sistema Operativo sea capaz de reconocer las órdenes de la base de datos, por lo que tanto su contenido como su estructura están enfocados al sistema, de modo que transforma los elementos del esquema externo al esquema interno para facilitar su funcionamiento. De forma restrictiva, el concepto de ***directorio*** se corresponde con el de ***catálogo***.

A veces, los conceptos relativos a diccionario y directorio de datos se agrupan en un solo elemento que posee la funcionalidad de ambos. A este conjunto se le llama ***diccionario***  o, mas generalmente, ***catálogo***, según el fabricante que lo implemente.

2.3.3.- El Repositorio de datos

El ***repositorio de datos*** es un diccionario de datos, diferente del anterior, gestionado por una herramienta CASE, que almacena información textual y gráfica sobre las características de un determinado objeto.

Poseen funcionalidades de directorio de datos y de diccionario de datos ya que en este caso almacena información de tipo texto y gráficos que es recuperable por el usuario. Los repositorios, que se emplean en la fase de análisis (descripción lógica de los datos), facilitan la labor posterior de descripción e implementación de las características allí definidas.

El diccionario/directorio/catálogo de datos almacena la información del mismo modo que si fuese una base de datos mas del sistema, empleándose las mismas órdenes que para el resto de las bases de datos.

Desde el punto de vista del funcionamiento del diccionario de datos hay que destacar los siguientes componentes software, transparentes para el usuario:

\*.- Un ***administrador de diccionario*** que se relaciona con el Lenguaje de Definición de Datos (DDL) encargado de procesar las peticiones al diccionario y almacenar en una tabla las operaciones realizadas.

\*.- Un ***procesador de consulta*** que se encarga de procesar las peticiones de los usuarios.

\*.- Un ***generador de informes*** y un traductor del Lenguaje de Manipulación de Datos (DML) que permite a los usuarios modificar el contenido del diccionario.

2.4.- Interacción del Sistema Gestor de Bases de Datos con el Sistema Operativo

El SGBD constituye un subsistema del sistema informático y, en particular, es un subsistema del software. Su funcionamiento, por tanto, estará muy interrelacionado con el de otros componentes del software y especialmente con el Sistema Operativo.

Aunque no es posible un estudio pormenorizado de dicho funcionamiento debido a la diversidad de SGBD y la diversidad del equipo físico en el que se apoya, si puede obtenerse una visión general analizando aquellos aspectos comunes a la mayoría de los SGBD actualmente operativos.

La diferencia entre el modo de acceso a un fichero y a una base de datos se centra en que, en el primer caso, el programa de aplicación accede al fichero por medio del subsistema de gestión de datos del Sistema Operativo, que es quien contiene los métodos de acceso.

Sin embargo, cuando se trata de una base de datos el programa de aplicación (que incluye en su lenguaje anfitrión el LMD embebido) se dirige al SGBD, el cual, a través del Sistema Operativo, accede a la base de datos.

La interacción, en un entorno concurrente, entre el SGBD, el Sistema Operativo y los Programas de Aplicación se muestra en la figura adjunta.

AC n

VE n

ATU n

**Programa de Aplicación**

UE n

**BASE DE DATOS**

SISTEMA OPERATIVO

VE2

VE1

E.L.G.

**SISTEMA DE GESTIÓN DE LA BASE DE DATOS (SGBD)**

4

5

6

**ALMACENAMIENTO INTERMEDIO**

7

Area de Comunicación

AC1

Nombre de la vista externa

VE1

Area de Trabajo del Usuario ATU1

**Programa de Aplicación**

Unidad de Ejecución 1 UE1

12

9

1

2

10

11

8

3

Por cada Programa de Aplicación (PA) que se está ejecutando, existe una Unidad de Ejecución (UE) donde se encuentra el Área de Trabajo del Usuario (ATU) con sus Áreas de Entrada y Salida (E/S) y un Área de Comunicación con el SGBD (AC) destinada a recibir los mensajes y la información de control procedente del SGBD. Desde el programa de aplicación se hace referencia a la Vista Externa (VE) permitida a tal programa. En la biblioteca del sistema se encuentran almacenados, además de los datos, la estructura lógica global y la estructura interna, así como las vistas externas que serán llamadas por los programas de aplicación de los usuarios.

