

Unidad de Trabajo 1
Sistemas de almacenamiento de la información

Sistemas de almacenamiento de la información

IES Palomeras Vallecas

Curso 2020/2021

Profesor: Alberto Ruiz

Conceptos fundamentales

- Vamos a trabajar con los conceptos de sistemas de información y bases de datos
- Explicaremos los conceptos de:
 - Información
 - Datos
 - Sistema
 - Sistema de información
 - Base de datos
 - Sistema gestor de bases de datos (SGBD)

Datos vs Información

- **Información:** conocimiento adquirido a través de la observación y el análisis de lo que percibimos a nuestro alrededor
 - La observación por sí misma no genera información
 - Es necesario analizar lo que percibimos para obtener conocimiento

Datos vs Información

- ¿Qué **información** obtenemos al observar esta imagen?



Datos vs Información

- **Datos:** representaciones de la información utilizando medios conocidos (símbolos numéricos, símbolos alfabéticos, algoritmos...)



Especie: humano

Género: mujer

Edad: entre 65 y 75 años

Raza: caucásica

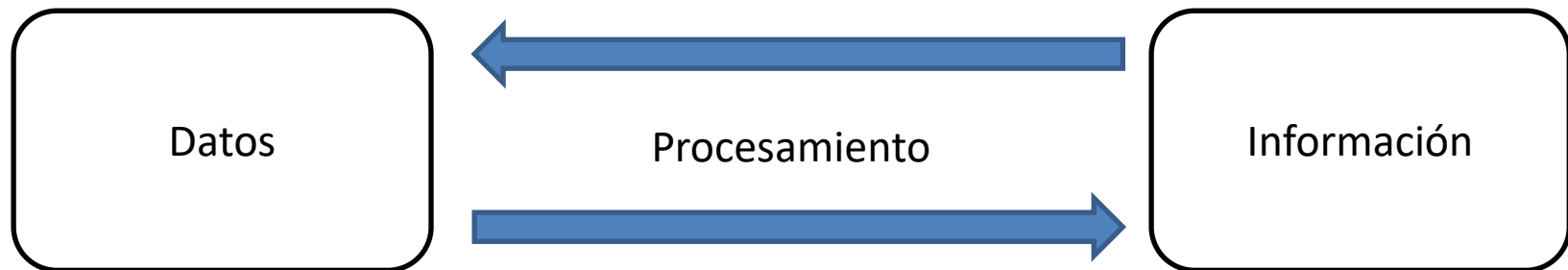
Color de ojos: marrón

Color de pelo: gris/blanco

Actitud: hostil

Datos vs Información

Codifico la información en forma de datos para poder almacenarla y compartirla con los demás



Interpreto los datos almacenados para obtener información y generar conocimiento

Datos vs Información

- En resumen...
 - La **información** es lo que llega hasta nosotros a través de nuestros sentidos
 - Los **datos** son la forma en que representamos y almacenamos dicha información

Concepto de Información

- Los **datos** están constituidos por los registros de los hechos, acontecimientos, transacciones, etc. Son la materia prima para obtener la información
 - Ejemplo: 7101200 es un dato
- La **información** implica que los datos estén procesados de tal manera que resulten útiles o significativos para el receptor de los mismos.
 - 7101200 puede significar que el día 7 del 10 a las 12:00 es la presentación del curso
- La información depende mucho del contexto.
 - 7101200 puede significar que la ciudad deportiva del Real Madrid costó 7.101.200 €

Concepto de Información

- La información debe de ser:
 - **relevante:** que interese para resolver el problema
 - **exacta:** estar libre de errores de cálculo o transmisión
 - **precisa:** nivel de precisión adecuado al problema
 - **oportuna:** no debe de tardar mucho tiempo entre que se emite y llega al usuario
 - **plena:** lo ideal es que sea completa
 - **significativa:** máximo contenido semántico posible
 - **coherente:** con el resto de información disponible
 - **segura:** protegida del deterioro o acceso no autorizado.
(*LOPD “Ley Orgánica de Protección de Datos”*)

