

Tema 1 .- Introducción a las Bases de Datos

Índice

1.- Introducción

2.- Objetivos de una Base de Datos (Características)

3.- Arquitectura de las Bases de Datos (Niveles)

4.- Conceptos de Bases de Datos y Sistemas de Gestores de Bases de Datos

5.- Modelos de Bases de Datos

6.- Arquitectura Cliente Servidor

1.- INTRODUCCIÓN

Existen en principio dos formas de manejar la información :

- **Ficheros**
- **Bases de Datos**

Un Fichero es un conjunto de datos relacionados entre si que tienen entidad propia y dan una idea completa de una cierta actividad.

Cada Fichero que utilizamos debe de tener un ***nombre*** para distinguirlos de los demás.

El ***nombre*** consta a su vez de 2 Partes :

Nombre y Extensión (separados por un punto)

- ❑ **Nombre** es un conjunto de caracteres, limitado en su número según las aplicaciones y/o lenguajes de programación (en muchos casos su longitud máxima es 8 caracteres).
- ❑ **Extensión** Es un conjunto de 3 caracteres que dependen de los lenguajes y del objetivo del mismo.

1.- INTRODUCCIÓN

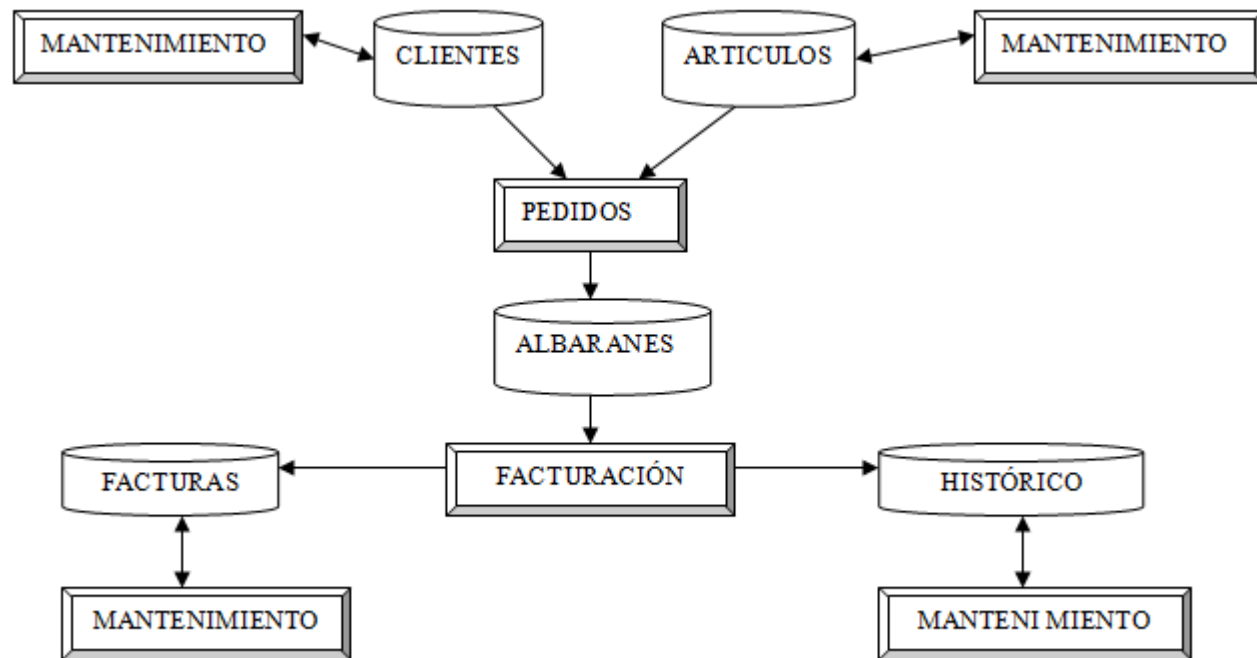
- El fichero se compone a su vez de **Registros** que son la cantidad mínima de información que se transfiere en una operación de Entrada / Salida.
- Los Registros se componen a su vez de **Campos** que son la mínima cantidad de información que tiene entidad por si misma (que tienen sentido por sí solos)
- Un esquema de un fichero podría ser:

Registro 1	Campo 1	Campo2	Campo 3	Campo m
Registro 2	Campo 1	Campo2	Campo 3	Campo m
.....	Campo 1	Campo2	Campo 3	Campo m
.....	Campo 1	Campo2	Campo 3	Campo m
Registro n	Campo 1	Campo2	Campo 3	Campo m

1.- INTRODUCCIÓN

Un **Aplicación Informática diseñada con ficheros** consiste en una serie de **Ficheros Independientes** entre si ,relacionados con una serie de **Programas** que nos permiten controlar informáticamente una cierta actividad humana.

