

# UD 1. INTRODUCCIÓN A LAS BASES DE DATOS

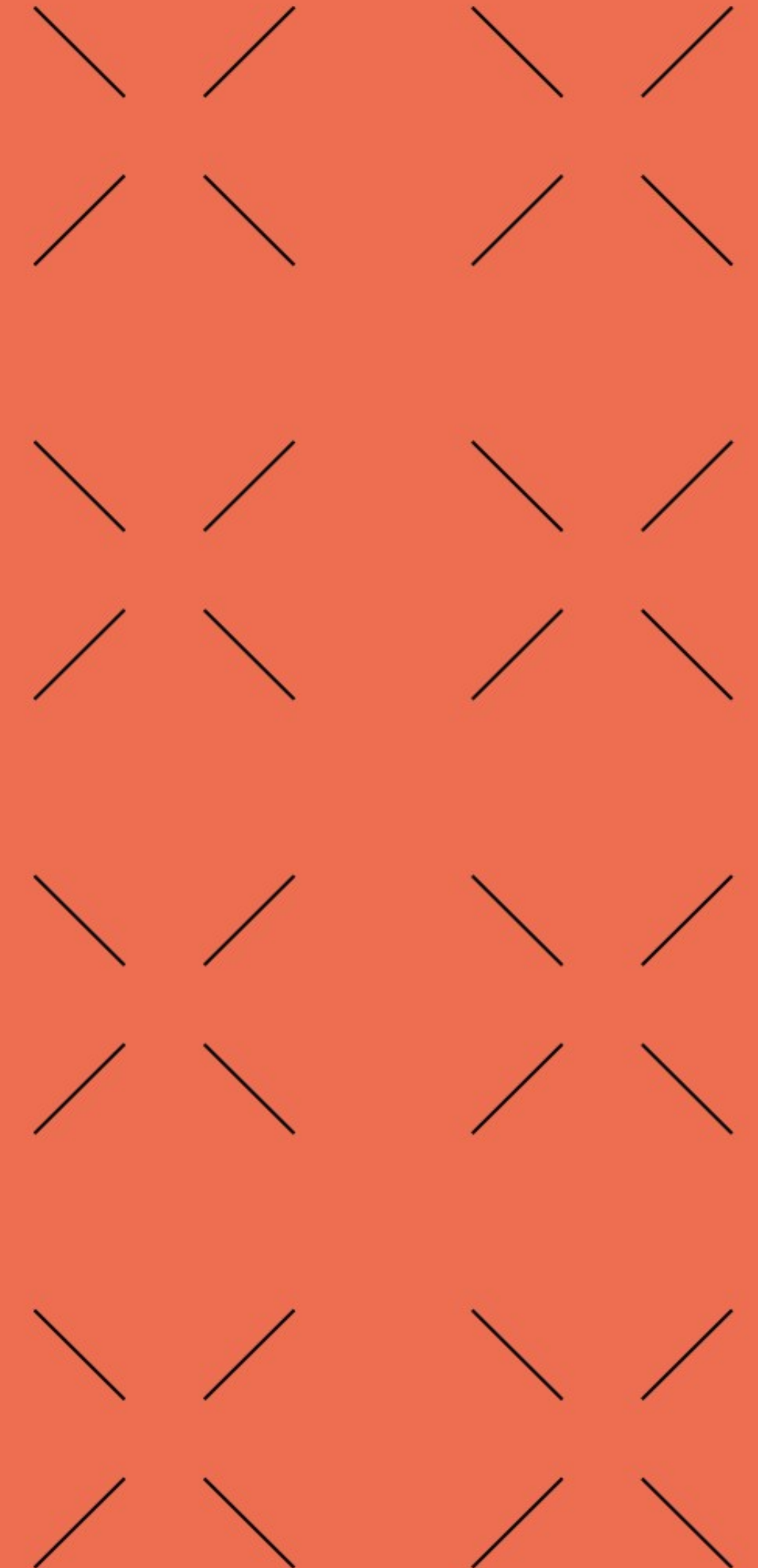
**Bases de Datos (DAW/DAM)**

**CFGS Desarrollo de Aplicaciones Web (DAW)**

**CFGS Desarrollo de Aplicaciones Multiplataforma (DAM)**

**Abelardo Martínez y Pau Miñana**

**Curso 2023-2024**



# Créditos



- Apuntes realizados por Abelardo Martínez y Pau Miñana.
- Basados y modificados de Sergio Badal ([www.sergiobadal.com](http://www.sergiobadal.com)) y Raquel Torres.
- Las imágenes e iconos empleados están protegidos por la licencia [LGPL](#) y se han obtenido de:
  - [https://commons.wikimedia.org/wiki/Crystal\\_Clear](https://commons.wikimedia.org/wiki/Crystal_Clear)
  - <https://www.openclipart.org>

# Contenidos

- 1.SISTEMAS DE ALMACENAMIENTO DE LA INFORMACIÓN
- 2.FICHEROS
- 3.BASES DE DATOS
- 4.SISTEMAS GESTORES DE BBDD
- 5.ARQUITECTURA DE NIVELES
- 6.ACTIVIDADES PROPUESTAS
- 7.BIBLIOGRAFÍA



# **1. SISTEMAS DE ALMACENAMIENTO DE LA INFORMACIÓN**

# Conceptos relevantes

## INFORMÁTICA

Informática (**información + automática**)

Ciencia que tiene como objetivo automatizar el trabajo con información.

- **DATOS**  $\Rightarrow$  hechos, eventos, transacciones, etc., que han sido registrados.

- **INFORMACIÓN**  $\Rightarrow$  datos que han sido procesados, clasificados y comunicados de tal manera que pueden ser entendidos e interpretados por el receptor.

## DATO

Cualquier elemento informativo que tenga relevancia para el sistema.

