

## TEMA 1

**Introducción a las Bases de Datos****1.1. Archivos tradicionales versus bases de datos.**

En cualquier organización, existe información que necesariamente ha de ser manejada por distintos departamentos o secciones. Así por ejemplo, en una empresa, los departamentos de contabilidad, compras y almacén, necesitarán para desarrollar su actividad procesar datos referentes a clientes, ventas y/o materiales. En un sistema basado en archivos tradicionales, cada uno de estos departamentos tendrá la información almacenada en ficheros de uso exclusivo, por lo que nos encontraremos tal y como muestra la figura 1.1 con tres archivos de materiales, dos de proveedores y dos más de clientes.

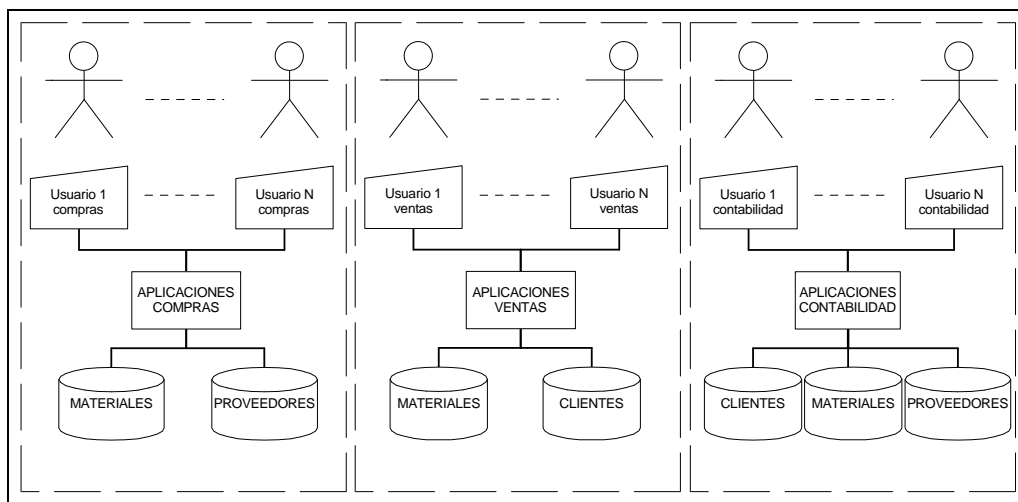


Figura 1.1.

Este esquema de almacenamiento puede llegar a plantear una serie de problemas de difícil solución:

- El primero de ellos y más evidente es el de la **redundancia**. Al estar los datos personales de un cliente en dos archivos, se duplicaría la cantidad de memoria secundaria necesaria para almacenarlos. Hoy día, esto no representa un grave problema al ser muy económica la memoria masiva, sin embargo en la década de los sesenta, cuando aparecieron las primeras bases de datos, la situación era muy distinta, ya que los discos eran periféricos caros y de difícil mantenimiento.

- Supongamos que un proveedor cambia de dirección, notificándolo al departamento de contabilidad al girarle la factura correspondiente a una entrega de material, pero no al de contabilidad. Tendremos dos direcciones distintas del mismo proveedor (siendo una de ellas errónea), dependiendo del fichero del que extraigamos dicho dato. A esto se le denomina **inconsistencia de datos**.

- La información contenida en los ficheros, solo es accesible mediante las aplicaciones desarrolladas para los distintos departamentos. Éstas cubren las necesidades de información de la organización en la mayoría de las ocasiones, pero frecuentemente, se requieren la emisión de un listado o la ejecución de un proceso no contemplado en el análisis del sistema de

información (y por tanto no incluido en la aplicación). De esta manera, el responsable del almacén puede tener necesidad de un listado de existencias ordenado por precios, cuando la aplicación proporciona dicho listado ordenado por códigos de materiales. La única opción sería la de pedir al departamento de desarrollo la inclusión de esta opción, con el consiguiente coste económico y retraso en la recepción del informe. Esta **limitación al acceso de los datos** es otro de los inconvenientes de los archivos gestionados de forma tradicional.

- El almacén, posee un terminal que recepciona las entradas de materiales recibidos y otro que controla la salida de éstos. En un momento determinado las existencias de tornillos de 3/8 son de 500 unidades y se produce una entrada al almacén de 300 unidades, y una salida de 100. Estos dos **accesos concurrentes** al fichero podrían ocasionar un error en la actualización del campo que almacena la cantidad de tornillos en existencias, pudiendo almacenar el sistema cantidades como 800 ó 400 unidades, cuando la correcta sería 700.

- Algunos datos, por motivos de **privacidad**, no deben de ser accesibles por la totalidad de los usuarios del sistema. La información del departamento de contabilidad, por contener datos económicos de otras empresas, no debe de ser conocida por los usuarios del almacén o los de compras.