El flujo de datos e instrucciones entre estos elementos es el siguiente:

1º.- Se produce una llamada desde una unidad de ejecución al SGBD (flecha 1); en la llamada se ha de hacer referencia a la vista externa implicada (flecha 2)

2º.- El SGBD analiza la llamada y completa los argumentos con la información de la vista externa a la que se ha hecho referencia en la llamada, así como con la información correspondiente a la estructura lógica global y la estructura interna con ella relacionada; esta información se encuentra previamente almacenada en los ficheros del sistema , desde donde pasa al SGBD (flechas 3 y 4).

3º.- Una vez comprobado el derecho del Programa de Aplicación (PA) a utilizar esta vista, y después de verificar su corrección, el SGBD traduce la llamada en las correspondientes órdenes para los métodos de acceso del Sistema Operativo (flecha 5).

4º.- El Sistema Operativo accede al soporte secundario (disco) donde se encuentran los datos (flecha 6)

5º.- Los datos a recuperar pasan del soporte donde se encuentra almacenada la base de datos al área de almacenamiento intermedio (buffers), y, si se tratase de una inserción o modificación pasarían en sentido contrario (flecha 7).

6º.- Los datos son transferidos desde el área de almacenamiento intermedio al área de trabajo del usuario de la unidad de ejecución desde donde se hizo la llamada (flecha 8) [o en sentido contrario si se hizo una inserción o una modificación], realizándose las correspondientes transformaciones entre las representaciones de los datos.

7º.- El SGBD, una vez terminada la operación de manipulación pasa al área de comunicación los indicadores de estado (flecha 9), en éstos se señala si la operación ha acabado satisfactoriamente o no, al tiempo que se dan otras informaciones sobre la operación realizada.

8º.- El Programa de Aplicación revisa el estado de los indicadores, que se encuentran en el área de control de la unidad de ejecución desde la que se efectuó la llamada, y toma las decisiones oportunas (flecha 10).

9º.- Los datos, que se encuentran en el área de E/S de la correspondiente unidad de ejecución, en el caso de que la operación haya terminado satisfactoriamente, ya pueden ser utilizados por el Programa de Aplicación (flecha 11).

2.5.- Interacción del Sistema Gestor de Bases de Datos con el Usuario: Lenguajes.-

Debido a las distintas funciones a las distintas funciones a realizar por el SGBD se hace necesario disponer de diferentes lenguajes y procedimientos que permitan la comunicación con la base de datos, tanto dirigidos a las funciones indicadas (definición, manipulación o control) como dirigidos a los diferentes tipos de usuarios a de procesos a realizar.

La tipología de los lenguajes de un SGBD se expresa en el cuadro adjunto:

Definición

Manipulación

**Por tipo de función**

**Por tipos de Usuarios y de Aplicaciones**

Informáticos

Finales

Aplicaciones formalizables

Aplicaciones no formalizables

Como se ve, las distintas características del proceso y del usuario determinan el tipo de lenguaje a realizar. En general, los usuarios informáticos, como el diseñador de la base, el administrador, analistas, programadores, etc., requerirán medios potentes y flexibles con los cuales consigan definir, administrar, extraer o manipular los datos de la base. Normalmente se apoyarán en un lenguaje de programación que están habituados a manejar (“***Lenguaje Anfitrión***”), para lo cual deberá permitir hacer llamadas desde un programa de aplicación al SGBD.

El conjunto de sentencias de manipulación del SGBD que pueden ser llamadas desde un lenguaje de programación permitiendo el acceso a la base de datos, se suele denominar ***sublenguaje de datos*** o también ***lenguaje huésped*** o  ***lenguaje embebido***.

Los SGBD admiten , en general, varios lenguajes de tipo anfitrión para manipular datos (Cobol, Ensamblador, Fortran, PL/I, Basic, Pascal, C, etc,.) . Así mismo, la práctica totalidad de los SGBD admiten lenguajes de 4ª generación que permiten el acceso a la base de datos, mediante sentencias embebidas en dicho lenguaje y escritas en un lenguaje de datos como SQL.

El usuario final, por su parte, requerirá medios simples para comunicarse con la base, lo que puede conseguirse mediante un lenguaje de manipulación autocontenido, que tenga una sintaxis sencilla, pero potente como para soportar demandas de información muy variadas o por medio de tratamientos parametrizados que suelen presentarse al usuario en forma de *menús*.