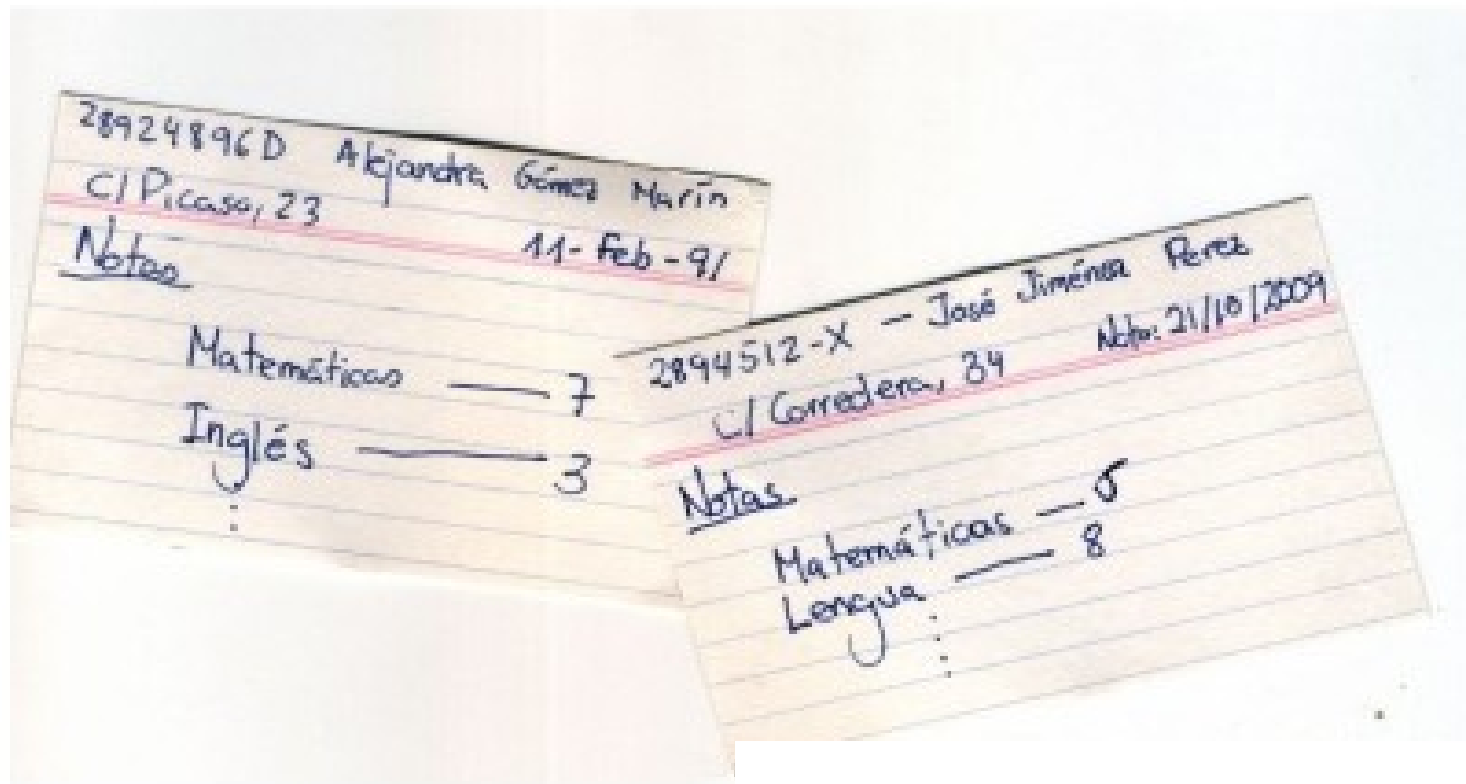


# Evolución del almacenamiento de información

Para poder almacenar datos hemos creado herramientas que han ido evolucionando:

**Papel → Cajones → Carpetas → Archivadores → 0/1 → Ficheros → Hojas de cálculo → Bases de datos**

En los años 60, usando papel y "boli"



En los años 70-80 se utilizaban ficheros de texto donde se guardaba la información.



alumnos.txt

DNI	NOMBRE	DIRECCIÓN	FECHA	NTD
2894512X	José Jiménez Pérez	C/ Corredera,34	21-10-90	
28924896D	Alejandra Gómez Marín	C/ Picasso, 23	11-02-91	
...				

asignaturas.txt

DNI	NOMBRE	ASIGNATURA	NOTA
2894512X	José Jiménez Pérez	Matemáticas	5
2894512X	José Jiménez Pérez	Lengua	8
.....			
28924896D	Alejandra Gómez Marín	Matemáticas	7
28924896D	Alejandra Gómez Marín	Inglés	3

A partir de los años 90 y posteriores, empiezan a utilizarse los Sistemas Gestores de Bases de Datos (SGBD) donde se comienza a almacenar la información en forma de tablas.

Alumnos (DNI, Nombre, Dirección, Fecha nacimiento)				
2894512X	José Jiménez Pérez	C/ Corredera,34	21-10-90	
28924896D	Alejandra Gómez Marín	C/ Picasso, 23	11-02-91	
...				
Asignaturas (Código, Nombre)				
001	Matemáticas			
002	Lengua			
003	Inglés			
Notas(DNI, Código_asignatura, nota)				
2894512X	001	5		
2894512X	002	8		
28924896D	001	7		
28924896D	003	3		



## 2. FICHEROS



# Fichero / Archivo

Un **fichero** (también denominado **archivo**) es un conjunto ordenado de datos que tienen entre sí una relación lógica y están almacenados en un soporte de información adecuado para la comunicación con el ordenador.

De este modo, un fichero es una abstracción del sistema operativo (SO) para el almacenamiento genérico de datos. La parte del SO encargada del manejo de ficheros se denomina “sistema de ficheros”.

Por ejemplo, los ficheros se pueden almacenar en un disco duro interno/externo, una memoria USB, un servidor en la nube, etc.





# Tipos de ficheros

**Ficheros de texto.** Están diseñados para ser leídos por seres humanos y pueden ser leídos/escritos por un editor/procesador de texto. Los ficheros de texto suelen llamarse también ficheros planos o ficheros [ASCII](#).