Un Posible esquema de un programa de Facturación podría ser:



1.- INTRODUCCIÓN

Las características generales de una aplicación con ficheros son:

- **Cada Fichero va Asociado al menos a un Programa** en el cual se definen una serie de acciones sobre el mismo, llamado normalmente *mantenimiento del fichero*:
Creación, altas, Bajas, Consultas, Modificación, Listados, etc.
 - **Hay Programas que relacionan dos o más ficheros** que pueden generar o modificar a otros ficheros.
 - **Existen Campos que se repiten en diferentes ficheros.**
 - **Si se añaden o suprimen campos de un fichero, se está OBLIGADO a cambiar todos los programas que usen dicho fichero.**
-

1.- INTRODUCCIÓN

- Una **Base de Datos** se define como un conjunto de datos, generalmente distribuidos en **Tablas** que pueden ser usados de forma independiente por distintos programas y/o usuarios.
 - La característica fundamental de una base de datos es la **Independencia total de los programas**.
 - Otra característica es la **Independencia Física**, es decir la distribución de los datos en los Dispositivos de Almacenamiento Masivo (DAM), es independiente de la estructura Lógica de la Base de Datos.
-

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Independiente del modo en que esté organizada una base de datos debe de intentar tener las siguientes características.

- ❑ **Versatilidad**
- ❑ **Desempeño**
- ❑ **Redundancia Mínima**
- ❑ **Capacidad de Acceso**
- ❑ **Integridad**
- ❑ **Simplicidad**
- ❑ **Seguridad y Privacidad**
- ❑ **Afinación**
- ❑ **Permanencia**

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Versatilidad :

- Los datos se podrán usar de diferentes maneras.
 - Cada Usuario y/o programa usará sus mismas fuentes de datos que podrán ser las mismas o distintas de otro usuario.
 - El sistema deberá permitir derivar datos y relaciones a partir de los ya existentes.
-

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Desempeño :

- El **tiempo de respuesta** en el acceso a los datos debe ser el adecuado. Es decir pocos segundos.
 - Debe de permitir el **acceso simultaneo**, de varias aplicaciones o usuarios.
-

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Redundancia Mínima :

La BD debe de perseguir que los mismos Items (datos) no se repitan en sitios distintos, siempre que no conlleve una mayor complejidad de la BD ni una reducción en su rendimiento.

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Capacidad de Acceso :

Es tarea de los diseñadores de la BD que dicha organización permita unos accesos rápidos y eso Depende de la *Organización Física de los Datos.*

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Integridad :

- Los Datos almacenados en la BD deben de ser **fiables según las restricciones** definidas por el sistema (en el diseño).
- Los fallos que se produzcan en el sistema **no pueden destruir los datos.**
- El sistema **debe de poder recuperar la información en caso de un fallo.**

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Simplicidad :

- La BD debe de estar basadas en representaciones Lógicas Simples que permitan su verificación.
 - La Representación de la BD debe de estar ***normalizada***, para que pueda ser entendida en cualquier Gestor de BD.
-

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Seguridad y Privacidad :

Los datos de la BD deben de estar protegidos ante :

- ❑ Fallos del Sistema.
 - ❑ Usos Indebidos o malintencionados.
 - ❑ Accidentes, robos, etc
 - ❑ Usuarios no identificados.
 - ❑ Acceso de usuarios a datos sobre los que no tiene permiso.
 - ❑ Fallos de hardware o software.
 - ❑ Descontrol de accesos indebidos por los usuarios.
-

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Afinación :

La Organización Física de los Datos debe ser independiente de la Organización Lógica.

Es decir **si tenemos que cambiar** ya sea por un aumento de volumen de datos o por ajuste para mejorar los accesos a los mismos; ello **no debe implicar modificaciones en programas, representaciones, etc.**

2.- OBJETIVOS DE UNA BASE DE DATOS (Características)

Permanencia :La BD debe de contar con herramientas que nos conecten el pasado con el futuro. Es decir la BD debe de :

- Contar con los datos almacenados previamente.
 - Estar preparada para futuros cambios de los datos y de los medios de almacenamiento, sin afectar a los programas (**independencia física de los datos**)
 - Al cambiar la estructura lógica de los datos (ej añadir algún campo) no debe de afectar a los programas (**independencia lógica de los datos**)
-

3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)

Los usuarios terminales de la BD deben de tener una visión **lo más Abstracta** de la misma (sin conocer la organización ni el almacenamiento de los datos) y **lo más simple posible** para que pueda interpretarlos y manejarlos.

Se distinguen 3 niveles en una BD, según la visión que se tengan de los datos:

- **Nivel Externo**
- **Nivel Conceptual**
- **Nivel Interno**

3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)

Nivel Externo :Es el nivel **más cercano a los usuarios**, que le permite ver los datos y actuar sobre ellos (según sus permisos) mediante los programas de la aplicación pertinente.

