

# Bases de Datos I



## Introducción



# BASES DE DATOS

La tecnología de BD es un componente esencial de casi todos los sistemas de información “tradicionales” y aplicaciones más recientes



- Sistemas de bancos, compras online, sistemas de salud, reservas (hoteles, vuelos, bibliotecas, etc.)
- BD Multimedia (almacenamiento de imágenes, videos, animaciones)
- Sistemas de Información Geográfica (GIS)
- BD de gran capacidad, sistemas de almacenamiento big data, sistemas No-SQL usados en redes sociales, búsquedas web, etc.
-

# ¿Qué es una BASE DE DATOS?

**Conjunto de datos almacenados, representando entidades y sus relaciones, lógicamente coherente** (no hay contradicción entre los datos relacionados ni pérdida de información, aún ante uso compartido), **con significado implícito** (representa aspectos del “universo de discurso”, Datos + contexto = Información) **y persistente** (los datos permanecen en el tiempo)

**Una BD es diseñada, construida y poblada con datos para un propósito específico**



**Contiene información relevante para una organización**  
que es clave para la correcta toma de decisiones



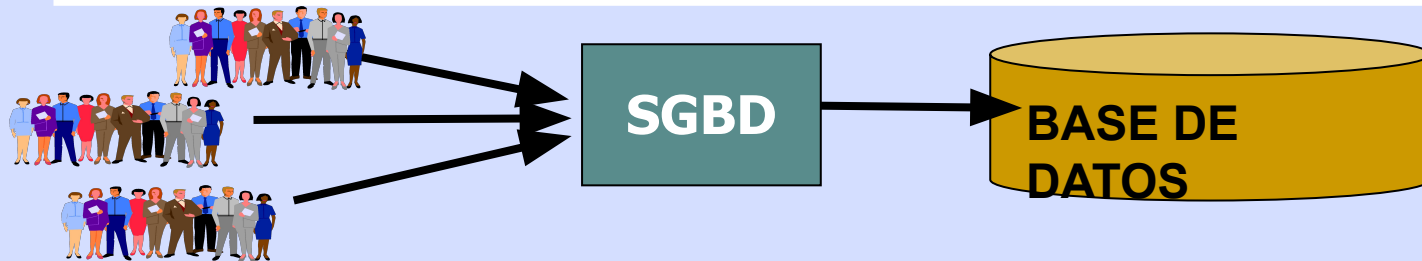
# ¿Por qué surgen las BD?

## Debilidades de los **sistemas de gestión de archivos (SGA)**:

- demasiado elementales para el tratamiento de los datos y sus interrelaciones, puede haber redundancia e inconsistencia de datos, problemas en la actualización simultánea por varios usuarios, etc.
- utilización mediante programas de aplicación demasiado compleja

**impulsaron el desarrollo de **software más sofisticado**:**

### **Sistemas de Gestión de Bases de Datos (SGBD)** o *DataBase Management System (DBMS)*



- Permiten definir, almacenar y recuperar datos de la BD de manera eficiente
- Los mismos datos son compartidos por diferentes usuarios/aplicaciones

# Historia de los sistemas de BD

## 1º Generación: Sistemas en red y jerárquicos

- A comienzos de los '60 Bachman creó el 1º DBMS en red (*Integrated Data Store*)
- IBM desarrolló un sistema jerárquico (*Information Management System*)
- Independencia de datos mínima, acceso complejo a los datos

## 2º Generación: modelo relacional

- 1970: Codd (IBM) propone el *Modelo de Datos Relacionales*
- 1976: Chen presentó el Modelo de Entidades-Relaciones
- Amplio desarrollo de SGBD relacionales comerciales (DB2 de IBM, Oracle, Informix)
- Lenguaje de consultas estructurado (SQL) estandarizado hacia fines de los '80
- Capacidad limitada para modelar datos

## 3º Generación: modelo relacional extendido y orientado a objetos

- Mejoras sobre el modelo relacional puro y lenguajes de consulta
- Extensión de capacidades, tipos de datos y aplicaciones



# Características de los SGBD

Brindan capacidad y soporte para:

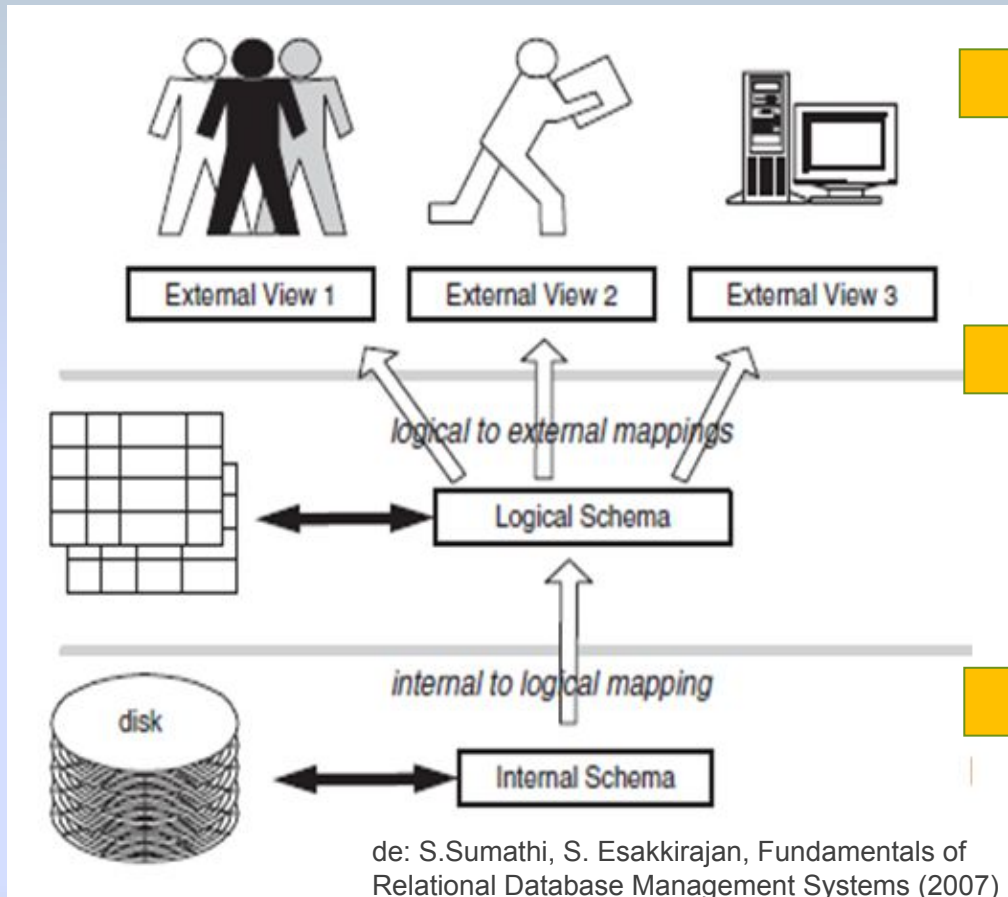
- **Control de redundancia** (para modelar relaciones entre los datos y mejorar las prestaciones) y **evitan riesgo de inconsistencias** al compartir la información
- **Seguridad y control de acceso** (los datos pueden ser compartidos, con posibilidad de distintos permisos a usuarios autorizados)
- **Integridad de datos:** se define la validez de los datos, mediante la especificación de restricciones o reglas
- **Concurrencia y recuperación:** gestión de acceso por distintos usuarios al mismo tiempo sin interferencias y se garantiza la recuperación ante fallas
- **Accesibilidad a los datos:** lenguaje para definición de datos y restricciones, consulta y actualización ocultando detalles de cómo se almacenan y actualizan

Requieren:

- **Inversión inicial** (en software, hardware, training, migración): el costo varía, aunque hay SGBD open source completos y versátiles (ej: PostgreSQL)
- **Comprender su alcance** para mayor aprovechamiento de sus posibilidades

# Arquitectura en 3 Niveles

(según ANSI/SPARC - American National Standards Institute, Standards Planning And Requirements Committee)



## Nivel externo

Descripción de vistas de usuario  
Puede haber tantos subesquemas de la BD como tipos de usuarios finales

