

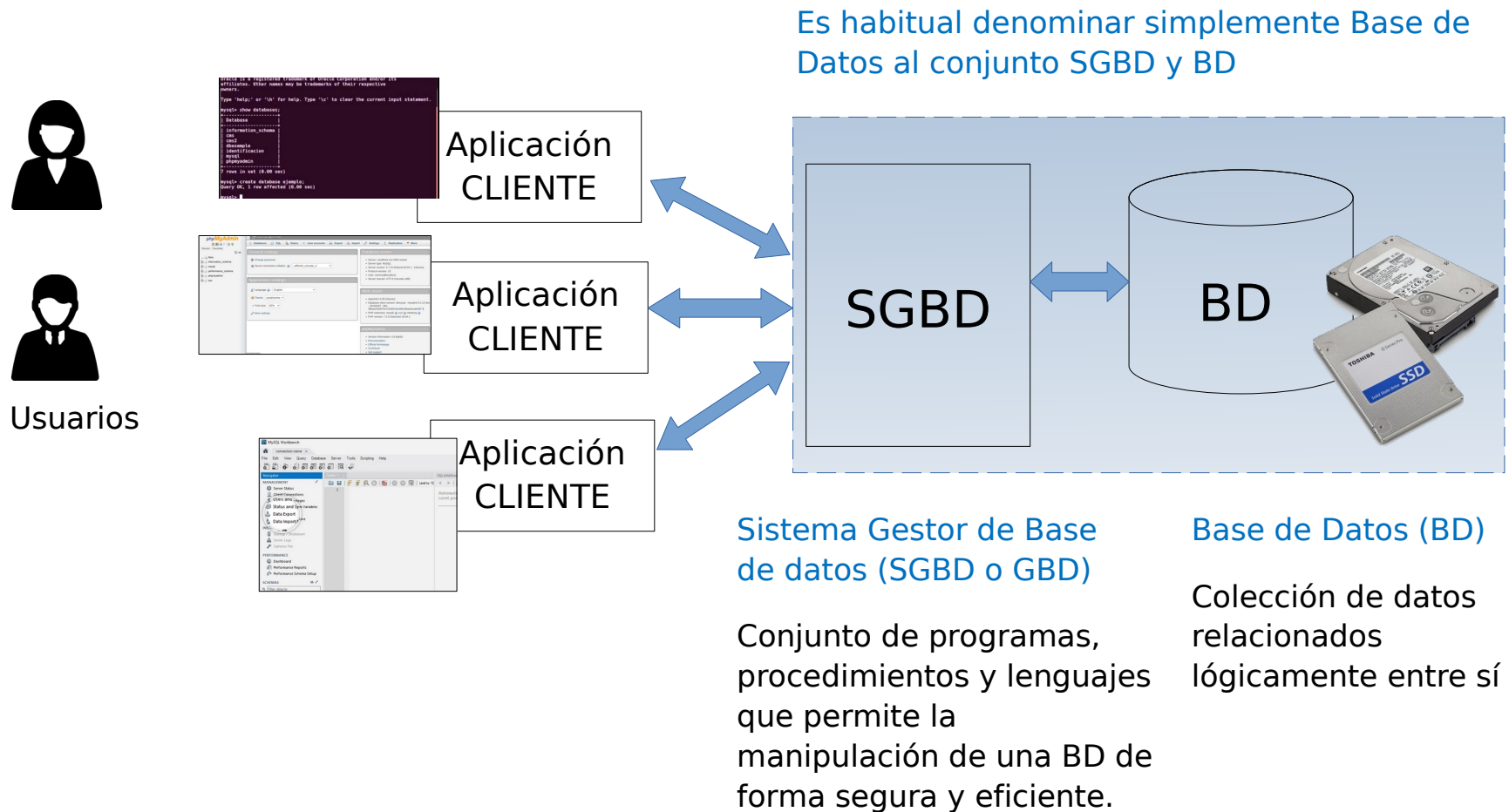
## Unidad de trabajo 1

### Almacenamiento de la Información

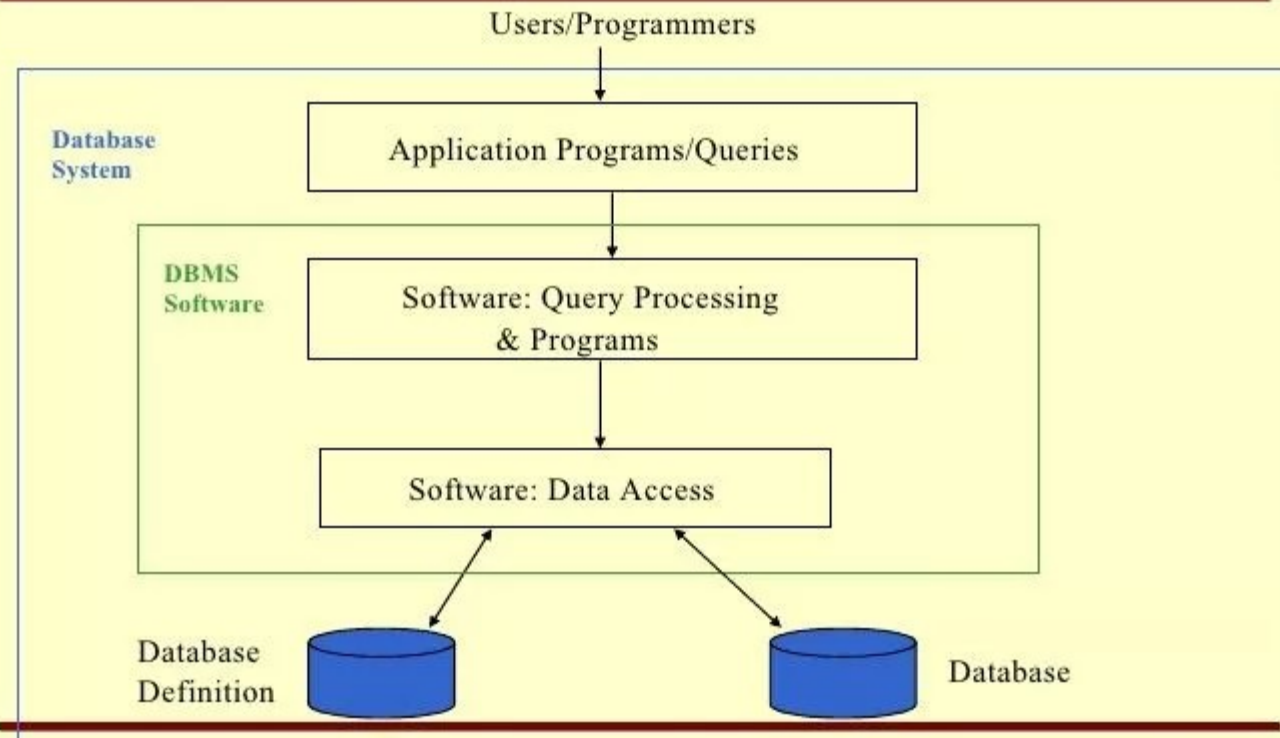
#### → Base de Datos

- Base de Datos y Sistema Gestor de Base de Datos
- Definición de BD
- Modelos de Base de Datos
- Ejemplo NoSQL: Big Table

# Base de datos y Sistema Gestor de Base de Datos



# Database System Environment



# Definición de Base de Datos

**Base de Datos (BD):** Es una colección de datos relacionados lógicamente entre sí.

Características:

- Estructura determinada. Existen varias estructuras según el **modelo** de BD.
- Definición y descripción comunes
- Almacenamiento con la mínima redundancia
- Posibilidad de acceso eficiente por parte de usuarios y aplicaciones

La base de datos no sólo contiene los datos de la organización, también almacena una descripción de dichos datos. Esta descripción es lo que se denomina metadatos y se almacena en el **diccionario de datos** o **catálogo**.