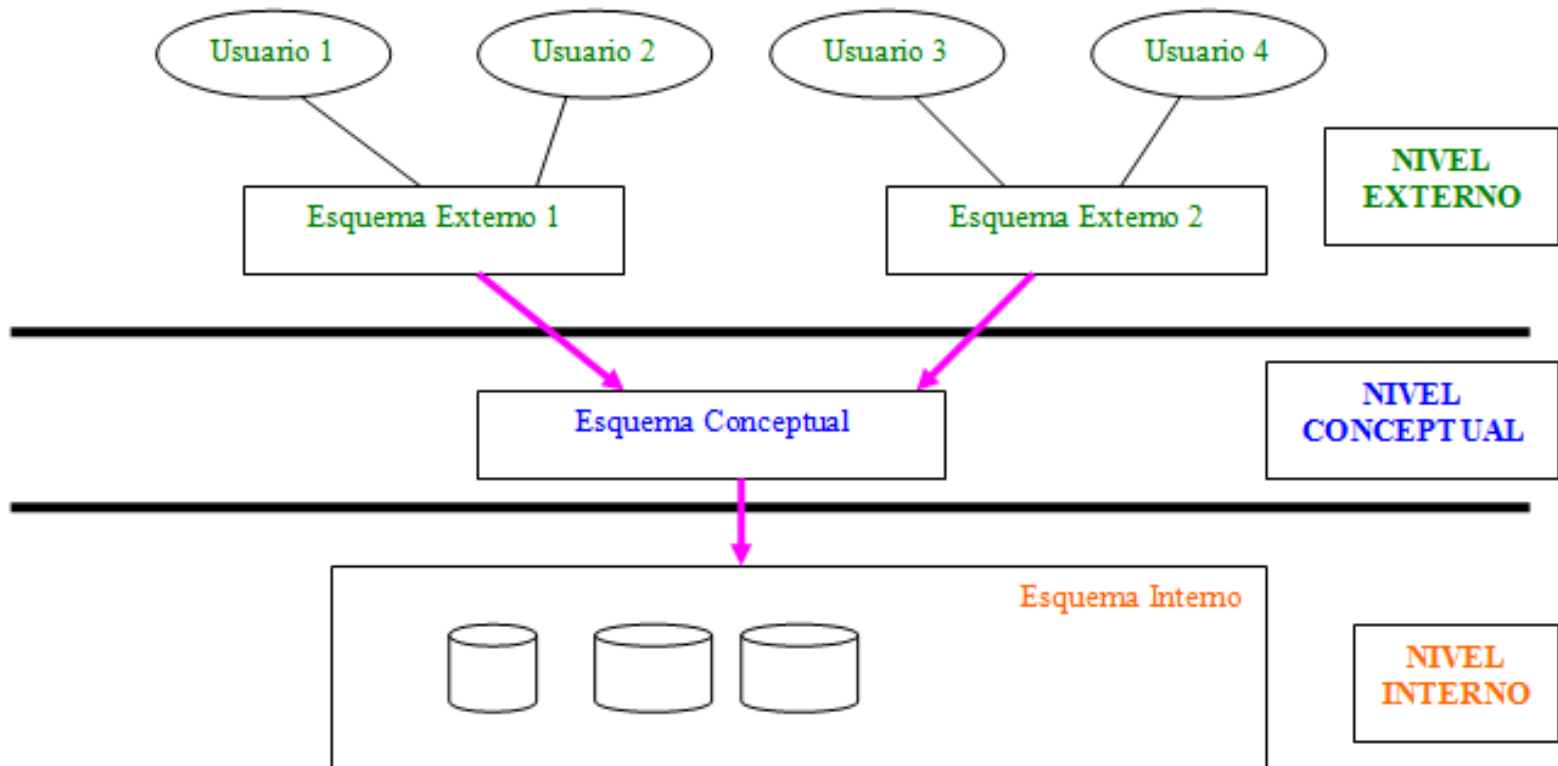
Nivel Conceptual (o Lógico) :Es la **visión global** de los datos a partir de los requerimientos de los usuarios.

- ❑ Se representa por el llamado **Esquema Conceptual**.
- ❑ En él se determinan las entidades, relaciones, dependencias, etc.
- ❑ No se tiene en cuenta la Organización Física ni los Métodos de Acceso.

Nivel Interno (Nivel Físico o Visión Física): Es el **más cercano al almacenamiento físico de los datos**. En el se definen:

- ❑ Los ficheros que contienen la información
- ❑ Su organización
- ❑ Su ubicación
- ❑ La forma de acceso a los registros
- ❑ Los tipos de registros: longitud, campos, índices, etc

3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)



3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)

Ejemplo: Notas de 1º DAW

NIVEL EXTERNO: Tabla de Usuario con el Curso, Alumno, Asignatura y Nota

CURSO	NOMBRE	ASIGNATURA	NOTA
3	JUAN	FCT	APTO
2	ANA	DFSI	5
3	ANA	SGBD	6
3	ANA	IUA	8
3	ANA	FCT	NO APTO
3	ANA	PRO	APTO
3	LUIS	SGBD	3

3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)