- **Ventajas:** usualmente se representan igual en todos los ordenadores, por lo que pueden moverse/copiarse/exportarse de un ordenador a otro sin ningún problema.
- **Inconvenientes:** no son tan eficientes de procesar como los de tipo binario.



**Ficheros binarios.** Están diseñados para ser leídos por programas específicos y están formados por una secuencia de dígitos binarios.

- **Ventajas:** son más eficientes de procesar que los ficheros de texto.
- **Inconvenientes:** están diseñados para leerse en el mismo tipo de ordenadores y con el mismo lenguaje/programa del ordenador que los ha creado. No se puede ver el contenido directamente.



Una **base de datos**, como veremos ahora, **se almacena (generalmente) en forma de ficheros binarios**, dado que la información que hay almacenada en ellos debe tener una estructura lógica y organizada siguiendo un estándar.

### **3. BASES DE DATOS**

# Concepto de base de datos



Los datos que se utilizan de manera informatizada se almacenan, habitualmente, en bases de datos (BBDD). Una base de datos es un **conjunto de datos que están organizados para un uso determinado**.

**Ejemplos:** base de datos de alumnos, clientes, proveedores, pacientes, vehículos, películas, etc.

Debemos distinguir 3 ámbitos diferenciados y al mismo tiempo fuertemente relacionados:

- **Mundo real.** Está constituido por los objetos (materiales o no) de la realidad que nos interesan y con los cuales tendremos que trabajar.
- **Mundo conceptual.** Es el conjunto de conocimientos o informaciones obtenidos gracias a la observación de la parte del mundo real que nos interesa.
- **Mundo de las representaciones.** Está formado por las representaciones informáticas, o datos, del mundo conceptual, necesarias para poder trabajar.

# Elementos lógicos

Una base de datos es un conjunto estructurado de datos que representa entidades, con sus propiedades o atributos y las relaciones que existen entre esas entidades. Una base de datos constará de los siguientes elementos LÓGICOS:

Elementos	Descripción
Entidades	<p>Son los objetos del mundo real que conceptualizamos. Son identificables, es decir, distinguibles los unos de los otros. Y nos interesan algunas (como mínimo una) de sus propiedades.</p> <p>Ejemplo: BD de un hospital. Posibles entidades podrían ser: paciente, doctor, analítica, habitación, etc.</p>
Atributos	<p>Son las propiedades de las entidades que nos interesan. Cada atributo tiene un tipo de dato: texto, numérico, fecha, hora, booleano (verdadero/falso), etc.</p> <p>Ejemplo (podrían ser atributos): nombre, edad, dirección, fecha nacimiento, teléfono, etc. Y sus tipos serían: texto, numérico, texto, fecha, numérico, etc.</p>
Valores	<p>Son los contenidos concretos de los atributos, las determinaciones concretas que logran.</p>

## Ejemplo completo:

- **Entidad.** Una película concreta es una ocurrencia de una entidad, porque es un objeto del mundo real, que hemos conceptualizado dentro de una entidad/categoría (la de las películas cinematográficas), y que al mismo tiempo es distinguible de otras ocurrencias de la misma entidad/categoría (es decir, otras películas).
- **Atributos.** De esta película nos interesarán algunos aspectos (atributos), como por ejemplo, el título, el director y el año de producción.
- **Valores.** Finalmente, estos atributos adoptarán unos valores concretos como por ejemplo, y respectivamente, Senderos de Gloria, Stanley Kubrick y 1957.