Sistemas de información

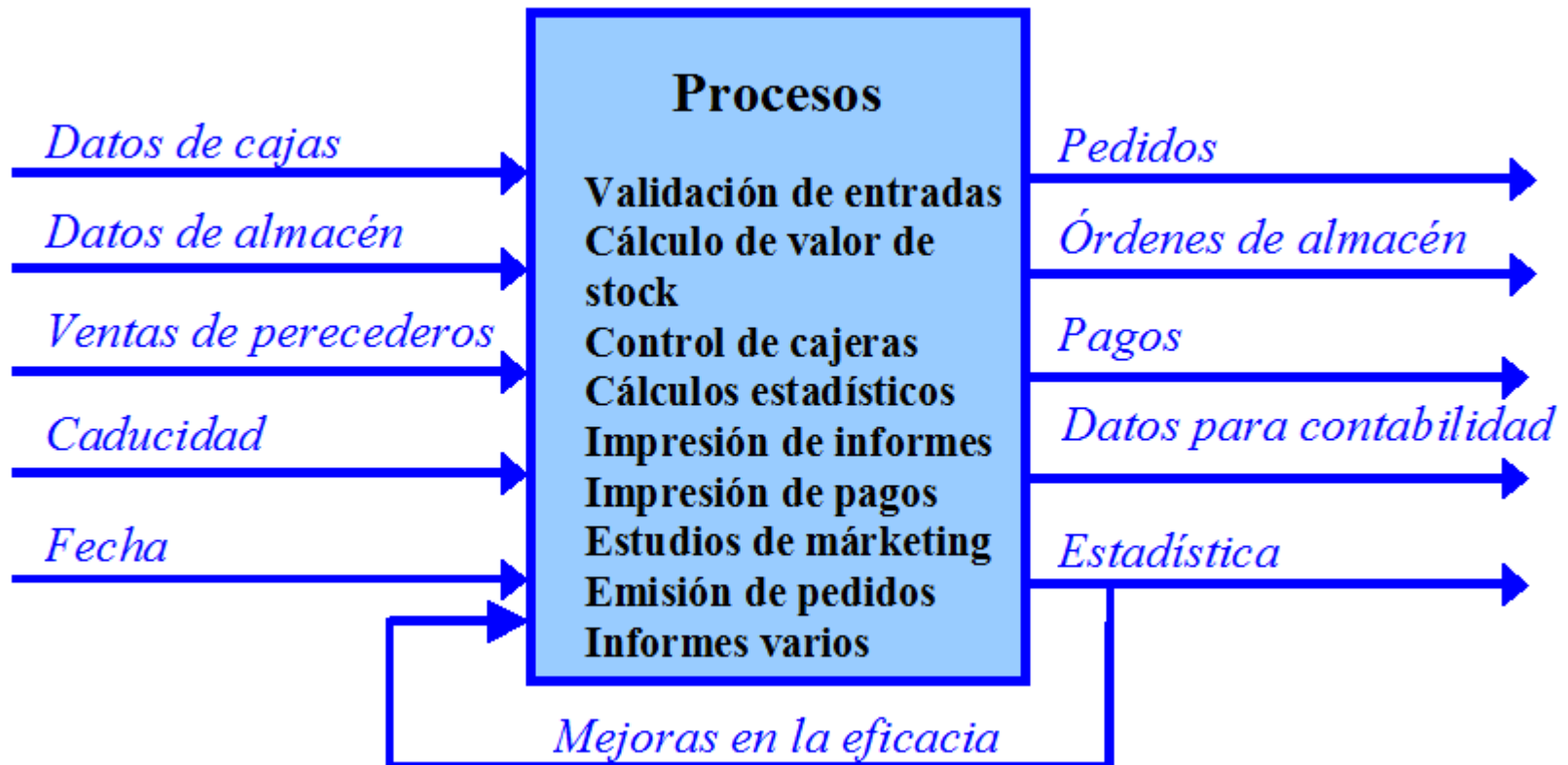
- Un **sistema** es un conjunto de cosas que ordenadamente relacionadas entre sí contribuyen a un determinado objetivo.” (*Real Academia Española*)
 - Ejemplos:
 - Sistema circulatorio del cuerpo humano
 - Sistema eléctrico en un hogar u oficina
 - Sistema solar

Sistemas de información

- ¿Qué es un sistema de información o SI?
 - Es una colección de personas, procedimientos y otros elementos relacionados entre sí diseñados, contruidos y mantenidos para **recoger, registrar, procesar, almacenar, recuperar y visualizar** información
 - Es decir, satisface las necesidades de información que pueda tener una empresa u organización
- En una empresa es un componente fundamental ya que permite gestionarla correctamente

Sistemas de información

- Ejemplo: SI de un hipermercado



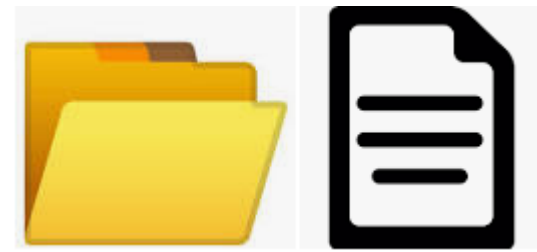
Sistemas de información

- El concepto de SI es anterior a los ordenadores
 - Antiguamente se utilizaban archivadores en papel
 - Se fueron incorporando herramientas como máquinas de escribir o calculadoras
- La aplicación del ordenador a los SI produce los **Sistemas de Información Automatizados (SIA)**.
- **SIA** : aquellos que aplican **Tecnologías de la Información**.



Ficheros

- El primer paso para automatizar los SI es almacenar la información en sistemas de **ficheros o archivos**.
- Observa que los nombres fichero, archivo o carpeta (y sus iconos) remiten a elementos físicos utilizados tradicionalmente para almacenar la información.

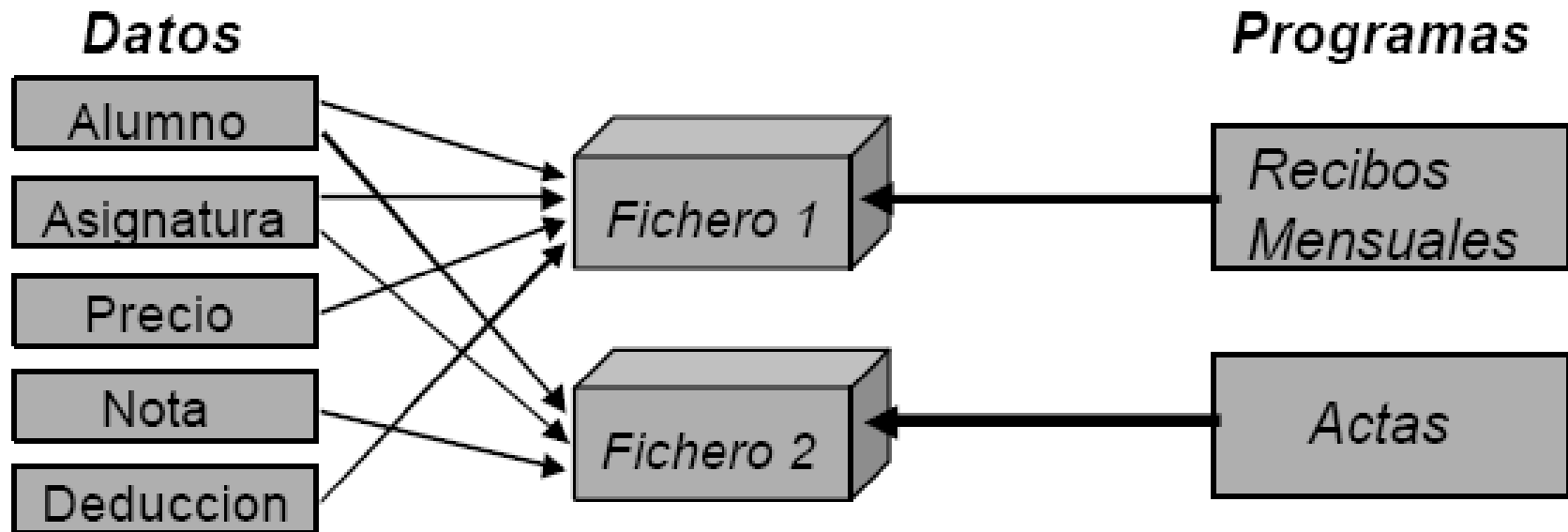


Sistemas de ficheros

- Un fichero es una estructura creada por el sistema operativo para poder almacenar datos.
- Un archivo posee un nombre y una extensión ([perro.jpg](#), [carta.doc](#), [tema.mp3](#)...) y contiene una secuencia de 0 y 1.
- La extensión permite al sistema operativo interpretar los datos almacenados correctamente
 - Recuerda: los datos necesitan interpretación

