

# Bases de Datos (I)

## ¿ Que es una base de Datos y el SGDB que la acompaña ?

En la última década, en las grandes instalaciones de España se ha producido un gran auge de los Sistemas Gestores de Bases de Datos. Dos son los que, atendiendo al tipo y a los sistemas sobre los que corren, han destacado respecto al resto: ADABAS y DB2.

El autor de este artículo no pretende profundizar en conceptos tan amplios como los que aquí se van a enunciar, solo intenta matizar términos y conceptos que son de uso común; y, de paso, hacer que este artículo sirva de introducción a los temas que se tratarán los próximos meses en esta sección: Fundamentos de las Bases de Datos ADABAS y DB2.

### Un poco de Historia

Aunque nunca se haya estudiado este tema, todo el mundo ha oído hablar del término *BASE DE DATOS*, y mas o menos tiene una idea de su significado.

Para Charles W. Bachman, diseñador del IDS (Integrated Data Storage) , primer Sistema de Gestión de Base de Datos, abreviadamente SGDB, el mundo de las Bases de Datos nació con las grabaciones en cinta magnética de los años 50.

Pero la verdad es que al principio solo había "datos", o conjuntos de caracteres encadenados que reproducían las características de los objetos o hechos que describían.

Estos datos, que podían formar grupos repetitivos, se organizaron en "ficheros" o "archivos", los cuales recogieron como herencia la organización de los ficheros de **ficha de cartulina** tan extendidos en esa época.

Estos ficheros, base de la llamada "**Informática de Gestión**", contenían la información de la realidad cotidiana, asociaciones, correspondencias, uniones entre elementos, etc., generando con ello una red en la que los elementos que la configuraban no podían tratarse de forma independiente.

El siguiente paso, por tanto, fue la aparición de los "**ficheros consolidados**", cuyos elementos se informaban unos a otros cuando ocurría un suceso, es decir, cada vez que se modificaba o añadía un registro, se actualizaban los punteros que le unían al resto de elementos. Esto permitía mantener una cierta coherencia en los datos, pero el coste de proceso que esto suponía llegaba a dar al sistema una pesadez de tratamiento a veces insostenible, siendo esta una característica importante de los sistemas de gestión de los años 60-70

Llegados a este punto, hay que destacar el hecho de que paralelamente, grandes empresas, organismos estatales (no olvidemos que el Gobierno de los Estados Unidos es el primer consumidor de informática del mundo) y paraestatales empezaron a interesarse por el concepto "*banco de datos*", como acumulación sistemática de informaciones sobre un campo de estudio particular. Este concepto pretendía tratar la información como si se tratara de un banco, cuya moneda fuera precisamente la información. Esta organización, llamada "**cooperativa de información**" a veces también recibe el nombre de Base de Datos, lo cual no es coherente ni se ajusta a la definición que sigue.

## Definición de BD

Después de esta breve reseña histórica, se puede considerar que la Base de Datos nace de la convergencia de necesidades específicas de las empresas y de las necesidades de organizar gigantescas agrupaciones de datos a escala profesional o nacional.

La definición que se da a continuación del significado de la Base de Datos se basa principalmente en criterios técnicos más que en criterios de utilización, y pretende recoger todos los conceptos y características de una base de datos.

Una base de datos es un conjunto (a menudo voluminoso, pero no necesariamente) de datos almacenados en soportes externos en los que se permite el acceso directo. Estos datos son utilizables por los programas de aplicación (batch u on-line) a fin de realizar altas, bajas, modificaciones y búsquedas, para lo cual disponen de procedimientos automáticos de búsqueda. Dichos datos están integrados, es decir, cada dato se encuentra almacenado una sola vez, tanto en su contenido propio como en las relaciones con otros datos de la misma base, relaciones que pretenden ser el fiel reflejo de las existentes en la realidad concreta que se desea representar.

### ¿Por qué las Bases de Datos necesitan un sistema gestor?

Cuando se habla del *registro de un fichero* se está hablando de un conjunto de información que tiene una estructura determinada.

Para poder leer la información contenida en un fichero, es necesario realizar las siguientes operaciones:

- Asignar el fichero al entorno del programa que le quiere usar (es lo que se entiende por ALOCAR el fichero).
- Abrir el fichero, para lo cual debe ejecutarse la rutina del sistema operativo que realiza esa función.
- Leer el registro, es decir, obtener una copia de la información que buscamos, desde el soporte en el que se encuentra (disco por ejemplo) sobre el buffer o área de memoria que el programa ha reservado para él.
- Hacer uso de dicha información.

