

Introducción a los Sistemas de Bases de Datos

Historia de las BD

Sistemas de archivos:

Surgieron debido a la necesidad de almacenar los datos para su correspondiente reuso (**persistencia**)

Desventajas:

- **Redundancia de datos**
- Dificultad de integración: **Los archivos no se pueden “combinar” fácilmente → diferentes formatos y estructura**
- Alto costo para la propagación de cambios
- Alto costo si se cambiaba la estructura de un archivo
- Riesgo de inconsistencias debido a actualizaciones simultáneas

Muchas aplicaciones usaban sus propios archivos

Informes que requieren datos de diferentes archivos

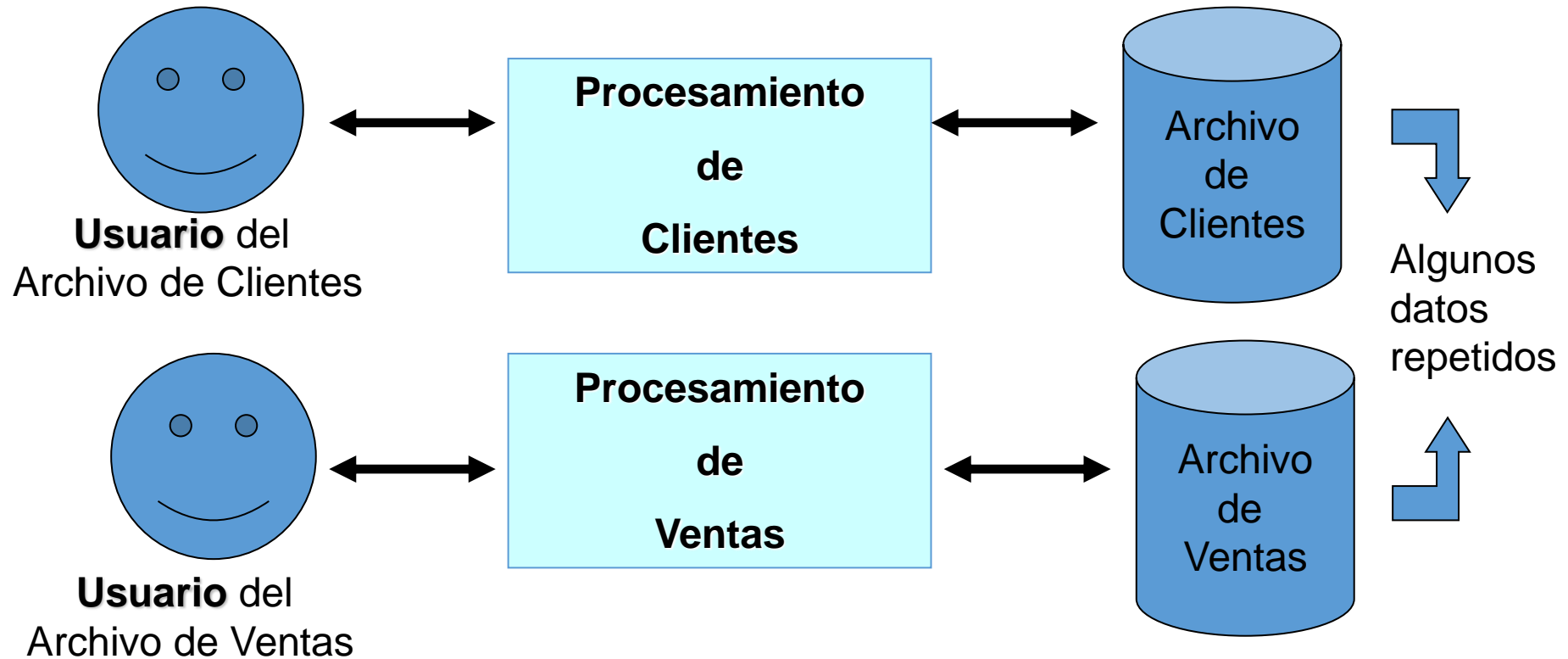
Historia de las BD

- Por ejemplo, suponga que usted tiene datos de **empleados:** **cédula, nombre, dirección, teléfono** y de **proyectos: nombre, valor, fecha de inicio**. Además, un empleado puede estar asignado a varios proyectos y en un proyecto pueden trabajar varios empleados. ¿Cómo guardaría usted estos datos? ¿En varios archivos? ¿En uno solo?
- En clase se discuten algunos aspectos sobre este ejemplo o sobre uno similar

Historia de las BD

- Note que un diseño físico (distribución de los datos en uno o varios archivos) será posiblemente beneficioso para ciertas aplicaciones (por ejemplo, para ciertas consultas) y será posiblemente desventajoso para otras...