Sistemas de ficheros

- Los programas accedían a los ficheros para obtener los datos y procesar la información



Inconvenientes de los archivos

- Redundancia de los datos
 - Los datos pueden estar duplicados en varios archivos, por ejemplo el teléfono de un alumno
 - Aumenta los costes de almacenamiento y acceso
- Inconsistencia de los datos
 - Es consecuencia de lo anterior. Imagina que el teléfono se modifica en un archivo, pero queda sin modificar en otros

Inconvenientes de los archivos

- Dependencia física-lógica (dependencia del programa respecto a los datos)
 - El programa es el que decide e interpreta la estructura de los datos
 - Si se desea modificar esa estructura, es necesario revisar, modificar y probar todos los programas que manipulen ese archivo

Inconvenientes de los archivos

- Dificultad de gestión del acceso a los datos
 - Como hemos dicho, las consultas están codificadas en el código del programa
 - Si surge la necesidad de una nueva consulta (por ejemplo, recibos de ventas ordenados por fecha) es necesario reprogramar el código
- Problemas de seguridad
 - Es difícil implantar restricciones de seguridad: muchas aplicaciones distintas acceden a los datos

Inconvenientes de los archivos

- Separación y aislamiento de los datos
 - Los datos están repartidos en distintos archivos, que a su vez pueden seguir distintos formatos
 - Es necesario realizar operaciones de sincronización para que todos los datos coincidan
- Dificultad para el acceso concurrente
 - Es muy complicado gestionar los accesos concurrentes
 - Suele derivar en inconsistencias

Inconvenientes de los archivos

- Problemas de integridad de datos
 - Es muy difícil establecer restricciones de integridad, ya que exigen muchas líneas de código en los programas
 - Además, muchos programas acceden a un solo fichero
 - Ejemplo de restricción: “no se puede insertar una nota de alumno en una asignatura si la asignatura aún no se ha creado”

La solución...

- Con todos estos problemas, no es extraño que se dedicaran años a investigar el desarrollo de **Sistemas Gestores de Bases de Datos o SGBD**
- Se empezaron a desarrollar en los años 60
- La idea es lograr una colección de datos accesible para todas las aplicaciones
- ¿Qué características debería tener un SGBD?

Objetivos de un SGBD

- **Abstracción de la información**
 - Una abstracción consiste en aislar un elemento del contexto o del resto de elementos que lo acompañan
 - Ejemplos de abstracción: conducir un coche, utilizar una lavadora, ejecutar un programa en el ordenador
 - Se proporciona al usuario una visión abstracta de los datos
 - El usuario no necesita conocer en detalle cómo y dónde se están almacenando los datos, ni qué operaciones de mantenimiento se realizan sobre ellos

Objetivos de un SGBD

- **Redundancia mínima**
 - Redundancia: aparición de un mismo dato más de una vez en la misma colección de datos
 - Observa que decimos “mínima” y no “nula”, ya que en determinados casos se admite cierta redundancia para acelerar las búsquedas

Objetivos de un SGBD

- **Consistencia**

- Consiste en que si hay varias apariciones de un dato en la colección, todas contengan el mismo valor
- Para conseguirla, cuando se modifica un dato hay que propagar esa modificación a todos los demás lugares de la colección en que aparezca el mismo dato

Objetivos de un SGBD

- **Seguridad**

- Protección de los datos frente al acceso accidental o intencionado por parte de individuos no autorizados
- El usuario podrá determinar quién puede acceder a la información y qué puede hacer con ella
- Para ello se suelen utilizar usuarios y contraseñas

Objetivos de un SGBD

- **Integridad**

- La base de datos debe conservar la corrección y validez de los datos que contiene
- ¿Por qué se podría perder la integridad?
 - Fallos del hardware que da soporte a los datos
 - Actualizaciones de datos interrumpidas
 - Inserción de datos no válidas (por ejemplo la edad -23)

Objetivos de un SGBD

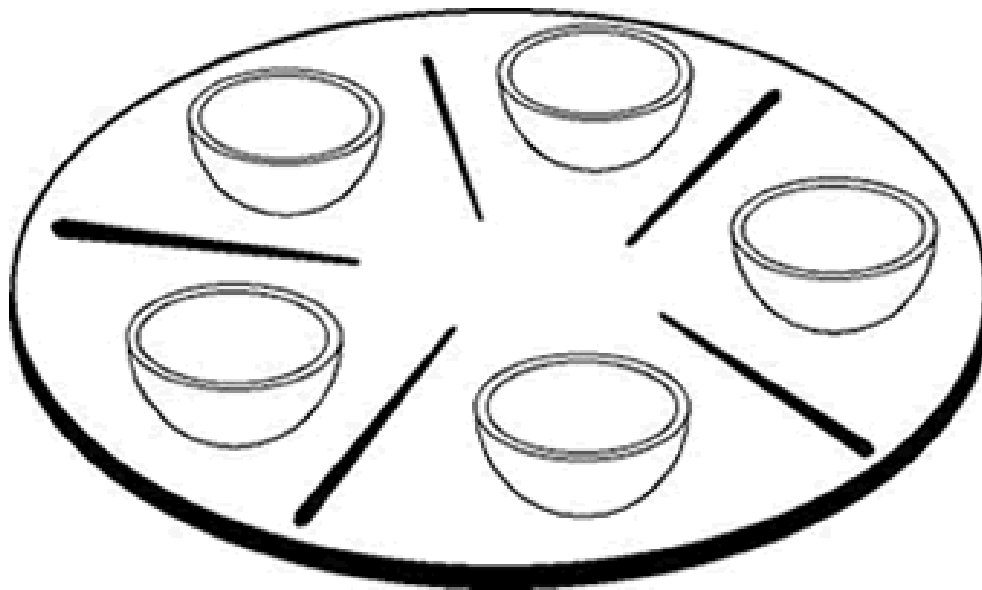
- Copias de **respaldo** (backup) y **recuperación**
 - Debe proporcionar mecanismos para:
 - Realizar copias de seguridad regularmente (backup)
 - Restaurar los datos originales a partir de una copia de seguridad