De esta forma, se ha traído al entorno del programa toda la información contenida en un registro. Pero, para poder hacer uso de esta información, en tiempo de codificación se ha tenido que redefinir el buffer sobre el que se va a depositar esta información. Es decir, se ha definido la plantilla que permitirá separar la lista de caracteres y números que componen el registro, en los distintos campos que dan significado a esa información.

Por otra parte, si se desea leer ese mismo fichero desde otro programa, de nuevo hay que incluir esa plantilla u otra en el nuevo programa, de modo que nos permita definir cada una de las posiciones del registro, de modo que, por ejemplo, las 15 primeras posiciones constituyen el nombre, que las 10 siguientes constituyen el sueldo, que las 8 siguientes constituyen la fecha de nacimiento, etc.

Esto presenta una serie de inconvenientes, como por ejemplo,

- todos los programadores tienen acceso a toda la información del registro, pues el registro se lee entero.
- Un mismo campo puede definirse y tener nombres distintos en distintos programas.

Para solventar, entre otros, estos inconvenientes, en los sistemas multitarea aparecieron los sistemas gestores de Bases de Datos.

### **¿Qué es un Sistema Gestor de Bases de Datos o SGDB?**

Un SGDB realmente es un conjunto de programas independientes de la aplicación que se esta ejecutando, y que junto con una serie de ficheros auxiliares, facilitan la gestión de recuperación de información y de grabación de la misma en los soportes magnéticos que la contendrán de forma permanente.

Cuando la información se encuentra en una base de datos, el programador no necesita saber si un determinado campo ocupa las 15 primeras posiciones o las 15 ultimas del registro, solo sabe que el campo se llama NOMBRE y que forma parte del fichero de PERSONAL, y que tiene una longitud de 15 bytes, por ejemplo.

Por otra parte, el SGDB determina como suyos todos los ficheros que van a componer la base de datos, por lo que ningún otro programa de aplicación podrá alocar dichos ficheros. Esto presenta al menos la ventaja de que esos ficheros se encuentran permanentemente alocados desde que se "arranca" la Base de datos, por lo que no hay que alocarles cuando se desea hacer uso de un fichero, solo hay que llamar al SGDB diciéndole que se desea recuperar los campos que se especifican de un registro determinado de tal o cual fichero, y ya él se encarga de devolver esa información en el buffer habilitado al efecto.

Al hacerlo de este modo, se resuelven los dos problemas que se planteaban en el apartado anterior, pues:

- solo se recuperan los campos que se especifiquen siempre y cuando se este autorizado para su manejo (ya que al ser el SGDB un programa que se interpone entre el programa de aplicación y el soporte físico de los datos, se le puede especificar determinados niveles de autorización para devolver determinada información).
- Todos los programas usaran el mismo nombre para referirse al mismo campo, ya que ese nombre es el que conoce el SGDB.

### **Tipos de Sistemas Gestores de Bases de Datos**

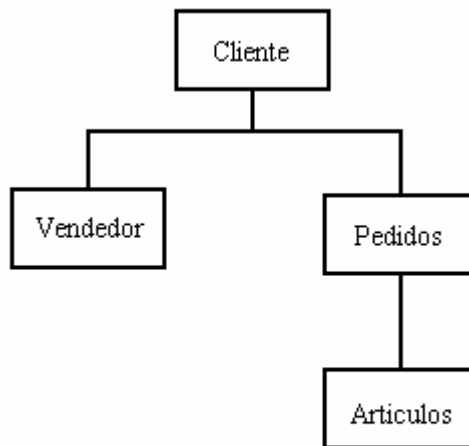
Como ya hemos dicho, una Base de datos debe, ser considerada, no solo por los soportes de la información, sino que además debe considerarse el Sistema Gestor de la Base de Datos o SGDB, es decir, el software que, al tiempo que sirve para independizar al usuario del almacenamiento, va a asegurar éste, en el sentido de que si no se conoce la organización del almacenamiento no se podrá sacar o modificar datos de su contenido.

Fundamentalmente son 4 los sistemas gestores:

- Jerárquicos
- En red.
- Basados en listas Invertidas
- Relacionales

Los sistemas **Jerárquicos** implementan estructuras basadas en un modelo lógico donde cada entidad se descompone de modo progresivo en otras entidades cada vez más elementales. El SET o registro lógico esta formado por un segmento raíz, del que pueden colgar varios hijos, de los que a su vez pueden colgar otros, etc. Cada uno de estos segmentos dependientes, están unidos entre si mediante distintos tipos de punteros que facilitan la recuperación de la información.