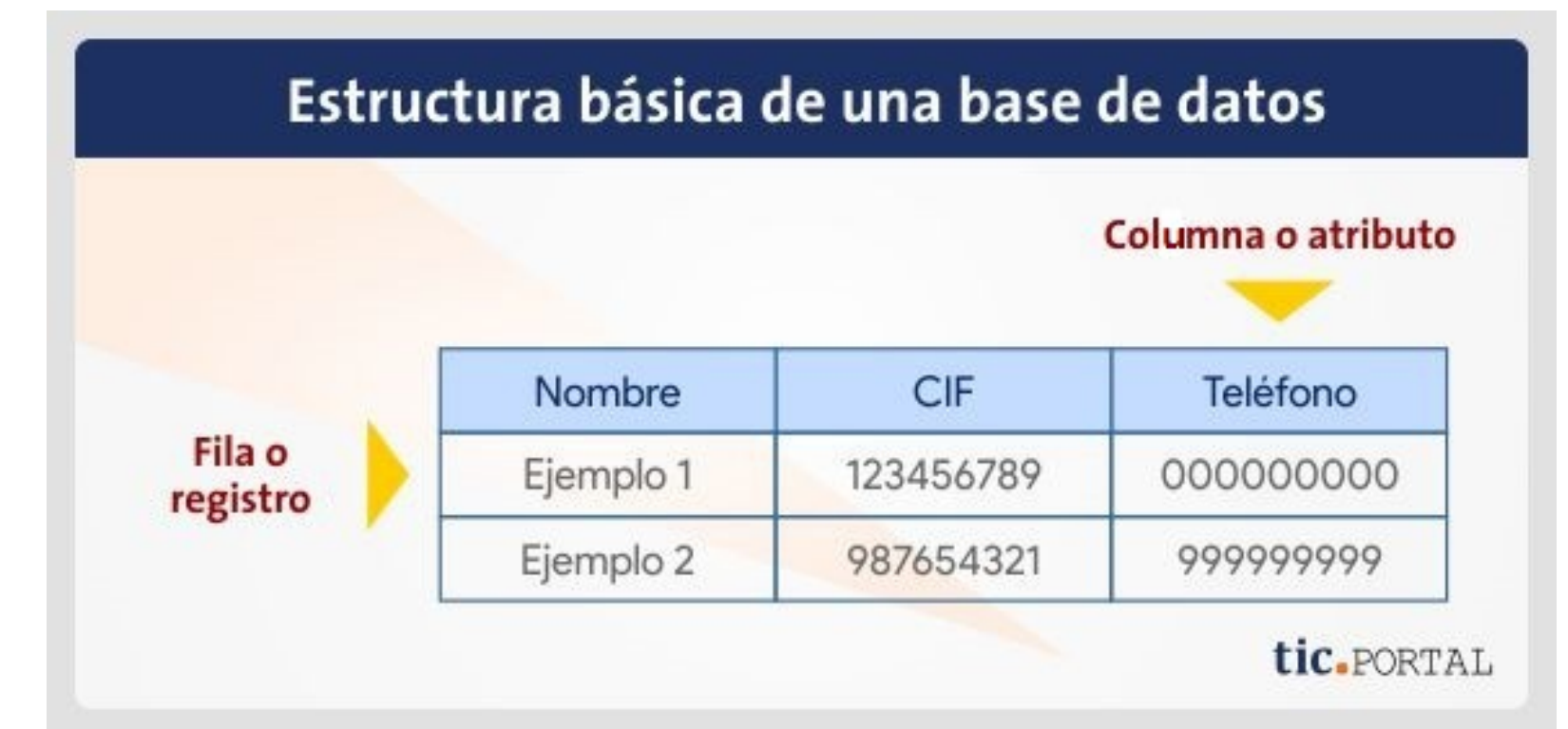
# Representación tabular. Tablas

La representación informática más frecuente en el ámbito de las BBDD es la **representación tabular** (o, lo que es lo mismo, en forma de tabla), la cual se implementa habitualmente en ficheros que se estructuran en registros y campos.

Cada una de estas tablas es una estructura que **se parece a las hojas de cálculo**, pues está dispuesta mediante filas y columnas.

Cada tabla representa una entidad genérica, y está estructurada en filas (agrupaciones horizontales de celdas) y columnas (agrupaciones verticales de celdas):

- Cada **fila** representa una entidad instancia.
- Cada **columna** representa un atributo.
- Cada **celda** (cada intersección de una fila y de una columna) almacena el valor que tenga el atributo de la entidad instancia de que se trate.