Objetivos de un SGBD

- Control de la **concurrency**
 - La concurrency es el acceso simultáneo
 - Es habitual que la BD esté en un entorno multiproceso y multiusuario
 - Esto puede generar problemas de consistencia
 - Solución del candado:
 - Si un proceso va a modificar un dato, debe esperar a tener un “candado”
 - Si no puede obtenerlo porque otro lo tiene, pasa a una cola de espera

Objetivos de un SGBD

- Control de la **concurrency**
 - El problema de la cena de los filósofos ilustra lo peligroso que es un problema de interbloqueo



Objetivos de un SGBD

- **Tiempo de respuesta**

- Los usuarios no deben tener que esperar mucho para obtener el resultado de sus consultas
- Es especialmente importante en accesos críticos como reserva de plazas en un tren o una operación bancaria

Objetivos de un SGBD

- Recopilando:
 1. Abstracción de la información
 2. Redundancia mínima
 3. Consistencia
 4. Integridad
 5. Seguridad
 6. Gestión de copias de seguridad
 7. Control de la concurrencia
 8. Tiempo de respuesta

Definiciones

- Una **base de datos** es una colección estructurada de datos relacionados entre sí, con redundancia mínima, y cuya finalidad es servir a una o más aplicaciones
- Un **sistema gestor de base de datos (SGBD)** está formado por
 - Un conjunto de datos relacionados entre sí
 - Un conjunto de programas que facilita el acceso a la base de datos
- El SGBD sirve por tanto de interfaz entre los datos y las aplicaciones

Componentes de un SGBD

- Los componentes principales de un SGBD son:
 - El gestor de bases de datos
 - El administrador
 - El diccionario de datos
 - Los lenguajes



El gestor de bases de datos

- Es un conjunto de programas transparentes al usuario
- Se encargan de:
 - Garantizar las características que se estudiaron antes (seguridad, integridad, consistencia, acceso concurrente, etc)
 - Interaccionar con el usuario y con el sistema operativo
 - Permitir que usuarios y programas manipulen los datos
 - Gestionar usuarios con distintos permisos de acceso
 - Cada vez ofrecen más prestaciones

El diccionario de datos (DD)

- Es una base de datos donde se almacena la descripción sobre la base de datos que se quiere manejar
- Hablamos de **metadatos**: datos relativos a los datos
 - Ejemplo: ¿podemos hablar sobre una conversación?
- Es una guía que describe detalladamente la base de datos y los objetos que la forman

El diccionario de datos (DD)

- ¿Qué podría contener?
 - La descripción de los datos contenidos
 - Las restricciones sobre los datos
 - Los valores por defecto de algunos datos
 - Los permisos que tiene sobre los datos cada usuario
 - La estructura física de los archivos que forman la base de datos

El administrador

- El SGBD describe distintas categorías de usuarios, pero siempre hay un usuario administrador (DBA, *DataBase Administrator*)
- Es el encargado de controlar y manipular la BD. Puede ser una persona o un equipo.
- Algunas de sus funciones:
 - Crear la base de datos, especificando el tipo de los datos, la estructura, etc.
 - Definir la estructura física, es decir, cómo y dónde se va a almacenar la información
 - Especificar reglas de integridad de los datos
 - Conceder autorizaciones de acceso a otros usuarios
 - Modificar cualquier aspecto de la BD si es necesario

Los lenguajes

- Todos los SGBD ofrecen distintos lenguajes para permitir realizar todas sus funciones
- Se trata de lenguajes con gramática sencilla, y fáciles de entender por usuarios no expertos
- Cada tipo de SGBD ofrece unos lenguajes distintos
- Existen tres tipos de lenguajes, adaptados a la función que se va a realizar

Los lenguajes

- **DDL** (Lenguaje de Definición de Datos)
 - Definición de los datos que se van a almacenar (qué tipo de datos, qué se almacena para cada uno, cómo se organizan...)

CREATE TABLE Alumnos

(
NumeroMatricula INTEGER UNIQUE,
Nombre VARCHAR(15) NOT NULL UNIQUE,
FechaNacimiento DATE,
);

Los lenguajes

- **DML** (Lenguaje de Manipulación de Datos)
 - Realizar consultas, insertar, borrar o modificar datos

```
SELECT * FROM empleados WHERE edad>18  
AND edad<45
```

```
SELECT AVG (edad) FROM empleados GROUP BY  
departamento
```

```
INSERT INTO animales (nombre, npatas, familia)  
VALUES ('caballo', 4, 'mamífero');
```