- Íntimamente relacionado con la privacidad, está el problema de la **seguridad** de los datos. En los archivos tradicionales, existen importantes dificultades para controlar la modificación de campos y la eliminación de registros.

- La existencia de un fichero de clientes en el almacén y otro el departamento de ventas no significa que ambos tengan la misma descripción de registro, ni la misma información cualitativa ni cuantitativa. Ambos ficheros tendrán datos comunes (nombre, dirección, teléfono) y datos no comunes que afectarán exclusivamente al departamento en que se encuentra. La información ha de estar almacenada sujeta a unas restricciones de formato en orden a mantener una **integridad de datos** que facilite el desarrollo de nuevas aplicaciones sobre los ficheros existentes. Así, se deberían de establecer los mecanismos para evitar encontrarnos con datos numéricos en campos como nombre de cliente, o valores negativos en el campo que indica las existencias de un determinado material.

## **1.2. Concepto de Sistema de Gestión de Base de datos.**

Para superar la problemática expuesta se desarrollaron los sistemas de gestión de bases de datos (SGBD). Un SGBD es un conjunto de información interrelacionada y un software para su gestión, control y manipulación. Algunos autores restringen el concepto de SGBD a dicho software, considerándose a la base de datos (BD) como la información almacenada en los ficheros, estando formado el sistema de base de datos por las base de datos y los programas que la controla. Para nosotros, a fin de no inducir a error y de diferenciar ambos aspectos optaremos por esta segunda opción.

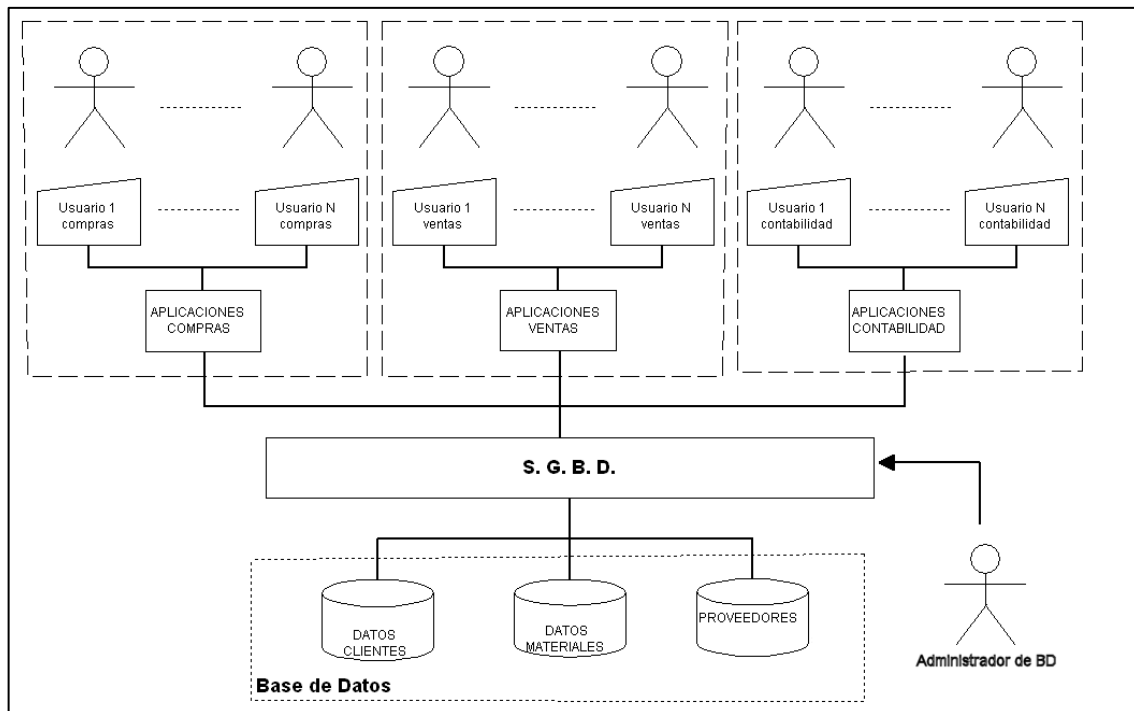


Figura 1.2.

La figura 1.2 esquematiza como se transformaría el sistema de la figura 1.1 al utilizar un entorno de base datos en lugar de archivos. En el sistema tradicional cada una de las aplicaciones deberá de realizar la descripción de los registros, así como la organización de los ficheros, organización, tipo de acceso ... etc. En un entorno de base de datos dichas especificaciones son realizadas por el SGBD, y es el administrador de la base de datos el encargado de realizar éstas y otras funciones sobre la BD.