**Registro**

ID_CLIENTE	NOMBRE	DIRECCION	TELEFONO
4545455	DAVID RAMIREZ	CRA 78 89-78	3256987
7896321	AMANDA GARCIA	CRA 56 78-80	3264789
8254569	MARIA MENDOZA	CLL 50 45-02	3697891
32456123	PEDRO DIAZ	CRA 80 5-45	2581453
32796353	MARTHA GONZALEZ	CLL 80 35-56	3593027
72207033	ANDRES PEREZ	CRA 7 45-50	3565859
74258943	JAVIER HERNANDEZ	CLL 50 59-78	3456901



# Elementos físicos

La implementación informática de las representaciones tabulares se materializa mediante los llamados **ficheros de datos**. Los ficheros se tienen que almacenar en algún dispositivo de memoria externa del ordenador, típicamente un disco duro, para conservar los datos permanentemente.

Una base de datos constará de los siguientes elementos FÍSICOS:

Elementos	Descripción
Registros	<p>La implementación de cada entidad instancia se denomina registro, y equivale a una fila de la representación tabular, con su conjunto de atributos. En un registro se almacena FÍSICAMENTE la información de cada entidad.</p> <p>Ejemplo: 131789, Pepe, Ginés Martínez, Estudiante, BD, 7/08/1993</p>
Campos	<p>La implementación de cada atributo se denomina campo, y equivale a una columna de la representación tabular. En el campo se almacenan FÍSICAMENTE los atributos de cada registro.</p> <p>Ejemplo: Dirección</p>
Valores	<p>Cada intersección de un registro y de un campo almacena el valor que tenga el campo del registro de que se trate.</p> <p>Ejemplo: “c/ Blas de Lezo 8”</p>



# Tipos de bases de datos según su localización

## 1) Distribuidas

Es un conjunto de bases de datos relacionadas lógicamente entre sí que se encuentran localizadas en diferentes ubicaciones físicas. Son menos comunes y se usan, generalmente, para almacenar ingentes cantidades de datos.

**Ventajas:** minimizan la pérdida de datos, ya que la información está distribuida entre varios servidores.

**Desventajas:** son más difíciles de programar y mantener.

## 2) Centralizadas

Se dedica un único equipo para el almacenamiento y gestión de los datos. Son las más comunes y las que veremos en este módulo.