La estructura y la sintaxis de estos tipos de lenguajes dependen de cada SGBD. Para modelo de datos en red, las normas Codasyl proponen especificaciones concretas de la sintaxis para los lenguajes de descripción y manipulación de los datos. Para modelos de datos relacionales el SQL es un estándar muy extendido que proporciona estas facilidades.

2.5.1.- Lenguajes de Definición de Datos

Los instrumentos que permiten al administrador de la BD describir los datos con facilidad y precisión, especificando sus distintas estructuras es lo que se denomina *Lenguaje de Definición de Datos (LDD)* . Suelen ser lenguajes autocontenidos y no necesitan apoyarse en ningún lenguaje de programación. El SGBD deberá facilitar los medios para describir la estructura lógica global, para hacer las especificaciones relativas a la estructura interna y para declarar las estructuras externas que sean requeridas para el desarrollo de distintas aplicaciones.

2.5.1.1.- Lenguajes de definición de la estructura lógica global

Desde el punto de vista lógico global el administrador debe disponer de un instrumento de descripción que permita asignar nombres a los campos, a los agregados de datos, a los registros, etc. estableciendo sus longitudes y sus características así como las relaciones entre estos elementos, especificar los identificadores e indicar restricciones semánticas que se han de aplicar a los diferentes objetos descritos.

2.5.1.2.- Lenguajes de definición de la estructura lógica interna

En teoría, el propio SGBD debería conseguir automáticamente la optimización del almacenamiento y recuperación de los datos y encargarse, a partir de la estructura lógica global, de definir la estructura interna adecuada sin intervención del usuario (administrador).

Para ello, habría que suministrar al SGBD las informaciones precisas sobre volúmenes, crecimiento previsto, tipos de registros mas accedidos, con indicaciones del número medio de accesos, relación entre actualizaciones y consultas, etc.

En la práctica, puede mejorarse sensiblemente la eficiencia si el administrador especifica características respecto a la estructura física, por lo que deberá disponer de un lenguaje de definición de la estructura interna o, simplemente, deberá dar valores a ciertos parámetros.

En muchos SGBD se suministra automáticamente por defecto una estructura interna, que es la que el sistema considera mas adecuada para la estructura lógica global definida, aunque el administrador deberá ajustar posteriormente dicha estructura interna para conseguir una mayor eficiencia.

2.5.1.3.- Lenguajes de definición de las estructuras externas

El SGBD debe poner a disposición de los usuarios los medios necesarios para recuperar o actualizar los datos contenidos en la base de datos, de acuerdo con la visión lógica o estructura externa (vista) que precise cada aplicación.

Al definir una estructura externa es preciso darle un nombre e indicar qué datos y qué interrelaciones de la estructura lógica global se encontrarán en la misma. Cuando se desee utilizar un esquema externo ya definido se podrá hacer referencia al mismo invocando su nombre desde el lenguaje de manipulación.

2.5.2.- Lenguajes de manipulación de datos

Para cumplir los objetivos asignados a la función de manipulación debe disponerse de lenguajes que ofrezcan a los usuarios la posibilidad de referirse a determinados conjuntos de datos, que cumplan ciertas condiciones (*criterio de selección*) como que un atributo que tenga un determinado valor, o un conjunto de atributos y valores que satisfagan cierta expresión lógica. Además del criterio de selección, es preciso indicar la estructura externa que se desea actualizar o recuperar.

Una vez especificados el criterio de selección y los datos a actualizar o recuperar el SGBD debe ocuparse de acceder al correspondiente soporte físico de donde se extraerán los datos definidos para su transferencia a un dispositivo de salida , o, si se trata de una actualización, en donde se insertarán, modificarán o borrarán los datos.

Pero al igual que el programador precisa de un lenguaje de manipulación que se embeba en un lenguaje de programación, el usuario no informático deberá disponer de también de un instrumento análogo (mucho mas sencillo) que le permita comunicarse con la base y extraer de ella o introducir en ella las informaciones que precise. Para ello, los SGBD suelen disponer de lenguajes autocontenidos para que, desde un terminal y en modo interactivo, el usuario pueda acceder a la base y manipular los datos almacenados en ella sin necesidad de apoyarse en un lenguaje de programación.