### 1.3. Niveles de Información.

El SGBD debe de proporcionar información a usuarios y desarrolladores a distintos niveles, representando cada uno de ellos una dimensión de abstracción de datos.

- **Nivel de visión:** a este nivel, cada grupo de usuarios posee conocimiento únicamente de aquella parte de la BD que le afecta. Este "conocimiento" ha de interpretarse a nivel de abstracción. El usuario sabe de la existencia de los datos, su significado y la relación que existe entre ellos, pero ignora cualquier detalle sobre su formato, tipo, estructura de almacenamiento y en general cualquier aspecto físico.

- **Nivel conceptual:** la unión de todas las vistas dan lugar al nivel conceptual de BD. A este nivel, se conoce la descripción de todos los datos y las relaciones existentes entre ellos.

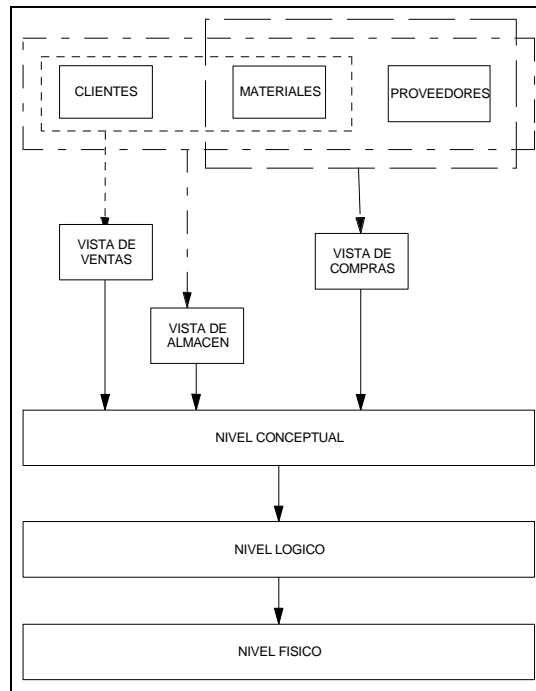


Figura 1.3.

- **Nivel lógico:** es un nivel intermedio entre el nivel conceptual y el nivel físico, siendo la consecuencia de aplicar un enfoque de datos al esquema conceptual. El usuario sigue desconociendo cómo se almacenan los datos físicamente, pero al trabajar ya sobre un determinado enfoque de datos, se tienen conocimientos sobre algunos aspectos como son la forma de relacionar los datos.

- **Nivel físico:** a este nivel se describen cómo se encuentran los datos almacenados físicamente en memoria secundaria. Este nivel es el más cercano al hardware, y se encuentra íntimamente ligado a él.

Cada uno de estos niveles de información se plasmará en un esquema; de esta manera tendremos un esquema de visión para cada una de las vistas, un esquema conceptual, un esquema lógico y un esquema físico. Nuestro estudio se limitará a los tres primeros, al ser el diseño físico secundario cuando se utiliza un enfoque de datos relacional, enfoque que utilizaremos en el diseño de B.D.

#### 1.4. Descripción funcional del SGBD.

El sistema de gestión de bases de datos se compone de un conjunto de módulos o componentes bien diferenciados

- **Lenguaje/compilador de definición de datos:** una vez diseñado el esquema conceptual, hemos de describirlo mediante un conjunto de instrucciones. Estas instrucciones pertenecen a un lenguaje integrado en el SGBD, y que de forma genérica se denomina lenguaje de definición de datos o DDL (Data Definition Language), y que como cualquier lenguaje de programación de alto nivel necesitará de un traductor para generar el código objeto a partir del código fuente.