**Modelo lógico de base de datos:** determina la **estructura lógica** de una base de datos y el modo de almacenar, relacionar, organizar y manipular los datos.

El modelo de base de datos NO determina cómo se visualizan los datos (listado, tabla, gráfico...), ya que eso depende de la apariencia de la capa de visualización (nivel externo o de visión en la arquitectura del SGBD) o de la aplicación cliente que utilice la base de datos.

A continuación se describen algunos modelos de base de datos:

- Modelo jerárquico o en árbol
- Modelo en red
- Modelo relacional
- Modelo multidimensional
- Modelo orientado a objetos y objeto-relacional
- Modelo NoSQL (no relacional)

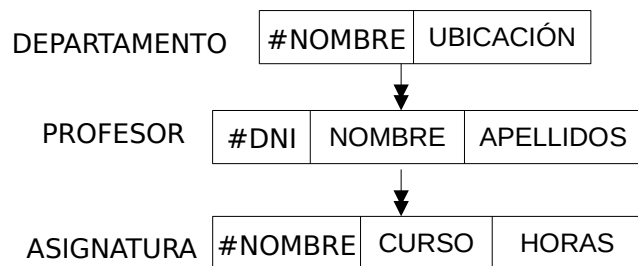
## → Modelo jerárquico o en árbol

Los datos se estructuran en nodos o segmentos que contienen atributos o campos.

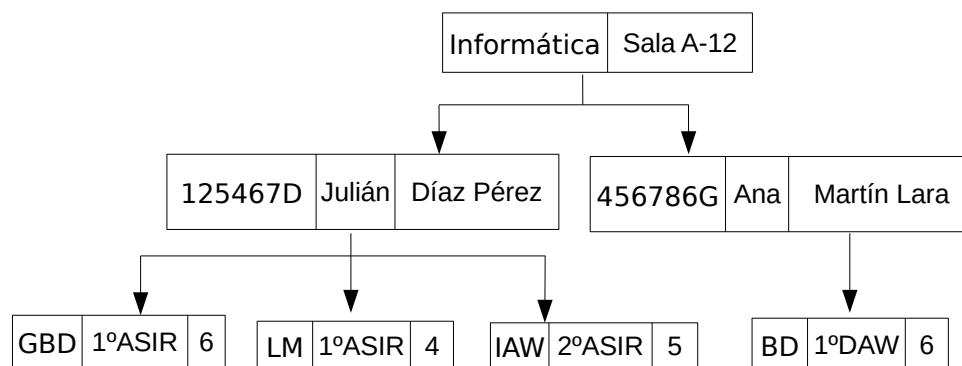
Los nodos se relacionan entre sí mediante jerarquías padre/hijo. Cada nodo tendrá un solo padre pero puede tener varios hijos.

La forma visual de este modelo es de árbol invertido, en la parte superior están los padres y en la inferior los hijos.