Historia de las BD



La dificultad para combinar archivos (diferentes formatos y estructura) promovía la redundancia de datos

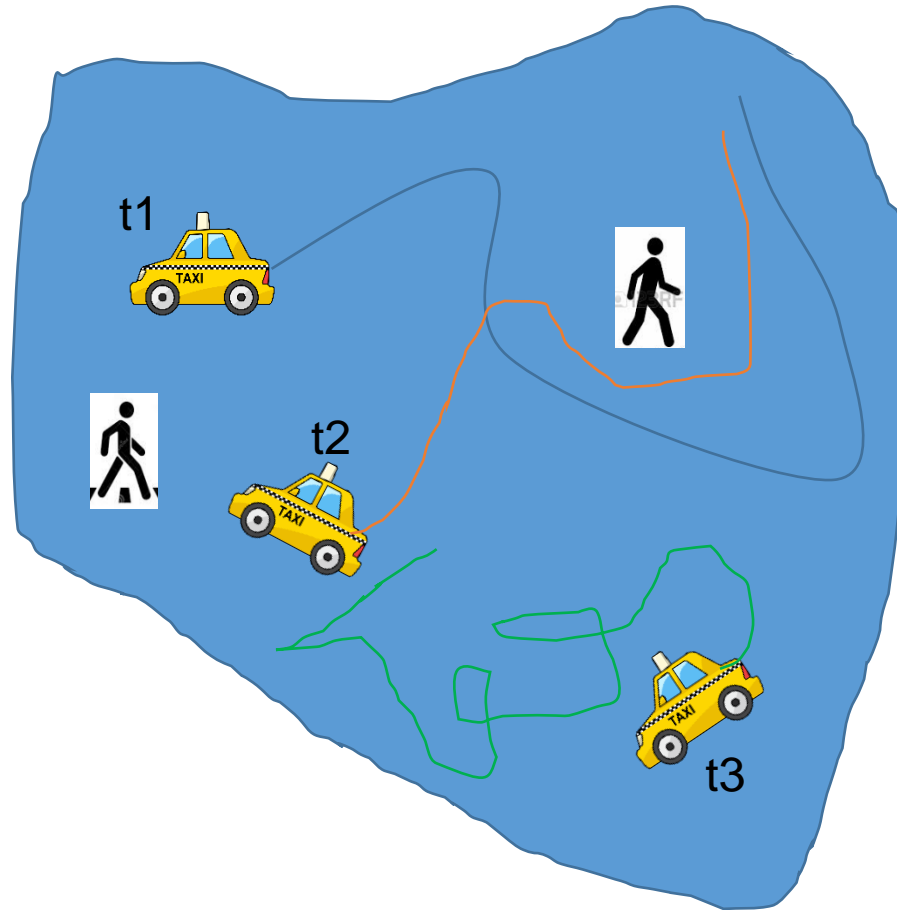
Historia de las BD

Sistemas de BD:

- Surgieron para solucionar los problemas de los sistemas de archivos
- Capacidades:
 - Manejo de **persistencia**
 - Soporte por lo menos de un **modelo de datos**
 - Soporte de un **lenguaje de alto nivel** que permita manipular y definir la estructura de los datos en forma eficiente → **SQL**
 - Control de acceso → **Seguridad**
 - **Evitar inconsistencias** al compartir los datos

Definir usuarios, roles, permisos...

Un ejemplo



Taxi	Hora	X	Y
t1	8:25 am	100	50
t1	8:30 am	80	42
...			
t2	6:45 am	99	48
t2	6:50 am	75	40
...			

El usuario pide un taxi por medio de una aplicación en su celular, el sistema detecta su posición, envía el taxi más cercano y los datos del taxi: conductor, placas...

Definiciones

- **Base de Datos (BD)**: Conjunto de datos (con su respectivo **esquema**) almacenados en algún medio (por ej. un disco) y usados con diversos propósitos por múltiples usuarios
- **Esquema de la BD**: Describe la estructura de los datos requeridos por la organización.
Esta estructura suele permanecer estática durante un gran período
- **Instancia de la BD**: Son los datos que en un determinado instante posee la BD.
Estos datos suelen cambiar (adiciones, borrados, actualizaciones) constantemente (¿excepto en que tipo de BD?)