NIVEL CONCEPTUAL: Definición de tablas

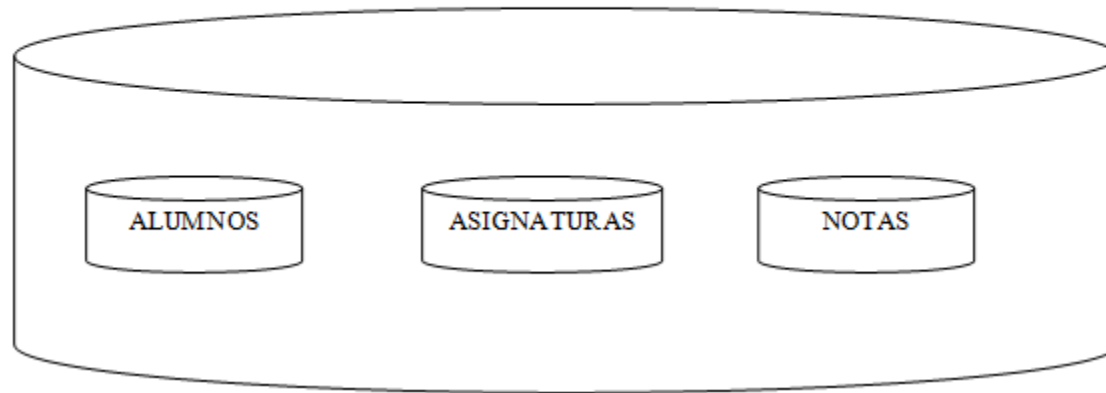
Nº EXPE.	NOMBRE	CURSO	DIRECCIÓN	POBLACIÓN
13126	JUAN	3	Rueda 1	Sevilla
13458	ANA	2	Mudo 3	Camas
13458	ANA	3	Mudo 3	Camas
13622	LUIS	3	Alfarería 1	Sevilla

CODIGO	ASIGNATURA
10	DFSI
11	SGBD
12	IUA
13	FCT
14	PRO

Nº EXPE	CURSO	CODIGO	NOTA
13126	3	13	APTO
13458	2	10	5
13458	3	11	6
13458	3	12	8
13458	3	13	NO APTO
13458	3	14	APTO
13622	3	11	3

3.- ARQUITECTURA DE LAS BASES DE DATOS (NIVELES)

NIVEL INTERNO:



4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

DIFERENCIA ENTRE BASE DE DATOS Y SISTEMA DE GESTIÓN DE BASES DE DATOS

Una **Base de Datos (BD)** es un conjunto de Ficheros relacionados que almacenan los datos correspondientes a una cierta actividad, su representación abstracta y una serie de restricciones de los mismos.

Un **Sistema de Gestión de Base de Datos (SGBD)** es un conjunto de programas que permiten administrar y gestionar la información de una BD, proporcionando a los usuarios realizar las siguientes tareas:

- ❑ Definición de los datos.
- ❑ Manipulación de los datos: inserción, actualización, borrado, consulta, etc
- ❑ Mantenimiento de la Integridad de los datos
- ❑ Control de la Privacidad
- ❑ Etc

4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

Además de estas funciones , el SGBD necesita de otros componentes personales que gestione la información de la BD de forma adecuada. Estos son :

- ❑ **El Gestor de la Base de Datos**
 - ❑ **Los Lenguajes**
 - ❑ **El Diccionario de Datos**
 - ❑ **El Administrador de la Base de Datos**
-

4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

Gestor de la Base de Datos: Conjunto de programas transparentes al usuario que se encargan de garantizar:

- ❑ La privacidad, seguridad e integridad de los datos.
- ❑ El acceso concurrente a los mismos.
- ❑ La interconexión con el Sistema Operativo
- ❑ La interfaz entre los datos, los programas y usuarios.
- ❑ Cualquier operación que ataque a la BD.
- ❑ El Gestor almacena en el Diccionario de Datos toda la descripción de la BD, usuarios permitidos y las autorizaciones pertinentes.

Existe un Usuario Administrador que realiza estas tareas.

4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

Los Lenguajes: El SGDB debe de proporcionar los lenguajes para definir y manipular los datos para todos los usuarios, sean administradores o no.

- **DDL : Lenguaje de Definición de Datos** (Data Definition Language)
Permite crear la estructura de la base de datos, incluyendo todos los objetos que puede incluir la misma (tablas, vistas, usuarios, procedimientos, funciones etc.)
- **DML : Lenguaje de Manipulación de Datos** (Data Manipulation Language)
Permite Manipular los datos : insertar, modificar, eliminar, recuperar, etc.
- **DCL : Lenguaje de Control de Datos** (Data Control Language)
Sirve para controlar el acceso a la información de la Base de Datos, definiendo privilegios, etc. De esta tarea se encarga el Administrador

4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

El diccionario de datos: es una base de datos que contiene los metadatos (datos acerca de los datos) de la base de datos, esto es:

- ❑ La definición de todos los objetos existentes en la base de datos: tablas con sus columnas, vistas, procedimientos, índices, etc...
- ❑ La ubicación física de los objetos y el espacio asignado a los mismos.
- ❑ Los privilegios y roles asignados a los usuarios.
- ❑ Las restricciones de las tablas.
- ❑ Estadísticas de uso de la base de datos.
- ❑ Información del consumo de recursos actual.
- ❑ Y un larguísimo etcétera...