### ESTRUCTURA LÓGICA



### EJEMPO DE BASE DE DATOS



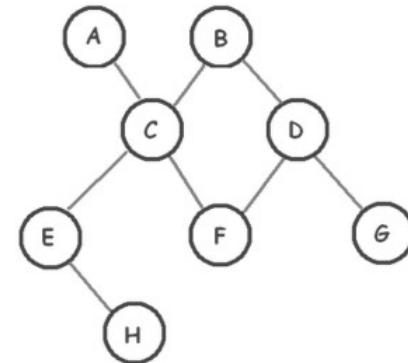
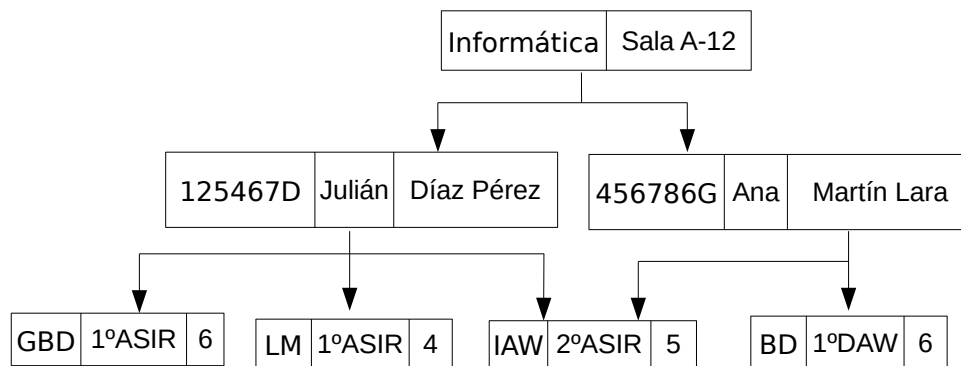
Es limitado en cuanto a los datos que puede representar (por ejemplo no podría representarse el que una misma asignatura sea impartida por varios profesores)

## → Modelo en red

Permite representar relaciones más complejas que el modelo jerárquico, ya que un elemento puede tener más de un padre. En contrapartida, el manejo puede ser más complicado.

Los registros se denominan nodos, que se relacionan con otros nodos mediante enlaces.

### EJEMPLO DE BASE DE DATOS



## → Modelo relacional

La base de datos es percibida por el usuario como un conjunto de tablas. Esta percepción es sólo a nivel lógico, ya que a nivel físico puede estar implementada mediante distintas estructuras de almacenamiento. Los datos son almacenados en tablas bidimensionales, llamadas relaciones.

Se llamará registro, entidad 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 una tupla. Requiere **normalización** de la información.

Se emplea el lenguaje **SQL**, un estándar implementado por los principales SGBD relacionales.

CLIENTE

Nombre	Apellido	DNI	Teléfono	Dirección
Julián	Pérez	45863217L	610588963	Calle Mayor 32
Andrea	Cosido	45234523M	678500145	Calle Cuesta 4

PRODUCTO

Referencia	Descripción	Precio
314	Cemento blanco	33.25
502	Lámina roble	12.5
503	Lámina pino	12.5
021	Polímero básico	11

COMPRAS

Referencia	DNI	Unidades
503	45863217L	5
503	45234523M	10
021	45863217L	7

Este modelo es muy utilizado hoy en día. Se profundizará más adelante.