La figura 1 muestra un modelo de este tipo de estructura, mientras que la figura 2 muestra la implementación de una ocurrencia concreta.



Estructura de Información Jerárquica

Figura 1

Desarrollo del esquema Jerárquico

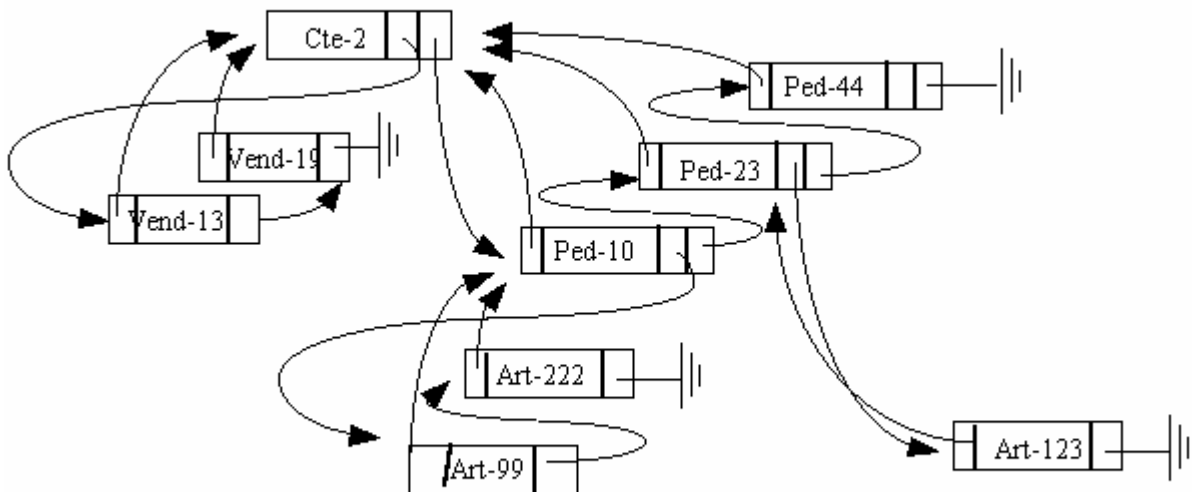


Figura 2

**Nota:** Al final del artículo se comenta esta realidad concreta

Las bases mas representativas de este tipo fueron IMS de IBM y SYSTEM-2000

Los sistemas en **RED** sustentan un modelo de datos que admite que un miembro tenga varios propietarios, pudiendo estos pertenecer a la misma o a varias clases de entidad propietario. El método usado para implementar este modelo se basa en punteros, los cuales materializan las relaciones entre los distintos elementos de las entidades.

La figura 3 muestra un modelo de este tipo de estructura, y la figura 4 muestra la implementación de una realidad concreta.