4.- CONCEPTOS DE BASES DE DATOS Y SISTEMAS GESTORES DE BASES DE DATOS

El Administrador de la Base de Datos (ABD o DBA)

El Administrador de la BD es una persona o grupo de personas responsable de la seguridad y el control de los datos. Si un programador desea crear algún objeto nuevo, deberá pedir autorización del administrador para que éste le ceda los privilegios necesarios para poder modificar las estructuras de la BD de la forma más conveniente del sistema. Esta cesión está limitada al tiempo necesario para su implementación.

Son tareas del Administrador :

- ❑ Definición y mantenimiento del Esquema Lógico
- ❑ Definición y mantenimiento del Esquema Físico
- ❑ Definición de Subesquemas o Visiones de usuario de la BD (Nivel Externo).
- ❑ Concesión de autorización de accesos y privilegios a los usuarios.
- ❑ Mantenimiento de la seguridad de los datos y recuperación de los mismos.

5.- MODELOS DE BASES DE DATOS

Los Modelos más extendidos son :

Jerárquico

En Red

Relacional

5.- MODELOS DE BASES DE DATOS

Jerárquico

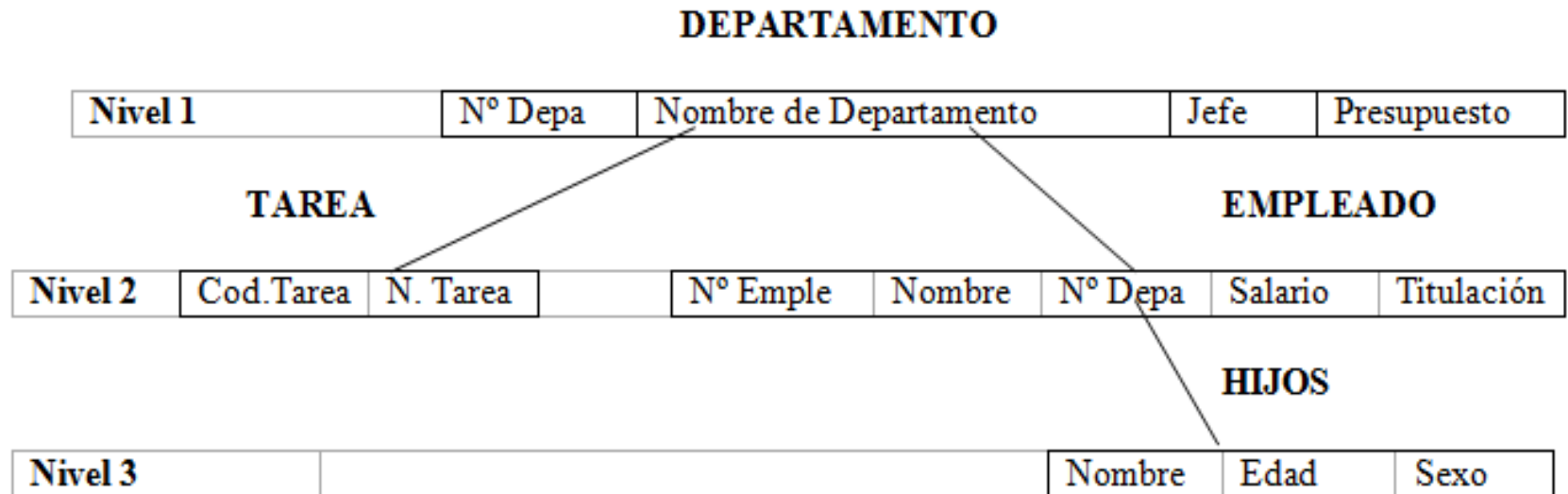
Se sirve de la estructura de **árboles** para la representación lógica de datos.

Un **árbol** está compuesto por **nodos**. En el nivel más alto de la jerarquía hay un único nodo llamado **Raíz**. Cada nodo representa un tipo de registro llamado **segmento**, con sus correspondientes campos

Sólo existen relaciones entre nodos **1 : M**. Es decir un Nodo Padre puede tener varios Nodos Hijos, pero un Nodo Hijo sólo puede tener un Nodo Padre.