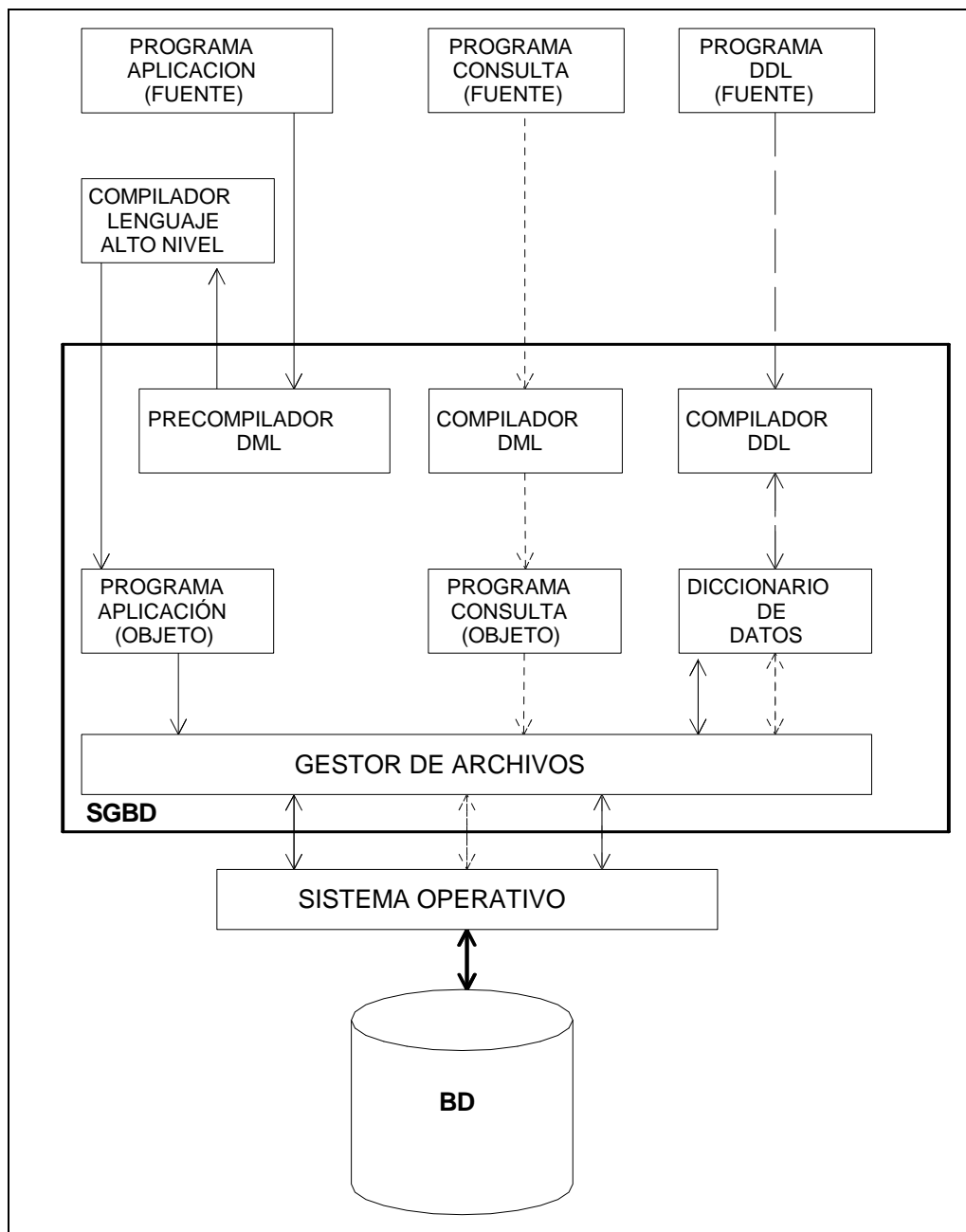


Figura 1.4.

**- Lenguaje/precompilador de manipulación de datos:** Una de las finalidades de los sistemas de bases de datos es implementar un sistema que permita acceder a la información de forma eficiente. Para ello el SGBD incluye un lenguaje de manipulación de datos o DML (data manipulation language)

El DML puede ser utilizado de dos formas diferentes. Por una parte, se incluyen sentencias DML en programas escritos en lenguajes de alto nivel (COBOL, Pascal, C ...), por lo que al primero se denomina lenguaje huésped, mientras que el segundo es el lenguaje anfitrión. Un programa que maneja información en un entorno de base de datos en lugar de un sistema de archivos no puede utilizar las instrucciones de entrada/salida propias del lenguaje, sino que sustituye éstas por instrucciones DML. Esto plantea un problema inmediato: el compilador del lenguaje anfitrión no reconoce las instrucciones "huésped", por lo que se hace necesario la inclusión en el SGBD de un precompilador que traduzca estas instrucciones DML en instrucciones reconocibles por el compilador del lenguaje anfitrión.

La otra forma de utilización del DML es mediante programas que contengan exclusivamente sentencias propias de este lenguaje o bien introduciéndolas directamente por el terminal para obtener información interactiva.

- **Gestor de base de datos o gestor de archivos:** se encarga de recibir las instrucciones DML y traducirlas a sentencias de bajo nivel comprensibles por el sistema operativo, que será en último término el que realizará los accesos a disco. Además realiza otras funciones como control de usuarios, control de seguridad y privacidad, recuperación de datos ante caídas del sistema y gestión de accesos concurrentes.

- **Diccionario de datos:** Un diccionario de datos es un sistema que almacena información sobre información. No es un componente exclusivo de los SGBD, pero es fundamental para ellos, ya que él se almacena sobre los datos y sus relaciones, esquemas, subesquemas, datos sobre ficheros, programas, restricciones, de integridad ... etc. El diccionario de datos es consultado continuamente por el resto de los módulos ya que él se encuentran almacenada información indispensable para su funcionamiento.