Los lenguajes

- **DCL** (Lenguaje de Control de Datos)
 - Gestionar privilegios, permisos, etc

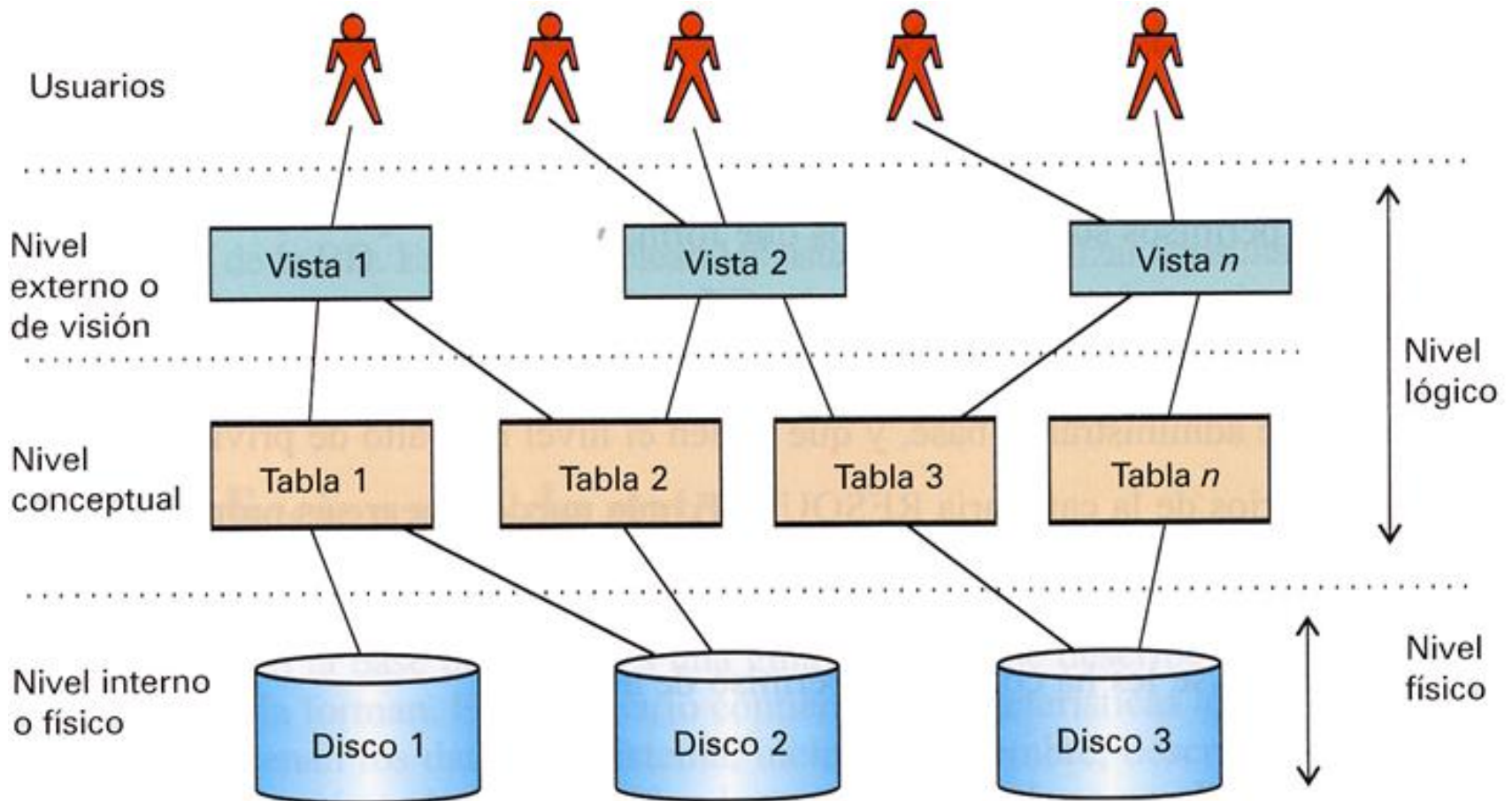
GRANT SELECT ON zoologico TO julia

REVOKE INSERT ON zoologico.animales FROM
juan

Arquitectura de un SGBD

- La más estandarizada es la propuesta en 1975 por el comité ANSI
 - ANSI : American National Standard Institute
- El propósito es separar muy bien los programas de aplicación de la base de datos física
- El esquema de una BD se define en tres niveles de abstracción distintos

Arquitectura ANSI



Arquitectura ANSI

- Nivel interno o físico
 - El más cercano al almacenamiento físico. Cómo y dónde se almacenan los datos, y cómo hay que acceder a ellos
- Nivel conceptual
 - Describe la estructura de toda la BD de forma comprensible para el usuario, incluyendo las relaciones y restricciones entre datos
- Nivel externo o de vistas
 - El más cercano a los usuarios. En él se describe sólo lo que puede ver o hacer un usuario concreto

ESQUEMA EXTERNO: Visión parcial de las tablas de la base de datos según el usuario.

Vista para el programa de listado de notas de alumnos con los siguientes datos: Curso, Nombre, Nombre de asignatura y Nota.

Curso	Nombre	Nombre Asignatura	Nota
1	Ana	Programación en lenguajes estructurados	6
1	Ana	Sistemas informáticos multiusuario y en red	8
2	Rosa	Desa. de aplic. en entornos de 4.ª Generación y H. Case	5
2	Juan	Desa. de aplic. en entornos de 4.ª Generación y H. Case	7
1	Alicia	Programación en lenguajes estructurados	5
1	Alicia	Sistemas informáticos multiusuario y en red	4

ESQUEMA CONCEPTUAL: Definición de todas las Tablas, Columnas y Restricciones.

Tabla ALUMNOS. Columnas: N.º Matricula, Nombre, Curso, Direccion, Población. Clave: N.º de matricula.

N_Matric	Nombre	Curso	Direc	Poblac
11111	Ana	1	C/Pilón 10	Oropesa
11110	Rosa	2	C/Las Viñas 26	Lagartera
11122	Juan	2	C/ Amapolas 24, 3F	Berrocalejo
23445	Alicia	1	C/Lamina 34	Caleruela

Tabla ASIGNATURAS. Columnas: Codigo, Nombre de asignatura. Clave: Codigo.

Tabla NOTAS. Columnas: N.º Matricula, Codigo asignatura, Nota.