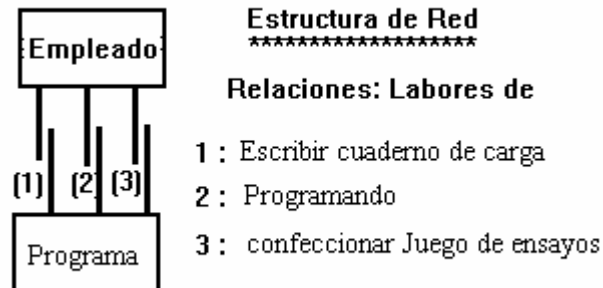


Figura 3

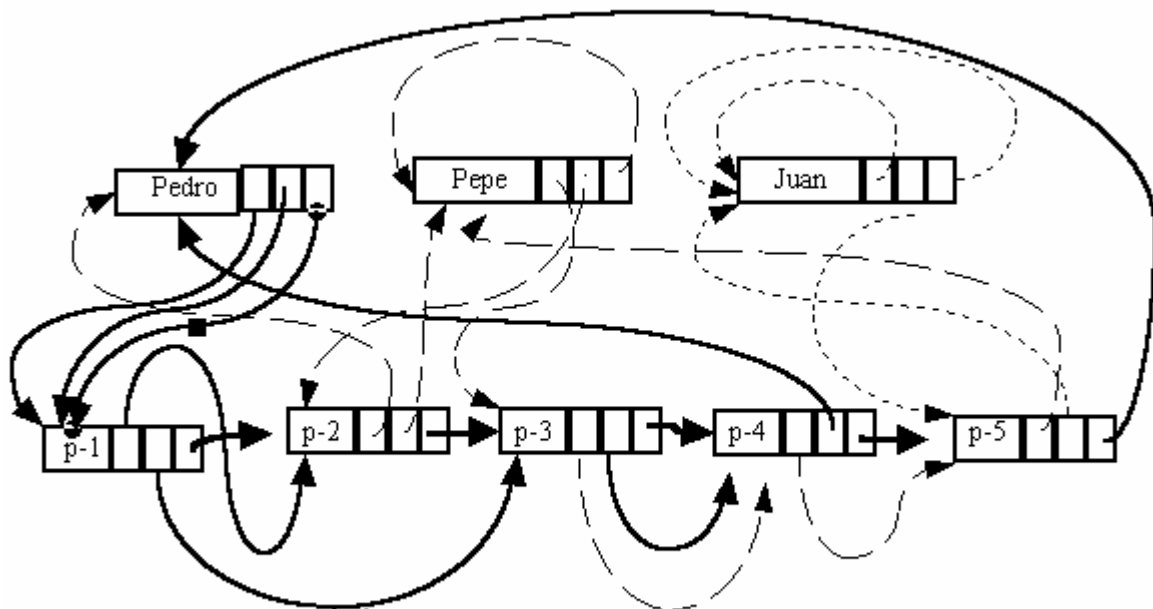


Figura 4 Desarrollo de la red

**Nota:** Al final del artículo se comenta esta realidad concreta

En este tipo de estructuras, se deben diferenciar las que seguían las normas CODASYL, tales como IDS, IDS II (Honnevell Bull), IDMS (IBM), y DMS (Univac)

Sin Normas Codasy: TOTAL (IBM, Honnewell, Univac) (solo admitía dos niveles )

Los sistemas basados en **listas Invertidas**:

El principio de este modelo consiste en asignar al registro lógico un identificador único, y usar dicho número en la confección de las listas que permitirán recuperar la información en base a los campos claves que se hayan definido.

Este es el modelo adoptado para la gestión de la base de datos ADABAS. (El artículo del próximo mes se dedica exclusivamente a tratar este sistema gestor)

Los sistemas **Relacionales**:

Este tipo de base de datos se fundamenta en el concepto matemático de RELACION.

Así, una relación n-aria R definida en los conjuntos C1, C2, C3, ... Cn no necesariamente disjuntos, es un subconjunto del producto cartesiano de esos n conjuntos.

A cada elemento de una relación binaria se le conoce con el nombre de "*par ordenado*" o "*Tupla*", pues esta formado por un elemento de C1 y otro de C2 y en este orden.

No obstante, y a pesar del enfoque matemático de este planteamiento, no se debe caer en la trampa de pensar que los conceptos vertidos en las BDRs no tienen nada que ver con los conceptos informáticos, ya que como se vera en próximos artículos, es posible traducirlos a algo que nos es tan familiar como las tablas de doble entrada.

Ejemplos de bases de Datos que implementen este modelo, si hacemos caso a la propaganda del fabricante, son prácticamente todas las que hoy en día se encuentran en el mercado, tales como Oracle, Informix, etc, pero, en grandes sistemas la que se destaca es DB2 (de IBM). (El tercer artículo de esta terna dedicada a los SGBD se dedica exclusivamente a tratar este sistema gestor)

### **Características generales de las BD.**

Como características generales cabe destacar las siguientes:

- Las Bd. suelen ser voluminosas, lo cual no quiere decir que porque un fichero de datos sea muy voluminoso, sea una base de datos.
- Las Bases de datos deben tener en cuenta su capacidad de crecimiento en un futuro.
- La integración de los datos se encuentra dentro de la Base de datos, pues cada aplicación no usa sus ficheros, sino que todos usan los mismos ficheros.
- Las Bases de datos, tal y como se desprende de la definición dada, han de tener acceso directo a los datos.
- Actualmente se entiende por base de datos, no solo el almacenamiento de las bases de datos, sino que esta incluido el sistema de gestión de bases de datos

Por ultimo, cabe destacar que aunque la definición dada anteriormente, especifica que los datos están almacenados una sola vez, es decir, que no hay *redundancia*, eso no es totalmente cierto en ninguno de los sistemas comentados, si bien, esta redundancia cada sistema tiende a que sea mínima y que este controlada por el propio sistema.

## Formas Normales

Al diseñar el modelo de datos en el que se basará una aplicación, se deben tomar una serie de precauciones a fin de evitar problemas a la hora de realizar altas, bajas y/o modificaciones de las tuplas. Las formas normales pretenden mantener la estructura lógica de los datos en una forma *normal* y limpia.