5.- MODELOS DE BASES DE DATOS

Ejemplo 1: Los Departamento de una empresa con 4 Segmentos y 3 Niveles

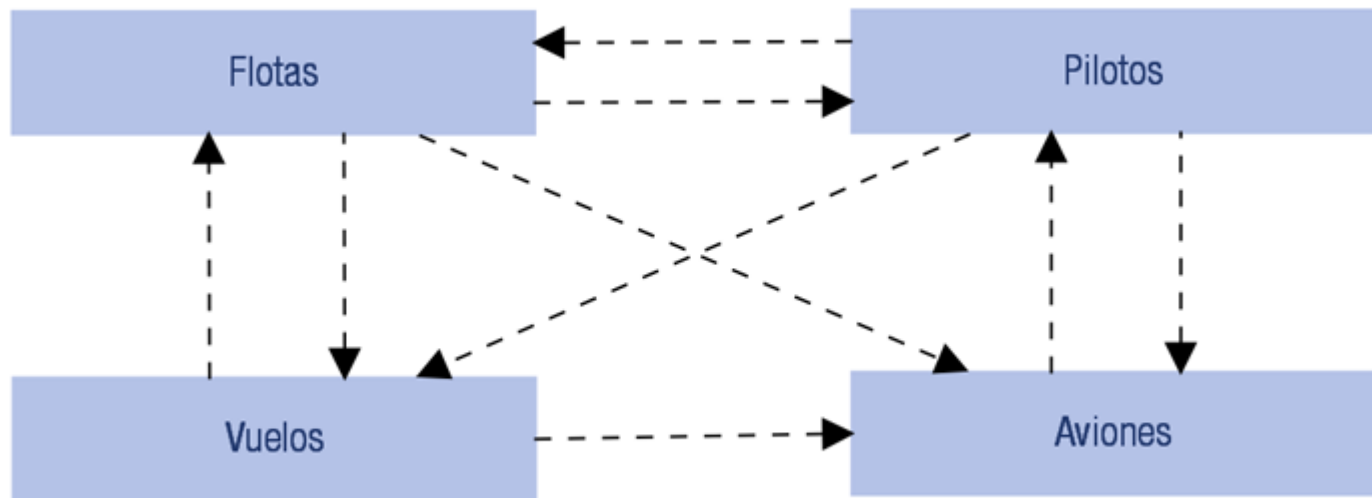


5.- MODELOS DE BASES DE DATOS

En red

Organiza la información en **registros** (también llamados nodos) y **enlaces**. En los registros se almacenan los datos, mientras que los enlaces permiten relacionar estos datos. Las bases de datos en red son parecidas a las jerárquicas **sólo que en ellas puede haber más de un padre**.

Las entidades se representan como **nodos** y las relaciones como **Líneas** que unen los nodos.



5.- MODELOS DE BASES DE DATOS

Relacional

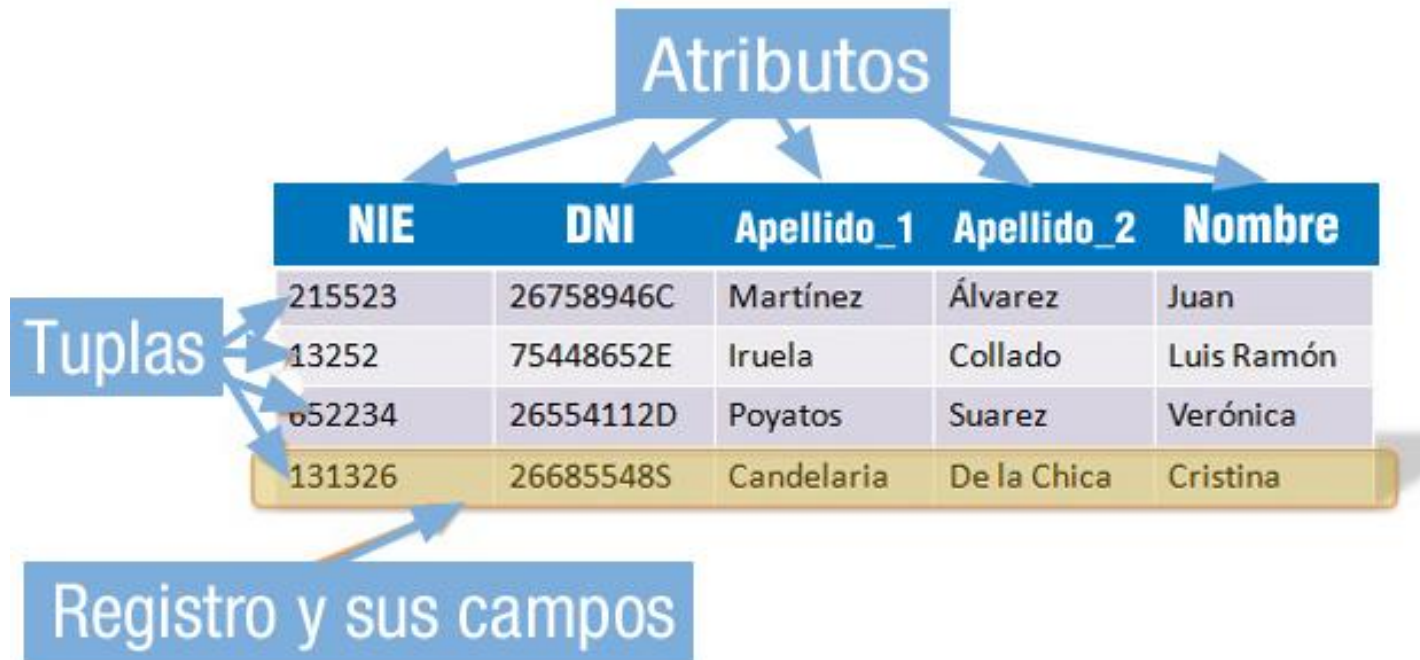
- Este modelo es posterior a los dos anteriores y fue desarrollado por Codd en 1970. Hoy en día las bases de datos relacionales son las más utilizadas.
- En el modelo relacional la base de datos es un conjunto de tablas sólo a nivel lógico, ya que a nivel físico puede estar implementada mediante distintas estructuras de almacenamiento.
- El modelo relacional utiliza **tablas bidimensionales** (relaciones) para la representación lógica de los datos y las relaciones entre ellos. Cada relación (tabla) posee un nombre que es único y contiene un conjunto de columnas.
- Se llamará **registro, o tupla** a cada fila de la tabla y **campo o atributo** a cada columna de la tabla.
- A los conjuntos de valores que puede tomar un determinado atributo, se le denomina **dominio**.
- Una **clave** será un atributo o conjunto de atributos que identifique de forma única a un registro.