Codigo	Nombre Asignatura
1	Desa. de aplic. en entornos de 4.ª Generación y H. Case
2	Programación en lenguajes estructurados
3	Sistemas informáticos multiusuario y en red

Matric	Codig	Nota
11111	2	6
11111	3	8
11110	1	5
11122	1	7
23445	2	5
23445	3	4

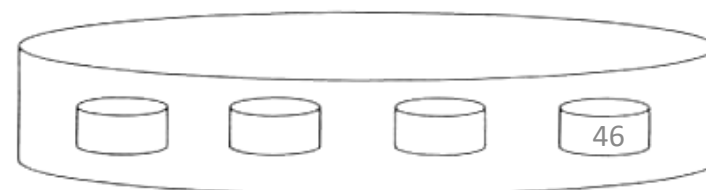
ESQUEMA INTERNO: Almacenamiento físico de los datos.

Archivo de Índices para ALUMNOS: Clave alumno, Dirección de la fila.

Archivo de Índices para ASIGNATURAS: Clave asignatura, Dirección de la fila.

Archivo de ALUMNOS: N.º Matrícula, Nombre, Curso, Dirección, Población.

Archivo de ASIGNATURAS: Código, Nombre de asignatura.

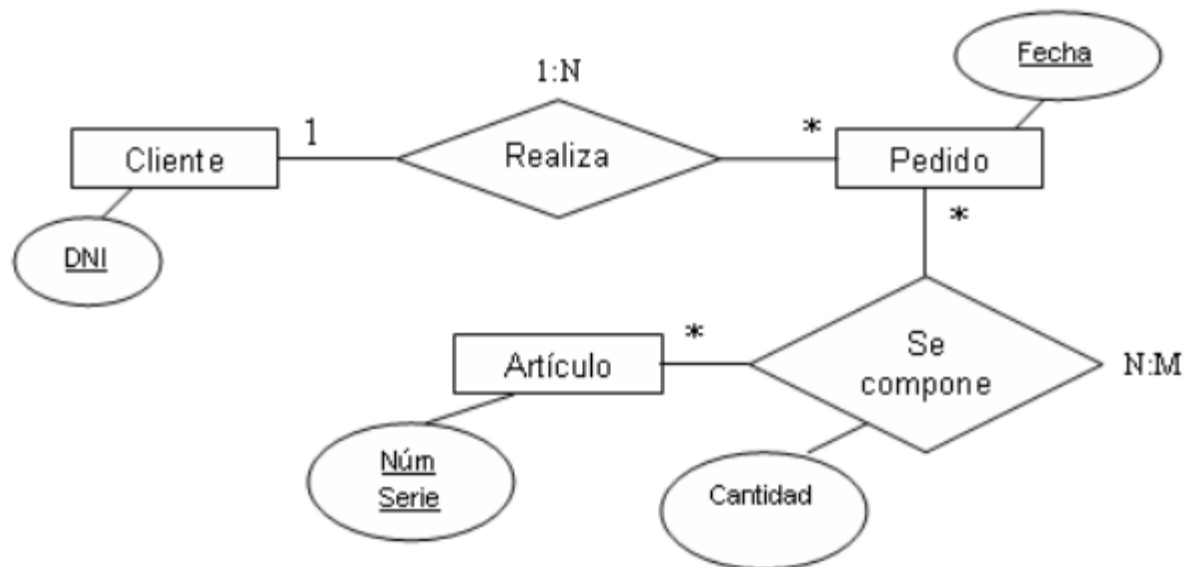


Modelo de datos

- Un **modelo de datos** es el conjunto de conceptos o herramientas conceptuales que sirven para describir la estructura de una BD: los datos, las relaciones y las restricciones que se deben cumplir sobre los datos
- Un problema de bases de datos se puede modelar de distintas formas según quién tenga que interpretar y manejar ese modelo
 - Por eso existen muchos modelos, por ejemplo:

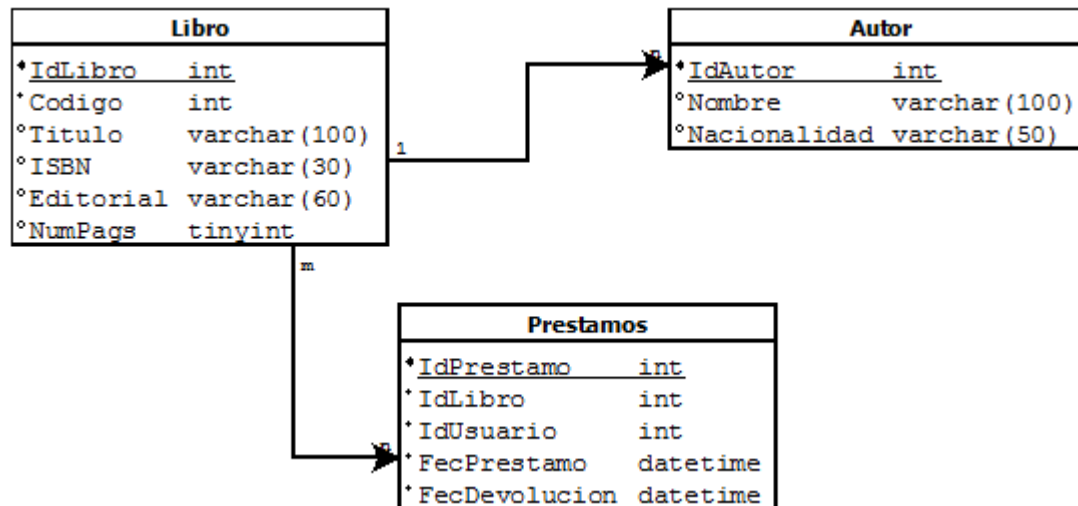
Modelo de datos

- Modelo Entidad-Relación
 - Es un modelo conceptual. Resulta intuitivo y es fácil para expresar el problema de una forma cercana al usuario