## Nivel conceptual

Descripción semántica de los datos, sus relaciones y restricciones

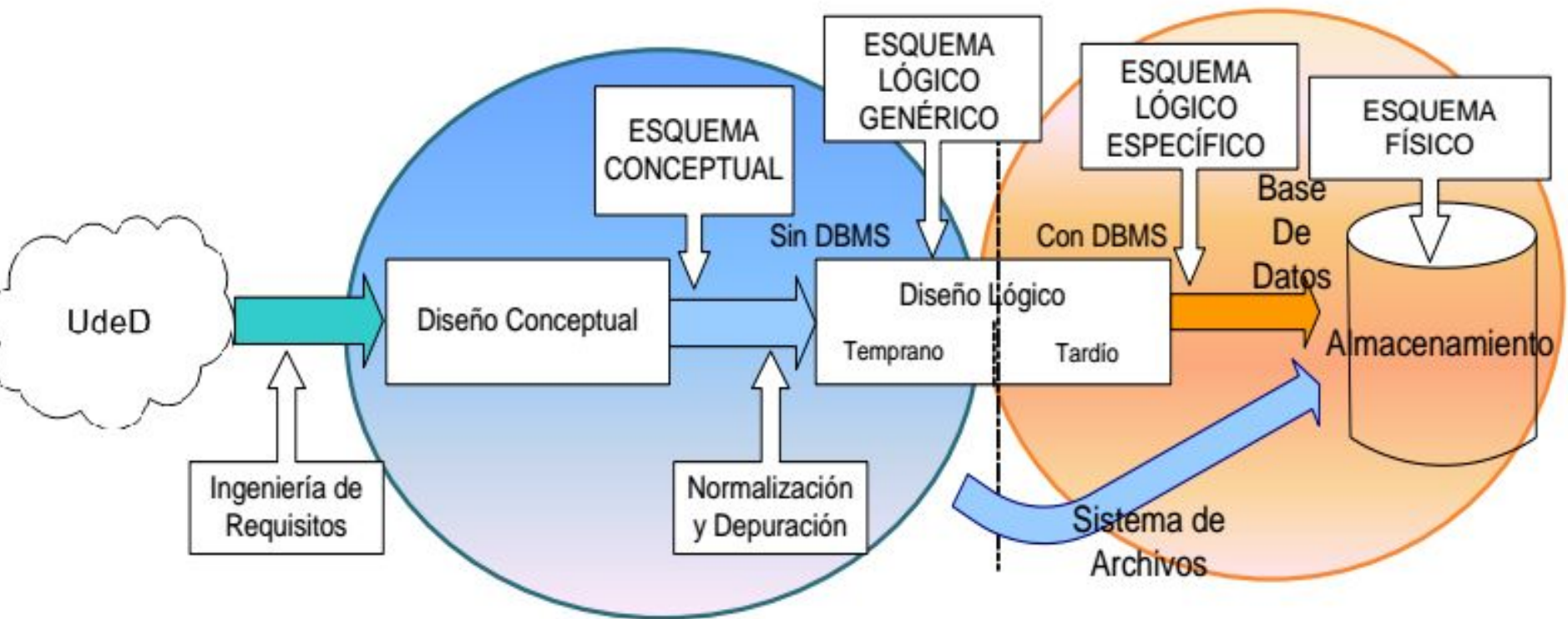
Algunos distinguen un **Nivel Lógico**: descripción según modelo de datos del SGBD

## Nivel interno o físico

Descripción y tipos de datos según un SGBD particular: tamaño y precisión, tipos de índices y de estructuras de almacenamiento utilizadas

Los niveles de abstracción facilitan la **independencia de los datos** entre un nivel y el superior

# Obtención del esquema de una BD





# Roles en una base de datos

## Administrador de la base de datos (DBA):

- Tiene control central sobre datos y programas de acceso
- Responsabilidad del manejo de aspectos técnicos de la BD
- Se encarga de mantener la integridad y disponibilidad de datos, control de seguridad y concurrencia, mantenimiento del sistema y prestaciones

## Diseñadores de la base de datos:

- Diseño lógico de la BD: relevamiento e identificación de datos, relaciones y restricciones
- Diseño físico de la BD: decidir la implementación física a partir del diseño lógico



## Programadores de aplicaciones:

- Implementan por medio de programas las especificaciones que darán servicio a los usuarios y los mantienen