Definiciones

- **Usuario final:** El que interactúa con la BD, usualmente a través de aplicaciones e interfaces
- **Usuario especialista:** El que diseña y programa aplicaciones para los usuarios finales
- **DBA (*DataBase Administrator*):** El que administra la BD → ¿Funciones?

SGBD

- **Sistema de Gestión de Bases de Datos (SGBD*):**
Sistema computarizado que permite la gestión de las BD
 - Es un conjunto de programas
 - Sirve de interfaz entre los usuarios, los datos y programas de la BD
 - Interactúa con el sistema operativo
 - Ejemplos: Oracle, SQL Server, DB2, PostgreSQL.

*DBMS: *Data Base Management System*

SQL

SGBD

- Soporta DDL (*Data Definition Language*): Lenguaje para la creación del **esquema**
- Soporta DML (*Data Manipulation Language*): Lenguaje para la inserción, actualización, borrado y consulta de datos
- Soporta DCL (*Data Control Language*): Lenguaje para la gestión de usuarios, roles, permisos, entre otros.

SQL incluye estos (sub) lenguajes

SGBD

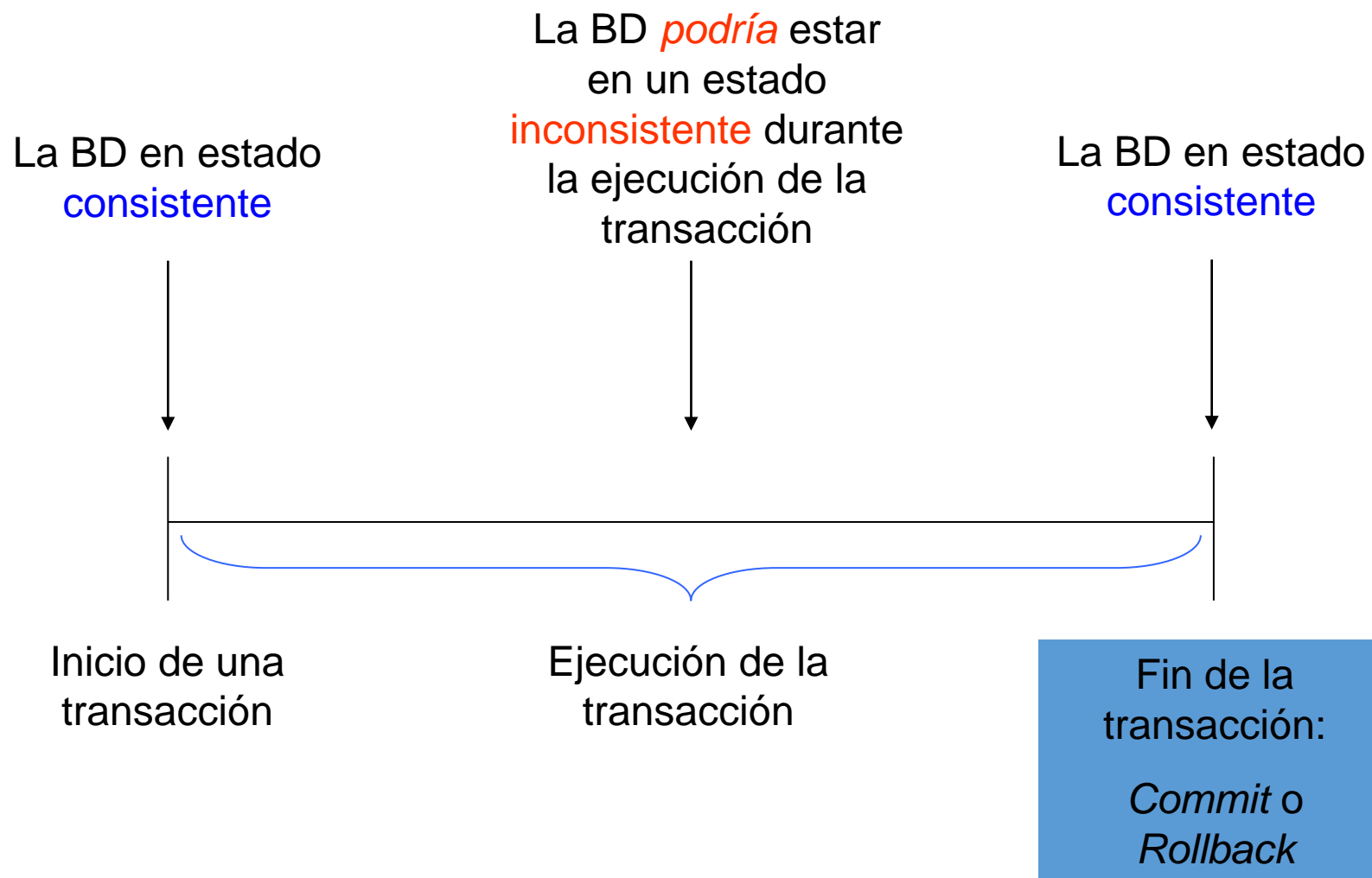
- Gestión de transacciones: una transacción es una unidad de trabajo que suele incluir varias operaciones sobre la BD (consultas, actualizaciones, borrados, ...)
- Recuperación ante fallas y para deshacer (*rollback*)
transacciones: Bitácora (*log* de transacciones)
- Manejo de respaldos (*backups*)
- Independencia de los datos (se explica posteriormente)
- Otros servicios