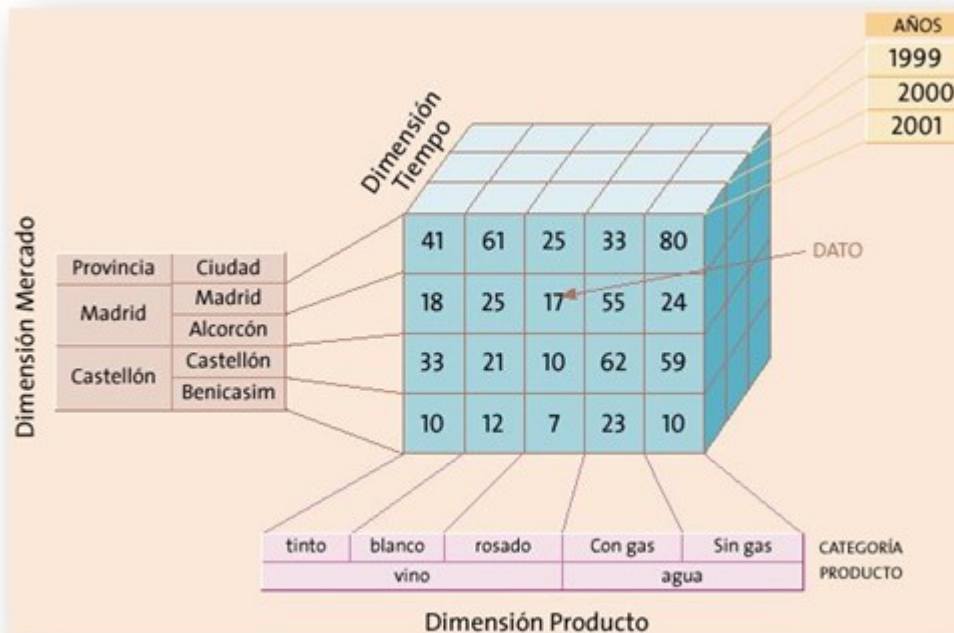


## → Modelo multidimensional

Se utilizan para aplicaciones muy específicas que requieren consultas complejas y alto rendimiento. Similar al modelo relacional, los datos se representan en tablas multidimensionales denominadas cubos, aunque realmente serían hipercubos (cubo de n dimensiones).

Cada dimensión o eje representa un campo o una métrica.

Un ejemplo de 3 dimensiones:

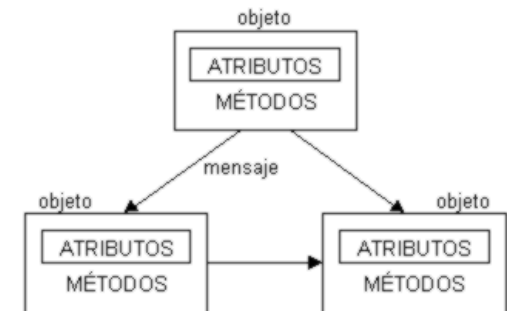


Empleado en sistemas OLAP (Procesamiento analítico en línea, On-Line Analytical Processing):

Tecnología que organiza grandes bases de datos empresariales y proporciona análisis complejo.

## → Modelo orientado a objetos

Desarrollada a partir de la programación orientada a objetos (POO u OOP). Define objetos con propiedades y métodos (operaciones). Los objetos con la misma estructura y comportamiento pertenecen a una clase y las clases se organizan en jerarquías.



Elimina limitaciones del modelo relacional. Gracias a este modelo se incorporan mejoras como la herencia entre tablas, los tipos definidos por el usuario e integra elementos multimedia (datos gráficos, imágenes, voz y texto) de manera natural. Esta estructura tiene gran difusión en aplicaciones web.

Se profundizará más adelante.

## → Modelo objeto-relacional

La base de datos objeto-relacional es una extensión de la base de datos relacional tradicional, a la cual se le proporcionan características de la programación orientada a objetos.

Incorpora herencia a las tablas, concepto derivado de la orientación a objetos.

Las tuplas (registros) pueden ser a su vez relaciones (tablas). Posibilita guardar objetos más complejos en una sola tabla con referencias a otras.

## → Modelo NoSQL (No relacional)

Se emplean cuando hay un gran volumen de datos y se requiere que el tiempo de respuesta sea corto. Suelen ser bases de datos distribuidas.

Relacional	No relacional
La estructura de la BD está claramente definida empeando tablas y registros. Datos normalizados y del mismo tipo.	Datos no estructurados o semiestructurados. Los datos no tienen por qué estar relacionados ni normalizados. Son heterogéneos.
Lenguaje SQL	No utiliza lenguaje SQL, aunque en ocasiones puede soportarlo ("Not only SQL"). Tienen sus propios lenguajes.

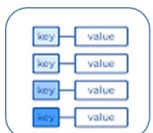
Existen varios modelos para almacenar y acceder a los datos:



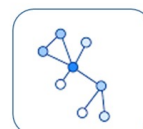
Almacén de documentos (document storage): los datos y los metadatos se almacenan jerárquicamente en documentos basados en JSON



Almacén de columna ancha (wide-column storage): existen tablas pero los campos y formatos pueden variar de una fila a otra.