Modelo de datos

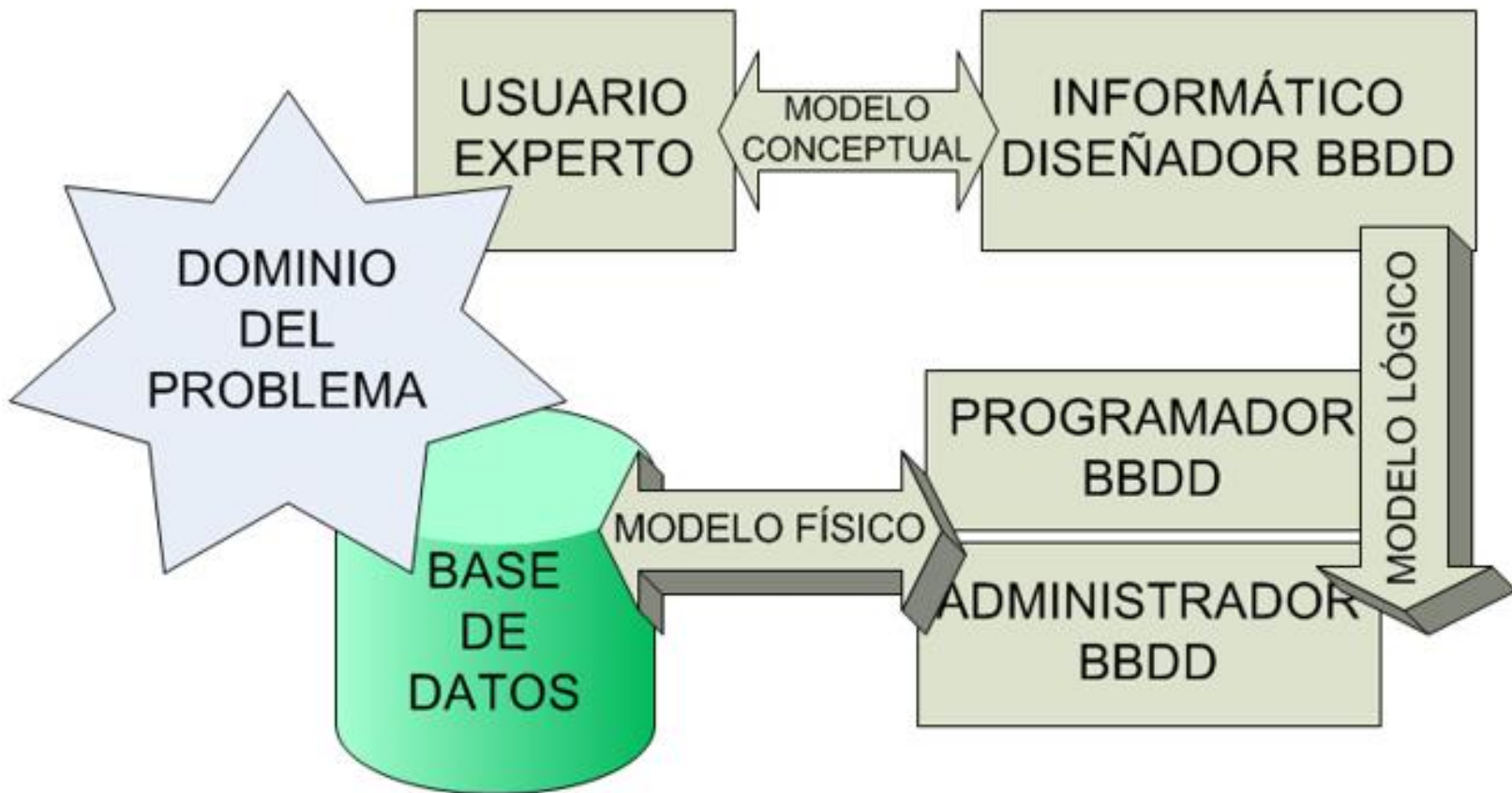
- Modelo relacional
 - Es un modelo lógico, más técnico que el anterior
 - Utiliza tablas bidimensionales para representar los datos



Modelo de datos

- Proceso habitual de modelado:
 - Se utiliza el **modelo conceptual Entidad-Relación** para expresar el problema. Habitualmente esto requiere conversaciones con el cliente
 - Utilizando unas reglas que estudiaremos, se transforma ese modelo en un **modelo lógico relacional**, más cercano al diseño físico de la BD.
 - Por último se implementa el **modelo físico** en un SGBD concreto, usando para ello el lenguaje de definición de datos DDL.

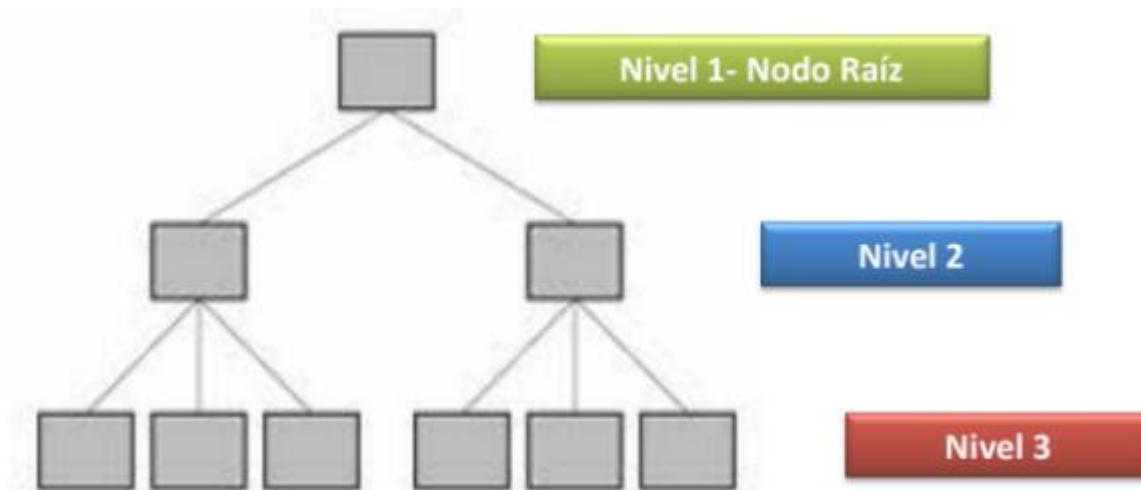
Modelo de datos



(imagen: Editorial Garceta)