**Ventajas:** son más sencillas de implementar que las distribuidas y la infraestructura requerida es menos compleja.

**Desventajas:** si cae el nodo (servidor) central, no se puede acceder a los datos.

## **4. SISTEMAS GESTORES DE BBDD**

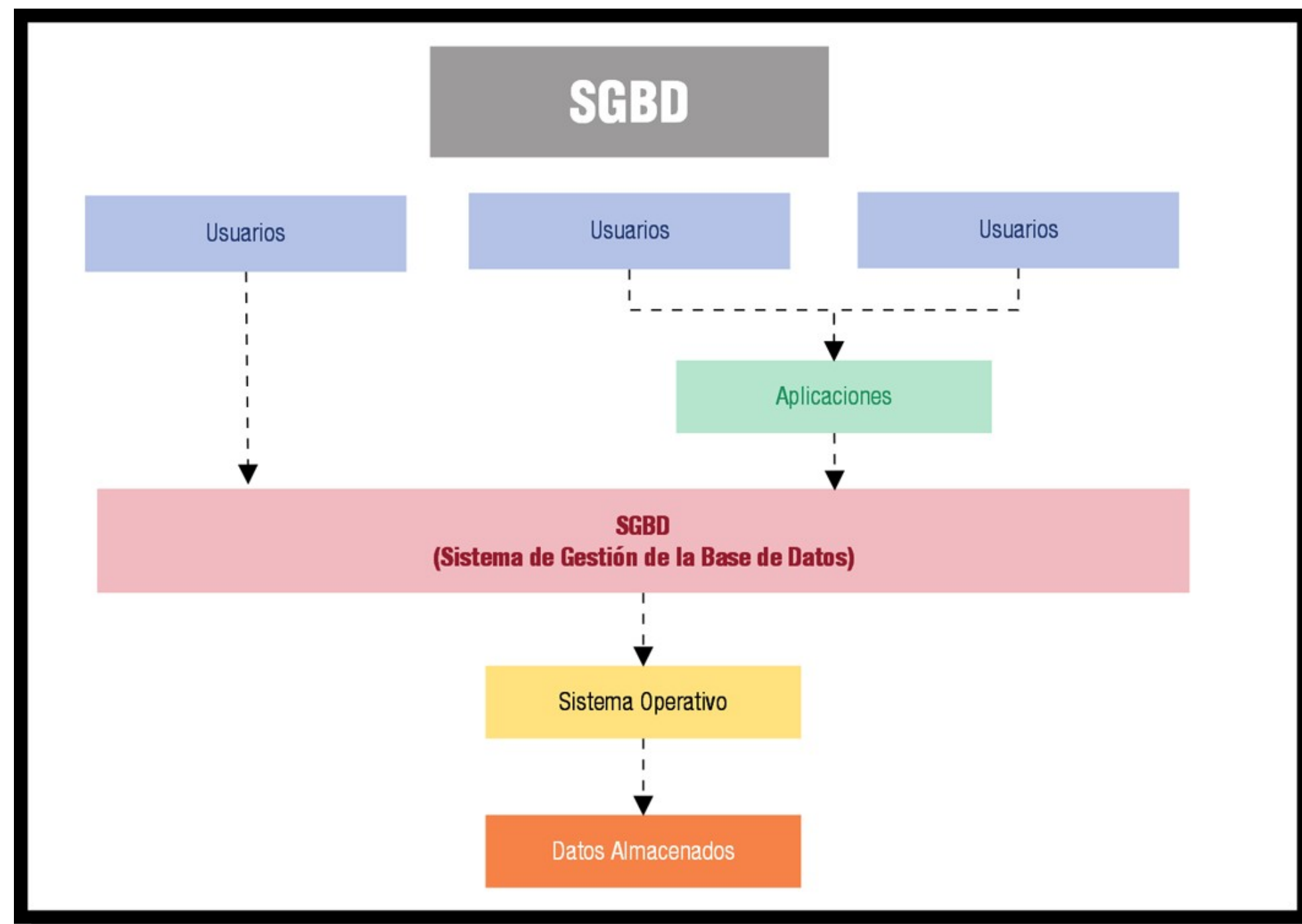
# SGBD. Definición

Un **Sistema de Gestión de Bases de Datos (SGBD)** o *DataBase Management System (DBMS)* es:

Una colección de **datos** relacionados entre sí estructurados y organizados

+

Un conjunto de **programas** que acceden y gestionan esos datos



Un SGBD actúa de intermediario entre nosotros y la base de datos, de modo que nos abstraer de la implementación física de ésta.

# Objetivos de los SGBD

Todos los SGBD del mercado quieren lograr una serie de objetivos y ofrecer una serie de funcionalidades, con más o menos acierto, que actualmente se consideran indispensables para el buen funcionamiento de cualquier sistema de información:

- Permitir consultas no predefinidas y complejas.
- Ofrecer flexibilidad e independencia de datos.
- Minimizar redundancia.
- Garantizar integridad de los datos y referencial.
- Permitir concurrencia de usuarios.
- Proporcionar seguridad de la información.



# Historia de los SGBD



Igual que en otros ámbitos de software (como por ejemplo, en el de los sistemas operativos), la evolución de los SGBD ha estado, a menudo, intrínsecamente ligada a la evolución del *hardware*.

Echa un vistazo a esta infografía de cómo han evolucionado las bases de datos desde las **jerárquicas** (prácticamente en desuso) a las novedosas **no relacionales** (MongoDB, Cassandra, etc.) pasando por las omnipresentes **relacionales** (Oracle, MySql, etc.).