Almacén de clave-valor (key-value storage): los datos son una pareja clave-valor



Almacén de grafos (graph storage): los datos son nodos con propiedades



## ¿Cómo almacena Google los datos?

La empresa Google almacena una gran cantidad de información recopilada de todas sus aplicaciones. Toda esa información se genera a una gran velocidad y está representada por datos muy heterogéneos.

Si se empleara una base de datos relacional no sería posible dar una respuesta rápida, ya que se realizarían chequeos para no violar las restricciones del modelo relacional. Además, obligaría a establecer de antemano el formato de los datos, y en ocasiones, como por ejemplo para guardar la actividad en una web de un usuario, la información se genera en tiempo real sin ninguna estructura preestablecida (clicks, movimiento de ratón, interacción con elementos dinámicos...).



**BigTable** es una base de datos **NoSQL** propietaria de Google que está montada sobre en GFS (Google File System), un sistema **distribuido** de ficheros en el que los ficheros están repartidos en múltiples máquinas (nodos). Los datos no están duplicados en Bigtable, pero el almacenamiento se realiza a través de ficheros (SSTables) en GFS, dónde sí se hacen **réplicas de los datos** para garantizar la **durabilidad**.

Es utilizado por Google para dar servicio a sus aplicaciones, como por ejemplo a Google Earth o Google Analytics, entre otras muchas. **Cloud BigTable** es un servicio que ofrece Google y que se puede contratar por otras empresas para almacenar sus datos.

Tiene la capacidad de guardar y acceder a enormes cantidades de datos a gran velocidad. Además **es escalable** pudiendo variar dinámicamente el número de nodos de la arquitectura sin afectar al servicio.



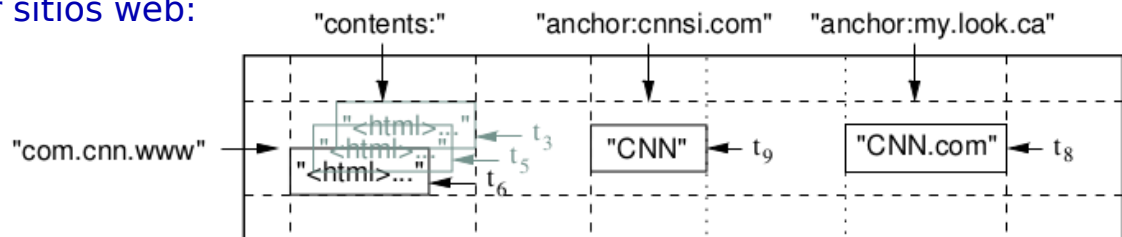
El **modelo de datos BigTable** es de tipo NoSQL **Wide-column storage**.

A grandes rasgos se basa en la definición de tablas multidimensionales, en las cuales se cargan celdas de **información no interpretada por el sistema**.

Una tabla se compone de miles de millones de filas y miles de columnas, identificadas con strings y que contienen celdas con los datos. En cada celda se pueden almacenar diferentes versiones de la información, diferenciadas por la marca de tiempo en que fueron creadas, aunque la aplicación cliente puede definir esa marca con otros valores que le convengan.

Por ejemplo, BigTable para almacenar sitios web:

- Clave de cada fila: URL de la web.
- Nombre de columna:  
elemento HTML (enlace,  
cabecera, contenido HTML...)



- El dato en cada celda tendría varias versiones, según la arca de tiempo.

La unidad básica de acceso a las tablas sin embargo no son filas y columnas, sino tablets (conjunto o rango de filas) y familias (conjunto de columnas). Si se elige el nombre de las filas de forma que las que sean consecutivas tengan algún tipo de relación lógica en sus datos, por ejemplo cercanía geográfica en caso de Google Earth, el sistema se optimiza minimizando los accesos.

**No se realizan chequeos de coherencia de los datos.** Es la aplicación cliente quien realiza el razonamiento de las relaciones entre ellos, que son transparentes para el sistema.