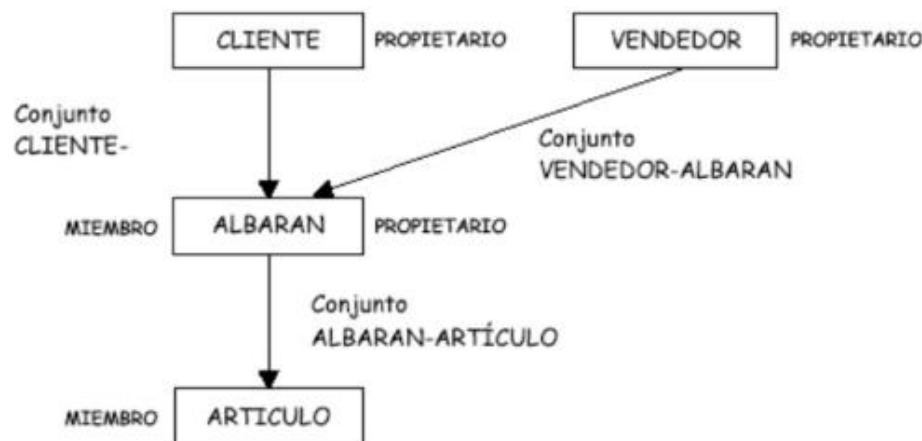
Otros modelos de datos

- Modelo jerárquico
 - Usa estructura de árbol para la representación lógica de los datos
 - Se usó en la década de 1970 para ficheros



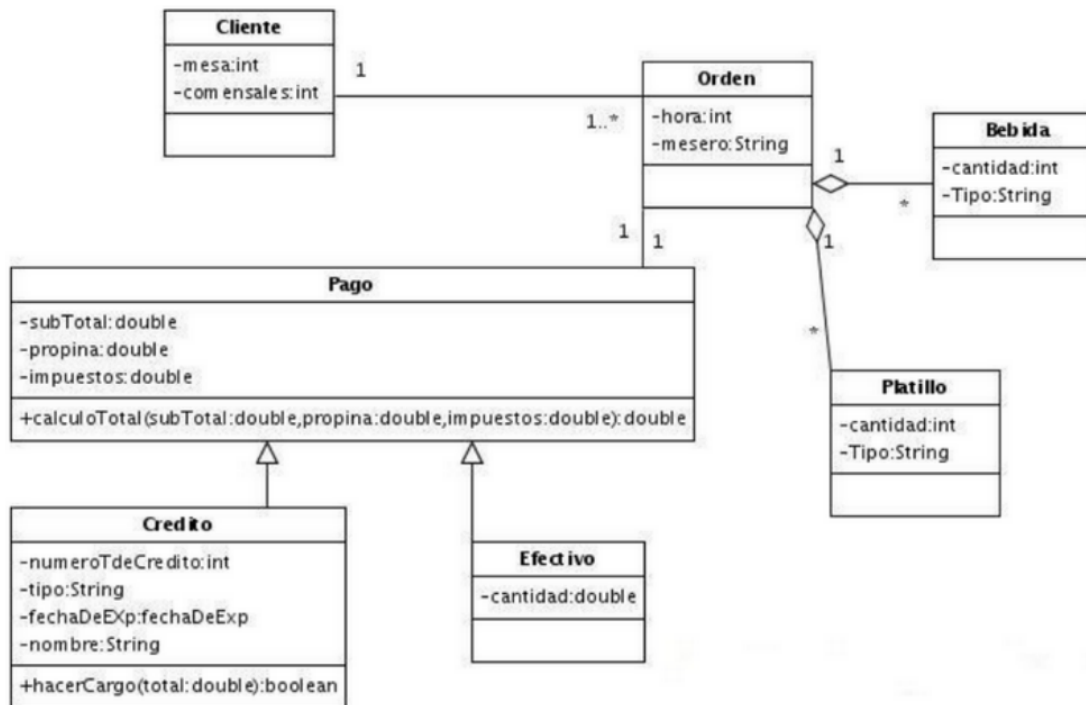
Otros modelos de datos

- Modelo en red
 - Es una generalización de las bases de datos jerárquicas
 - Se basa en la estructura de un grafo: nodos y líneas que los unen (como la red de Metro)



Otros modelos de datos

- Modelo orientado a objetos
 - Se basa en la programación orientada a objetos
 - Los datos son objetos y tienen propiedades y métodos



Aplicaciones de las bases de datos

- Banca: clientes, cuentas, préstamos, transacciones, compras con tarjeta de crédito...
- Líneas aéreas: reservas, información de horarios...
- Centros educativos: información de estudiantes, matrículas, asignaturas, cursos...
- Telecomunicaciones: llamadas realizadas, facturas, saldo de tarjetas de prepago...
- Ventas: clientes, productos, compras, pedidos...
- Recursos humanos: empleados, salarios, impuestos sobre sueldos, generación de nóminas...

El futuro

- El contenido fundamental de este módulo ha estado vigente durante 60 años (los modelos relacional y entidad relación, el lenguaje SQL...). Si algo funciona, ¿para qué modificarlo?
- En los últimos años han surgido bases de datos de tamaño muy grande (big data), y los usuarios participan generando ingentes cantidades de contenido (redes sociales)
- Se están desarrollando SGBD llamados NoSQL (not only SQL) que flexibilizan el almacenamiento en tablas y buscan estructuras y lenguajes más ligeros y flexibles para lograr consultas más rápidas, sacrificando en ocasiones algunas características (como la consistencia) en favor de la rapidez

Ejemplos de SGBD

- Se pueden clasificar según diferentes criterios, por ejemplo:
 - Si están pensados para hogar o pequeñas empresas (Access), medianas empresas (MySQL, MariaDB) o grandes (Oracle)
 - Si siguen el modelo relacional o son NoSQL (mongoDB, cassandra)