5.- MODELOS DE BASES DE DATOS

- Las tablas deben cumplir una serie de requisitos:
 - Todos los registros son del mismo tipo.
 - La tabla sólo puede tener un tipo de registro.
 - No existen campos o atributos repetidos.
 - No existen registros duplicados.
 - No existe orden en el almacenamiento de los registros.
 - Cada registro o tupla es identificada por una clave que puede estar formada por uno o varios campos o atributos.
- A continuación puedes observar cómo es una relación con sus tuplas y atributos en el modelo relacional.

5.- MODELOS DE BASES DE DATOS

RELACIÓN ALUMNOS

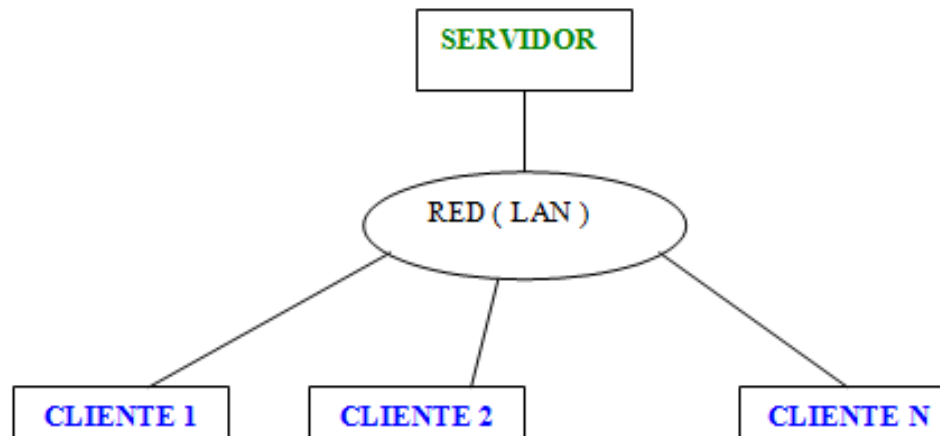


6.- ARQUITECTURA DE CLIENTE / SERVIDOR

La Arquitectura de Cliente / Servidor consta de 2 partes: **Servidor y Cliente**

Servidor: Es normalmente un gran ordenador que actúa como depósito de los datos y además contiene también el Sistema Gestor de la Base de Datos (SGBD)

Cliente: Son las estaciones de trabajo (PCs), que están conectados en red con el Servidor, al cual le solicitan una serie de servicios.



6.- ARQUITECTURA DE CLIENTE / SERVIDOR

El Software de Cliente/Servidor posee una serie de componentes que se pueden asociar a uno o a otro. Dichos componentes son:

- **Software de Gestión de Datos:** Se encarga de la manipulación y gestión de datos requeridos por las aplicaciones. Reside normalmente en el Servidor.
- **Software de Interacción con el Usuario:** Implementa las funciones que se asocian a una interfaz gráfica de usuario . Reside normalmente en el Cliente.
- **Software de desarrollo:** Se utiliza normalmente para desarrollar aplicaciones. Reside normalmente en el Cliente.

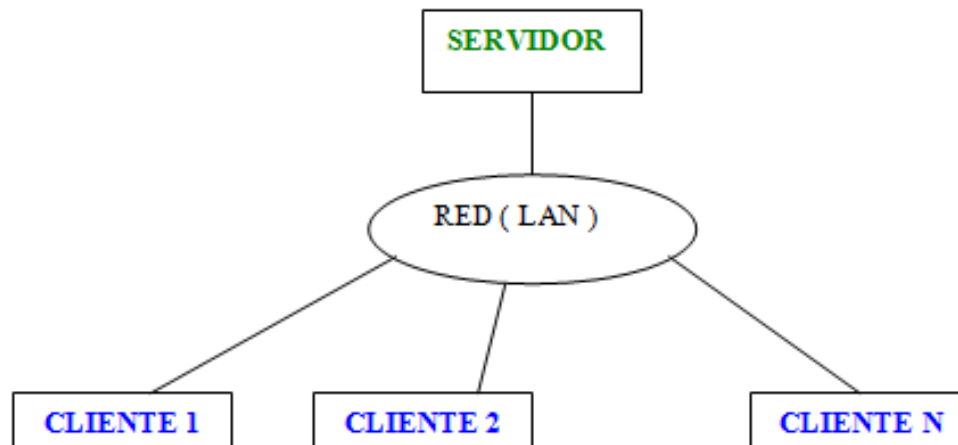
6.- ARQUITECTURA DE CLIENTE / SERVIDOR

La arquitectura de cliente servidor permite varias configuraciones:

- **Anfitrión**
- **Cliente/Servidor**
- **BDs Distribuidas**
- **BDs con Servidores de Aplicaciones**

6.- ARQUITECTURA DE CLIENTE / SERVIDOR

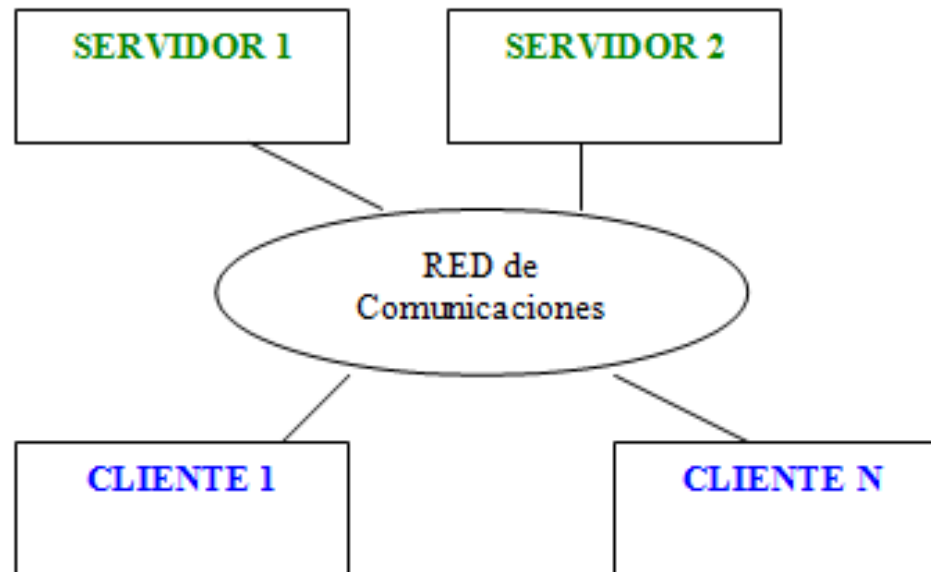
- **Anfitrión:** La máquina cliente y servidor es la misma. Los datos y aplicaciones están juntos.
- **Cliente/Servidor:** La BD reside en una máquina, el servidor, y los usuarios acceden a la BD como clientes a través de una red.



6.- ARQUITECTURA DE CLIENTE / SERVIDOR

■ BDs Distribuidas:

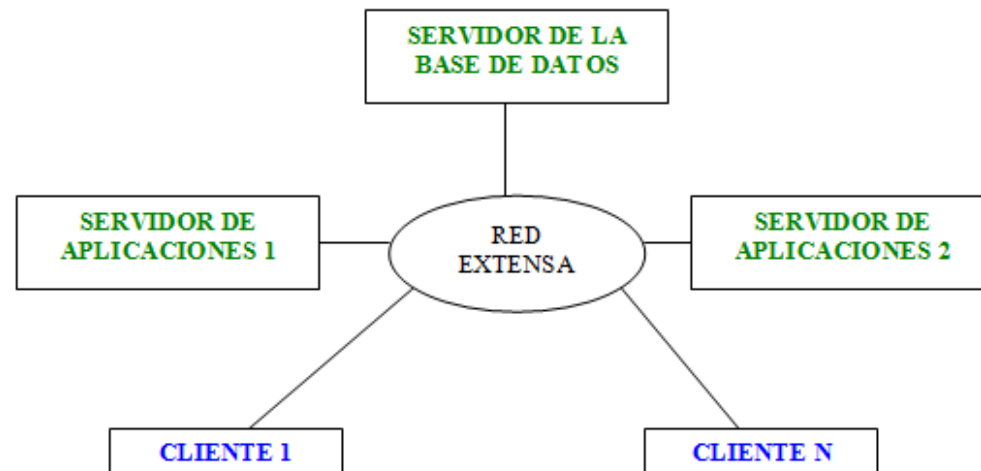
La BD está repartida entre varios servidores y los usuarios no conocen la ubicación física de los datos.



6.- ARQUITECTURA DE CLIENTE / SERVIDOR

■ BDs con Servidores de Aplicaciones:

Existen distintos servidores donde residen las Aplicaciones. Los Clientes acceden a los datos y aplicaciones a través de una Red de Área Amplia ó Extensa (WAN) e Internet. Los clientes no requieren ninguna administración pues **TODOS LOS DATOS Y APLICACIONES RESIDEN EN LOS SERVIDORES** y el cliente se limita a solicitar una serie de servicios.



6.- ARQUITECTURA DE CLIENTE / SERVIDOR

La arquitectura en Servidores de Aplicaciones tiene **2 ventajas** fundamentales sobre la de Cliente / Servidor:

- Como las aplicaciones se instalan en Servidores, en los clientes **no se necesita ninguna administración. Cualquier Cliente accede a la BD y sus distintas aplicaciones a través de un navegador WEB** que soporte Java.
- La Cliente/Servidor **ocasiona más gastos** en la administración de la BD, ya que **obliga a instalar las aplicaciones en todos los clientes**. Además crea grandes exigencias a la red e imposibilita el uso de aplicaciones en Redes Extensas (WAN).