La mayoría de los SGBD utilizan como lenguaje de manipulación de datos el Lenguaje Estructurado de Consultas o SQL.

Atendiendo a su utilización, los LMD pueden ser *procedimentales* (lenguajes en los que, además de *qué* se quiere es preciso indicar el algoritmo que establece *cómo* obtenerlo) o *no procedimental* (lenguaje en el que basta decir *qué* se quiere sin explicar *cómo* se obtiene)

Atendiendo a la forma en que se recuperan o actualizan datos, los LMD pueden ser *Navegacionales* (recuperando o actualizando los datos registro a registro) o *Especificacionales* (actuando sobre conjuntos de registros)

Por último, atendiendo al momento en que actúan sobre la BD, los LMD pueden ser *diferidos* (actuando en procesos por lotes) o *conversacionales* (actuando de modo interactivo con la BD).

2.6.- Modelos de Datos.-

En una primera aproximación puede decirse que un **Modelo de Datos (MD)** es un conjunto de conceptos que permiten describir, a distintos niveles de abstracción, la estructura de una base de datos, a la que se denomina **esquema**. Según el nivel de abstracción de la arquitectura ANSI a tres niveles en el que se encuentre la estructura descrita, el modelo que permite que su descripción será un modelo **externo**, **global** o **interno**.

Los ***modelos externos*** *permiten representar los datos que necesita cada usuario en particular con las estructuras propias del lenguaje de programación que va a emplear*.

Los ***modelos globales*** *ayudan a describir los datos para el conjunto de los usuarios de la Base de Datos.*

Los ***modelos internos***, llamados también ***modelos físicos***, están *orientados a la máquina, siendo sus elementos de descripción punteros, índices, agrupamientos*, etc.

**MODELOS DE DATOS**

**EXTERNO**

**\* Punto de vista de cada usuario en particular**

**GLOBAL**

**\* Punto de vista del conjunto de los usuarios (Empresa)**

**INTERNO**

**\* Punto de vista de la máquina**

Los modelos internos no están estandarizados ni existen en realidad como tales modelos sino que son propios de los productos comerciales. Por su parte, los modelos externos utilizan los mismos conceptos que los correspondientes modelos globales. Por lo que es en éstos últimos en los que se centrará la exposición.

Los **modelos globales** se clasifican en:

**CONCEPTUALES**

- Enfocados a describir el mundo real, con independencia de la máquina

**CONVENCIONALES O LÓGICOS**

- Implementados en el SGBD

**MODELO DE DATOS GLOBALES**

**Jerárquico**

**Codasyl**

**Relacional**

**Modelos conceptuales o modelos de Alto Nivel.-** Facilitan la descripción global del conjunto de la información, con independencia de la máquina, por lo que sus conceptos son cercanos al mundo real (entidades, atributos, interrelaciones, etc.). Son modelos de análisis, no de implementación.

**Modelos convencionales.-** Se encuentran soportados por el SGBD y están orientados a describir los datos a nivel lógico para el SGBD (suelen recibir también el nombre de ***modelo de Base de Datos***) por lo que sus conceptos son propios de cada sistema gestor (tablas y relaciones en el modelo relacional, árboles en el jerárquico, redes en el Codasyl, etc.)

El **Modelo de datos**, tanto lógico como físico, es el instrumento que se aplica a los datos para obtener el **esquema**.

**MUNDO REAL**

**ESTRUCTURA DE DATOS**

**(ESQUEMA)**

**MODELO**

**DE**

**DATOS**

Conviene distinguir entre **esquema** o descripción de la estructura de la base de datos, y **ocurrencia del esquema** o los datos que se encuentran almacenados en el esquema en un determinado momento. El esquema no varía mientras no varíe el mundo real que describe, en tanto que la ocurrencia del esquema, esto es, los datos contenidos en él, son distintos en el transcurso del tiempo.

Puede entonces definirse, de forma mas precisa, un **modelo de datos** como “*un conjunto de conceptos, reglas y convenciones que permiten describir y manipular (consultar y actualizar) los datos de un cierto mundo real que se desea almacenar en una base de datos*”.