SGBD

Un SGBD debe garantizar las propiedades ACID para las transacciones:

- *A*tomicity (Atomicidad)
- *C*onsistency (Consistencia)
- *I*solation (Aislamiento)
- *D*urability (Durabilidad)

SGBD



SGBD

- Bloqueos: compartido (**s**) y exclusivo (**x**).

Matriz de bloqueos

			Transacción 1 tiene:
	x	s	
x	No	No	
Transacción 2 solicita:		Sí	

Quando una transacción termina (ya sea con *commit* o *rollback*) se liberan sus bloqueos

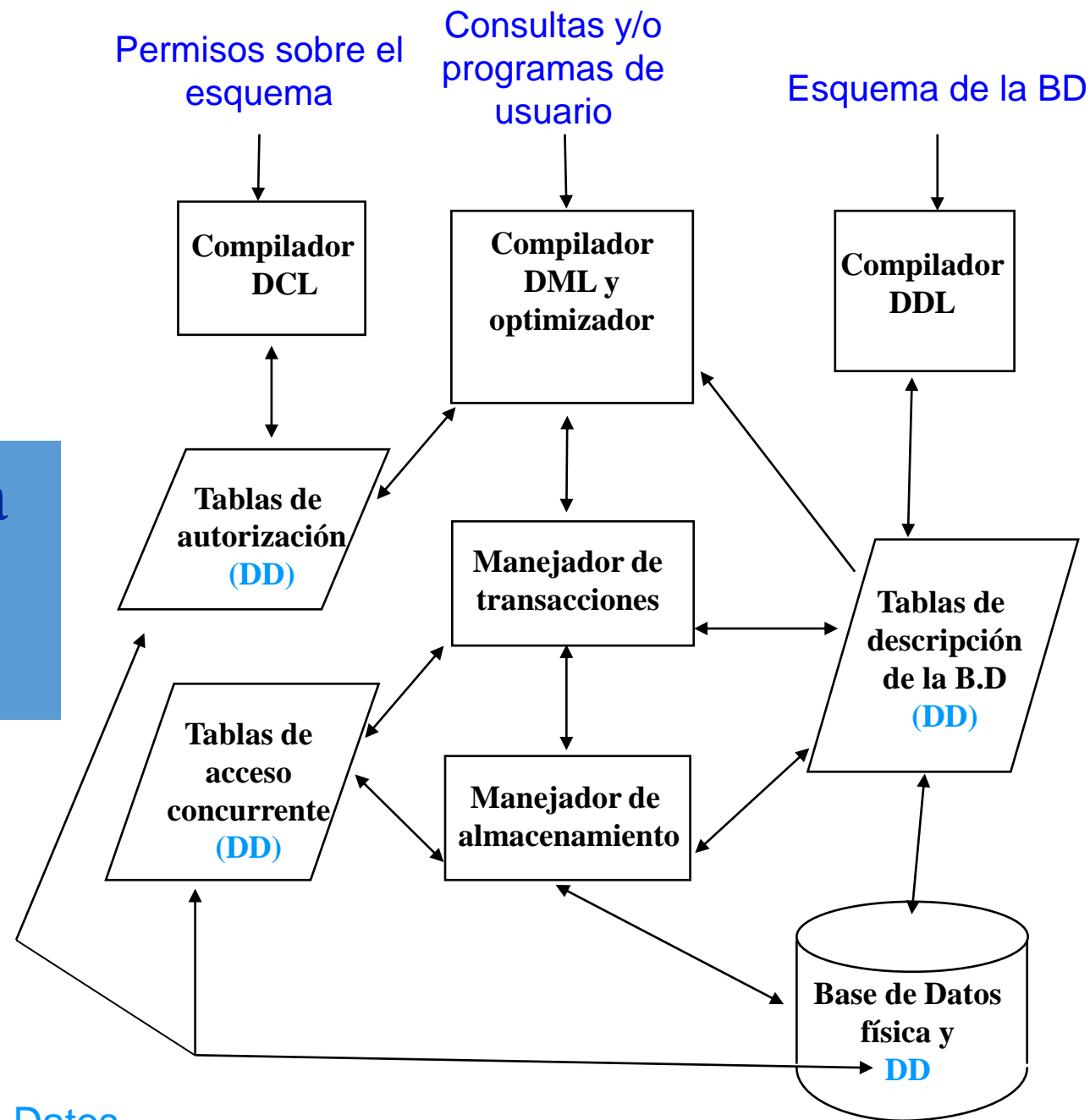
SGBD

Ejemplo: Saldo de la cuenta #8: \$100.

Analizar la siguiente situación, donde la transacción 1 hace *commit* y la transacción 2 hace *rollback*.

Transacción 1	Tiempo	Transacción 2
	t1	Sumar \$50 al saldo de la cuenta #8
Considerar dos casos: a) Leer el valor de la cuenta #8 b) Sumar \$20 al saldo de la cuenta #8	t2	
	t3	<i>Rollback</i>
<i>Commit</i>	t4	

Arquitectura de un SGBD



DD = Diccionario de Datos

Bases de Datos

SGBD

- **Diccionario de Datos (DD):**
 - Contiene datos sobre el esquema de la BD, los usuarios, los permisos de acceso, etc.
 - Almacena los datos que permiten la traducción entre los tres niveles* de la arquitectura ANSI-SPARC (*American National Standards Institute, Standards Planning And Requirements Committee*).
 - Es un catálogo autodescriptivo, datos sobre los objetos existentes en el sistema → **Datos sobre los datos** (Metadatos).
- **Optimizador de consultas:** Define el plan de ejecución de las operaciones solicitadas por los usuarios, de tal manera que se ejecuten de manera eficiente.