Se dice que una relación está *sin normalizar* cuando existen tuplas en las que algún atributo es múltiple. (figura-5)

Codigo	Sueldo	Comisiones mensuales											
		1	2	3	4	5	6	7	8	9	10	11	12

Figura 5 : Relacion sin normalizar

*Primera forma normal (1FN)*: En un modelo se dice que está normalizado hasta la primera forma normal (1NF en Inglés) cuando ninguna entidad del modelo contiene atributos múltiples. (Figura-6)

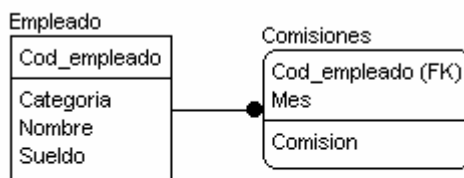


Figura 6 : Diseño normalizado

*Segunda Forma Normal (2FN)*: Esta forma se cumple cuando, además de cumplirse la Primera forma normal, todo atributo no primario, depende de la clave concatenada en su totalidad. Es decir, los atributos no clave dependen de la totalidad de la clave concatenada.

*Tercera Forma Normal (3FN)*: Esta forma se cumple cuando, además de cumplirse la Segunda forma normal, cada atributo no primario depende de una única clave primaria. Es decir, los elementos no clave dependen de una única clave.

*Cuarta Forma Normal (4FN)*: Esta forma introduce el concepto de dependencia multivalor o multivaluada (DMV o MVD).

*Quinta forma normalizada (5FN)*: Esta forma se basa en el concepto de dependencia de asociación o dependencia JOIN.

## Integridad referencial

Este concepto, básico en los SGBD, establece que una tupla de una relación que haga referencia a otra relación, debe referirse a una tupla existente de esta otra relación. Este principio, asegura por lo tanto, la congruencia entre las tuplas de las distintas relaciones.

## Comentarios a las figuras que acompañan el artículo:

**Figura 1:** representa una estructura de información Jerárquica:

**Figura 2:** Representa una realidad concreta.

- El segmento *CLIENTE* tiene dos punteros:
  - el primero apunta al primer vendedor que le ha atendido.
  - el segundo puntero apunta al primer pedido que ha realizado.
- El segmento *VENDEDOR* tiene dos punteros:
  - El primero apunta al cliente que ha generado esta ocurrencia.
  - el segundo apunta al siguiente vendedor que ha atendido a este cliente.
- El segmento *PEDIDO* tiene tres punteros:
  - El primero apunta al cliente que ha realizado el pedido
  - el segundo apunta al primer artículo del pedido.
  - el tercero apunta al siguiente pedido del cliente.
- El segmento *ARTICULO* tiene dos punteros:
  - El primero apunta al Pedido al que pertenece
  - El segundo apunta al siguiente artículo del pedido.

El cliente 2 ha sido atendido por los vendedores 13 y 19.

Ha hecho el pedido 10, en el que pedía los artículos 99 y 222, el pedido 23, en el que pedía el artículo 123, y el Pedido 44, ...

**Figura 3:** Representa el modelo de datos de una red simple, en la que se establecen tres relaciones entre la entidad PROGRAMADOR y la entidad PROGRAMA, de modo que :

- Un PROGRAMADOR ha realizado labores de *ESCRIBIR EL CUADERNO DE CARGA DEL PROGRAMA*.
- Un PROGRAMADOR ha realizado labores de *PROGRAMAR EL PROGRAMA*.
- Un PROGRAMADOR ha realizado labores de *CONFECCION DEL JUEGO DE ENSAYO* para el PROGRAMA.

**Figura 4:** Materializa la realidad de las labores realizadas por los Programadores Pedro, Pepe y Juan, mediante los tres punteros que tienen todos los segmentos.

- El primer puntero materializa la primera relación:
  - Pedro : ha confeccionado el cuaderno de carga de P-1 y P-2.
  - Pepe : los de P-3, P-4 y P-5.
  - Juan : No ha realizado ninguno.
- El segundo puntero materializa la relación "Ha programado"
  - Pedro : ha programado P-1, P-3 y P-4
  - Pepe : P-2
  - Juan: P-5
- El Tercer Puntero materializa la relación "Ha confeccionado el juego de ensayo del programa"
  - Pedro : Ha confeccionado los de P-1, P-2, P-3, P-4 y P-5
  - Pepe : ninguno.
  - Juan: Ninguno.