Por lo que respecta a la relación existente entre los modelos y los lenguajes de datos, hay que destacar que los modelos son la base para los lenguajes, aunque el nivel de abstracción de éstos últimos es menor, ya que el lenguaje es el modelo mas una sintaxis. La existencia de distintos lenguajes puede proceder tanto del modelo como de la sintaxis; por ejemplo, el lenguaje SQL es el resultado de aplicar una determinada sintaxis al modelo relacional.

2.7.- Definición del modelo de datos

Las propiedades de un modelo de datos son de dos tipos: **Estáticas** o invariantes en el tiempo que responden a lo que se suele entender como estructura, y **Dinámicas**, que se corresponden con las operaciones que se aplican a los datos o valores almacenados en las estructuras, los cuales varían en el transcurso del tiempo al aplicárseles dichas operaciones.

2.7.1.- Estática del modelo

Está compuesta por dos tipos de elementos

a.- ***Elementos permitidos:***

No son los mismos para todos los modelos de datos, variando especialmente en su terminología, pero, en general son:

1º.- *Objetos* (entidades, relaciones, registros, etc.)

2º.- *Asociaciones entre objetos* (interrelaciones, etc.)

3º.- *Propiedades o características de los objetos o de las asociaciones* (atributos, campos, elementos de datos, etc.).

4º.- *Dominios*, conjuntos nominados de valores sobre los que se definen las propiedades.

La representación de estos elementos depende de cada modelo de datos, pudiendo ser en forma de ***grafos*** (modelo jerárquico o en red) en forma de ***tablas*** (modelo relacional) o en ambos (modelo Entidad/Interrelación)

b.- ***Elementos no permitidos o restricciones:***

Cada modelo de datos impone limitaciones a las estructuras que admite, bien por sí mismo, ***restricciones inherentes***, las impuestas por el universo que se está modelando, ***restricciones de integridad*** o ***semánticas***.

Las primeras varían de un modelo a otro y no permiten describir ciertas estructuras. Por el contrario, las restricciones de integridad son facilidades que se ofrecen al diseñador con el fin de pueda representar en el esquema, lo mas fielmente posible, la semántica de los datos.

2.7.2.- Dinámica del modelo

Los valores que toman los distintos objetos de un esquema en un momento determinado reciben el nombre de **ocurrencia** o **estado de la base de datos** en ese momento, deben cumplir las restricciones de integridad, al pasar de una ocurrencia a otra mediante una variación en la base de datos (alta, baja o modificación) así como las posibles restricciones asociadas al cambio de estado.

La componente dinámica del modelo consta de un conjunto de operadores que se definen sobre la estructura del correspondiente modelo de datos, ya que no todas las estructuras admiten el mismo tipo de operaciones. *La aplicación de una operación en una ocurrencia de un esquema transforma a ésta en otra ocurrencia.*

Una **operación** tiene dos componentes:

**Localización o enfoque (selección).-** Bien localiza una ocurrencia de un objeto indicando un *camino* o bien un conjunto de ocurrencias especificando una *condición*. En el primer caso se trata de un **sistema navegacional** y en el segundo se trata de un **sistema de especificación**.

**Acción.-** Tiene por fin realizar sobre la o las ocurrencias previamente localizadas mediante la operación anterior, una operación, que puede consistir en una *recuperación* o una *actualización* (inserción, borrado o modificación)

La distinción entre localización y acción es meramente formal, aunque algunos lenguajes como el LMD de Codasyl tiene dos mandatos distintos, uno para expresar la localización y otro para la acción. SQL sin embargo, reúne ambas operaciones en un único operador.

2.8.- Tipos de modelos de datos.-

Los sistemas Gestores e Bases de datos se clasifican en tres tipos de estructuras de datos diferentes en función de su estructura lógica:

**Modelo en Red**: Una estructura de datos en red también llamada **plex** se caracteriza por permitir la existencia de relaciones de “muchos a muchos” (M: N), de manera que cada nodo hijo pueda tener mas de un padre. Los modelos en red suelen adaptarse al estándar definido por la Conferencia sobre Lenguajes de Datos CODASYL.

La estructura de datos del modelo CODASYL utiliza los siguientes elementos básicos:

ELEMENTO.- Unidad de datos más pequeña a la que se puede hacer referencia. Debe tener un nombre y contiene un valor de algún tipo definido de datos (booleano, numérico, carácter, etc.)