- **Manejador de transacciones:** Controla el acceso y la concurrencia de operaciones.

*Se ven posteriormente.

SGBD

- **Manejador de almacenamiento:** Con dos componentes:
 - El manejador de archivos: Recupera desde disco los bloques que contienen los datos solicitados por una transacción.
 - El manejador de *buffer*: Mantiene en memoria principal los datos más usados y decide cuando llevar a disco alguno de sus bloques.

Ventajas de un SGBD

- Reuso de datos y programas
- Control de redundancia
- Estandarización ¿en qué sentido?
- Concurrencia
- Es posible equilibrar las cargas de los requisitos (establecer prioridades)

Ventajas de un SGBD

- Integridad (Se cumplen las reglas establecidas → ¿Qué tipo de reglas? ¿Establecidas por quién?)
- Seguridad
- Rapidez de desarrollo
- Mantenimiento y reingeniería: cambios en el esquema sin cambiar los programas que los usan (hasta cierto punto)

Desventajas de un SGBD

- Tamaño
- Susceptibilidad a fallas (*Discutible*)
- Complejidad en la recuperación a fallas (*Discutible*)
- Lentitud debido a la cantidad de **verificaciones** que debe hacer



¿Cómo cuáles?

BD especializadas

- ⇒ BD para la toma de decisiones (*Data Warehouse*).
- ⇒ BD distribuidas, paralelas, *blockchain*.
- ⇒ BD deductivas, temporales, multimediales (imágenes, audio, videos) → consultas sobre estos tipos de datos.
- ⇒ BD orientadas a dispositivos móviles.
- ⇒ BD geográficas: Sistemas de Información Geográficos (SIG), *trayectorias*.
- ⇒ BD nativas para XML, JSON.
- ⇒ Big Data: BD NoSQL (*Not only SQL*) → No necesariamente garantizan ACID.