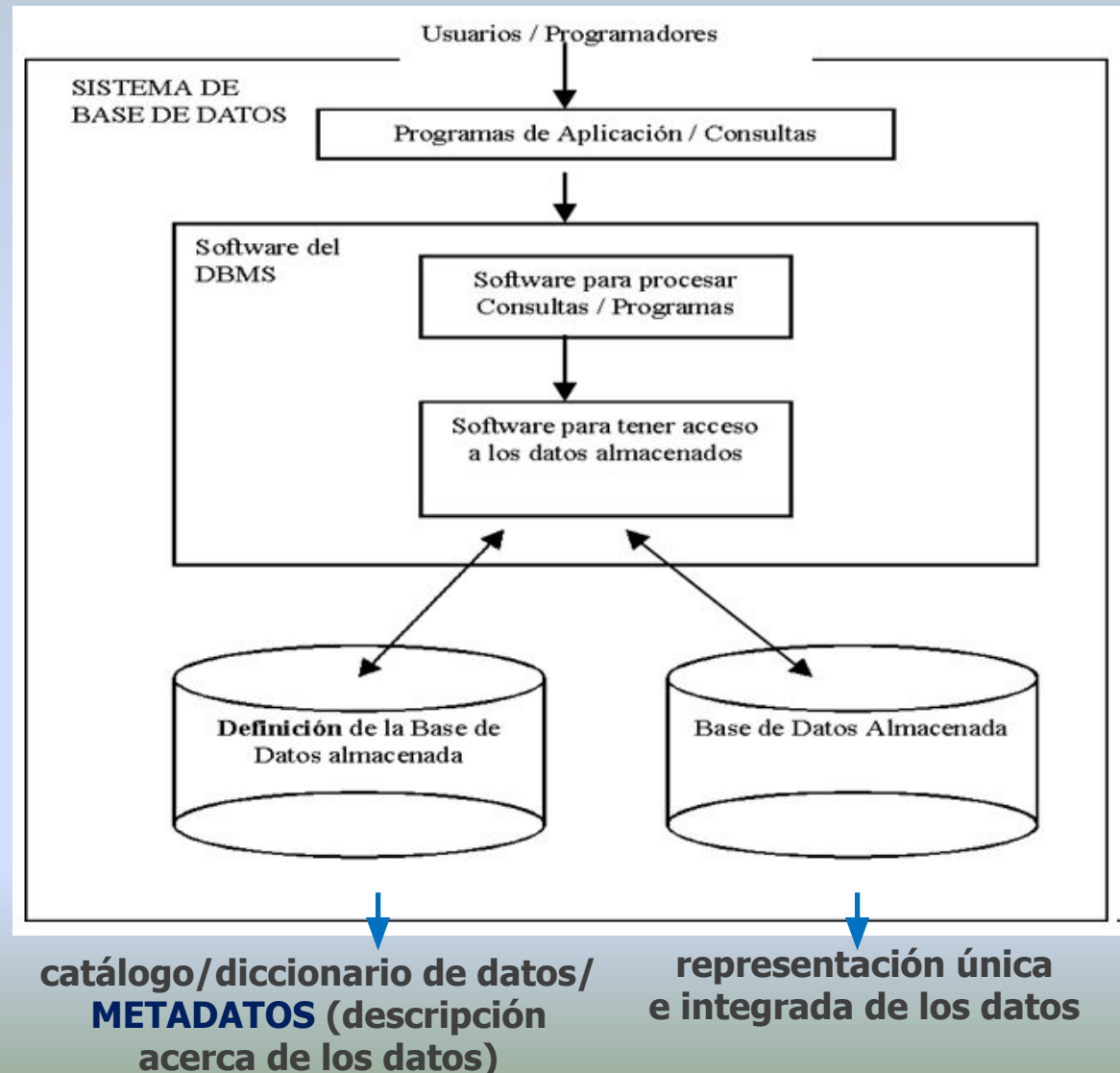
## Usuarios finales:

- Clientes de la base de datos (consultan, actualizan,...)