AGREGADO DE DATOS.- Conjunto de datos colocados consecutivamente. Puede ser una matriz, una fecha, etc.

Tanto los Elementos como los Agregados de datos se corresponden con los campos de los ficheros clásicos.

REGISTRO o ARTICULO.- Colección nominada de elementos de datos. Es la unidad básica de acceso y manipulación de la base de datos y se corresponde con el concepto de registro de los ficheros clásicos.

CONJUNTO o SET.- Conjunto lógico de dos o más tipos de registros que establece una vinculación entre ellos. Normalmente está formado por un registro llamado de **tipo propietario** y uno o más registros llamados de **tipo miembro**. Es el elemento fundamental y característico del modelo CODASYL y el origen de muchas de las restricciones de este modelo.

AREA.- Subdivisión de la base de datos que contiene un conjunto de registros pertenecientes a uno o más Registros Tipo (SET).

Sistemas Gestores de Bases de Datos que se adaptan al Modelo CODASYL son el DMS y RDMS de Unisys, el IDMS de Culliname, el TOTAL de Cincom, el EDMS de Xerox, el PHOLAS de Philips, el DBOMP de IBM, el IDS de Honeywell, etc.

Un modelo en red que no se adapta al enfoque CODASYL es el IMAGE de Hewlett Packard.

**MODELO JERÁRQUICO:** Es un modelo diseñado especialmente para representar situaciones en las que predominan relaciones del tipo “uno a muchos” (1:N).

Es un modelo muy rígido con un gran fundamento matemático, desarrollado a partir de la práctica, a diferencia del modelo en red que se desarrolló con el fin de establecer estándares detallados.

Este modelo está dominado por el sistema IMS de IBM que lo utiliza como estándar. Otro sistema basado en el enfoque jerárquico es el SYSTEM 2000 de Intel.

El modelo jerárquico tiene una estructura de árbol invertido. Su estructura es pues arborescente compuesta por nodos, que representan las ENTIDADES, enlazados por ARCOS, que representan las ASOCIACIONES o INTERRELACIONES entre dichas entidades.

El nivel superior de esa estructura está ocupado por una única entidad bajo la cual se distribuyen el resto de las entidades. La estructura jerárquica es, pues, un caso particular del modelo en red, con fuertes restricciones adicionales derivadas de que las asociaciones del modelo jerárquico deben formar un **árbol ordenado**.

La terminología para describir las estructuras arborescentes es la siguiente:

RAÍZ.- El nodo mas alto de la jerarquía, La raíz siempre es única.

PADRE.- Aquel nodo que tiene descendientes, todos los cuales están situados al mismo nivel.

HIJO.- Nodo vinculado a otro u otros de nivel superior.

HOJAS.- Aquellos nodos que no tienen hijos.

CAMINO.- Líneas que unen nodos entre sí. Un camino que termina en una Hoja se denomina RAMA.

El recorrido de los distintos nodos de un árbol se realiza en el modelo jerárquico en **preorden**, recorrido que requiere secuencialmente tres pasos:

Visitar el nodo raíz

Visitar el Hijo Izquierdo

Visitar el Hijo Derecho

La diferencia fundamental entre el modelo en red y el modelo jerárquico se centra en que el primero permite a un registro de tipo hijo tener mas de un padre, mientras que el modelo jerárquico todo nodo tiene un único padre, salvo el raíz, aunque un nodo puede tener muchos hijos.

**modelo relacional:** Modelo diseñado por Codd en los años 70 que propone que todos los datos de la base estén representados en **tablas** constituidas por **filas** y **columnas**. A las tablas se les da el nombre de RELACIONES, de donde el modelo toma su nombre.

Para manejar los datos de las tablas se utilizan dos lenguajes: el **álgebra relacional** y el **cálculo relacional**. Ambos utilizan operadores lógicos para la manipulación de los datos.

El sistema relacional se caracteriza por:

La facilidad para su representación lógica, al presentarse sobre tablas separadas y no ofrecer ningún tipo de jerarquía.

La posibilidad de manipular las tablas con sentencias simples, sin necesidad de procesar registro a registro.

Poder acceder a datos de cualquier columna de la tabla o a filas de una o más relaciones con gran facilidad