# Esquema de acceso a los datos

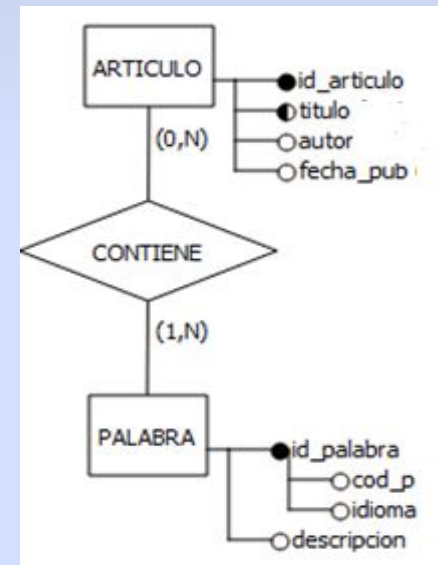


El SGBD transforma una solicitud de usuario (*esquema externo*) a una solicitud expresada según el *esquema conceptual* y garantiza el acceso a los datos almacenados (*esquema interno*)



# Modelo de Datos

- Colección de herramientas conceptuales para describir los datos, sus relaciones, semántica en el UdeD y restricciones a sus valores
- **Modelo de datos relacional:** uno de los modelos más populares, subyacente a la mayoría de las BD actuales
- Modelo o Diagrama de **Entidades-Relaciones (Extendido) - DEREExt:** Representación gráfica de entidades, atributos, relaciones y restricciones
- Derivación a modelo relacional según **Reglas de transformación**
- Existen otros modelos de datos: **orientado a objetos, objeto-relacional, post-relacional**





# Structured Query Language (SQL)

Es un lenguaje para **definición** y **manipulación** de datos

- desarrollado inicialmente en laboratorios de investigación de IBM
- se transformó en estándar en 1986 (SQL-86) y ha tenido numerosas revisiones
- SQL:1999 incorporó triggers y características OO (SQL3)
- Nuevas versiones: incorporaron nuevas características, no sustancialmente diferentes de SQL3

Es **Declarativo**: se indica qué datos se requieren, sin especificar cómo

- **Lenguaje de Definición de Datos (DDL)**: permite crear y modificar el esquema de la base, tablas, restricciones, vistas, etc.
- **Lenguaje de Manejo de Datos (DML)**: permite consultar y actualizar datos (inserción, modificación, eliminación)

**Otras posibilidades:**

- **SQL empotrado**: Manipulación de datos empotrada desde un lenguaje anfitrión
- **SQL Procedural**: Lenguaje de programación para escribir PSM (Persistent Stored Modules): Triggers, Funciones y Stored Procedures
- **Lenguajes de 4º Generación (4GL)**: Generadores de formularios, informes, gráficos

# SQL



## Structured Query Language (SQL)

### Algunas sentencias SQL:

- **Definición de Datos (DDL):**

**CREATE TABLE** <nom\_tabla> (.... ); → **creación de una tabla**

**ALTER TABLE** <nom\_tabla> ... ; → **modificación de una tabla**

**DROP TABLE** <nom\_tabla> ; → **eliminación de una tabla**

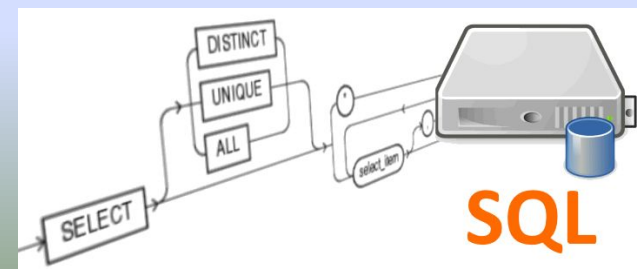
- **Manejo de Datos (DML):**

**INSERT INTO** <nom\_tabla> (...) **VALUES** (.... ); → **inserción de tuplas**

**UPDATE** <nom\_tabla> **SET** (...) **WHERE** (.... ); → **modificación de datos**

**DELETE FROM** <nom\_tabla> **WHERE** (.... ); → **borrado de tuplas**

**SELECT** <lista\_atrib> **FROM** <lista\_tablas>  
**WHERE** (.... ); → **consulta de datos**





# BIBLIOGRAFÍA

**Date, C., "An Introduction to Database Systems". 7º ed., Addison Wesley, 2000**

**Elmasri, R., Navathe, S., "Fundamentals of Database Systems", Addison Wesley, 2011 (también: de Pearson, 2016, 7º ed.)**

**Silberschatz, A., Korth, H, Sudarshan, S., "Database System Concepts", 6º ed., WCB/McGraw Hill, 2010**

**Sumathi S., Esakkirajan S., "Fundamentals of Relational Database Management Systems", 2007**