<https://www.bbvaapimarket.com/es/mundo-api/infografia-tipos-de-bases-de-datos/>

# SGBD comerciales

En esta página podemos encontrar la lista de los SGBD más utilizados actualmente:

<https://db-engines.com/en/ranking>

Rank			DBMS	Database Model
Aug 2023	Jul 2023	Aug 2022		
1.	1.	1.	Oracle +	Relational, Multi-model i
2.	2.	2.	MySQL +	Relational, Multi-model i
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model i
4.	4.	4.	PostgreSQL +	Relational, Multi-model i
5.	5.	5.	MongoDB +	Document, Multi-model i
6.	6.	6.	Redis +	Key-value, Multi-model i
7.	↑ 8.	↑ 8.	Elasticsearch	Search engine, Multi-model i
8.	↓ 7.	↓ 7.	IBM Db2	Relational, Multi-model i
9.	9.	9.	Microsoft Access	Relational
10.	10.	10.	SQLite +	Relational

Principales SGBD de la actualidad







# Terminología



Es muy importante que tengas clara la diferencia entre base de datos y sistema gestor de bases de datos y sus siglas.

- **Una** base de datos: **BD** de películas
- **Varias** bases de datos: **BD** de películas y clínicas
- **Un** Sistema Gestor de Bases de Datos: **SGBD** MySQL

Aunque a menudo se confunden los términos, cuando hablamos de **BD/BBDD** nos referimos a los **DATOS** y cuando hablamos de **SGBD** nos referimos al **software y a los datos**.

**Sí. Las siglas del módulo son BD, aunque deberían ser BBDD.**



# Aplicaciones de gestión y productividad personal



- **No son SGBD.**
- Tienen diversas funcionalidades integradas: gestión personal, calendario, agenda, bloc de notas, etc.
- Permiten organizar la información de distintas formas.
- Diseñadas para gestionar pequeñas cantidades de datos.
- Por ejemplo, Notion, Zoho One, Wunderlist, Evernote, etc.



# SGBD SQLite



SQLite es un **motor de bases de datos muy ligero, de código abierto y escrito en C**, que fue creado por D. Richard Hipp. Es un software que permite a los usuarios interactuar con una base de datos relacional y donde podremos guardar todo tipo de información relacionada con un programa o una app.

## Características principales:

- **Reducido tamaño.** Una base de datos se almacena en un único archivo, un rasgo que lo distingue de otros motores de base de datos.
- Hace **uso de SQL**, un lenguaje específico para trabajar con bases de datos. Por ello, las consultas y órdenes se pueden enviar a este servidor directamente en este lenguaje.
- Además, sus bases de datos cuentan con la mayor parte del **estándar SQL-92**, por lo que funcionan sin problemas con otros programas que trabajen con instrucciones SQL.
- Sus versiones más recientes permiten trabajar sin problemas con **bases de datos de hasta 2 TB de tamaño**.

# SGBD SQLite. Ventajas e inconvenientes

## Ventajas:

- SQLite funciona como un servidor propio e independiente, evitando tener que realizar consultas externas en procesos separados. O lo que es lo mismo, se incluye la base de datos y el motor dentro del programa.
- Este hecho permite una gran accesibilidad: copiar una base de datos no es más complicado que copiar el archivo que almacena los datos, compartir una base de datos puede significar enviar un archivo adjunto de un correo electrónico.

## Inconvenientes:

- La portabilidad de SQLite hace que sea una mala elección cuando muchos usuarios diferentes están actualizando una tabla al mismo tiempo (para mantener la integridad de los datos, solo un usuario puede escribir en el archivo a la vez).
- En SQLite el acceso es directo y sin restricciones; es decir, no hay usuarios, por lo que requiere algo más de trabajo para garantizar la seguridad de los datos privados.
- No ofrece exactamente la misma funcionalidad que muchos otros sistemas de bases de datos.
- SQLite no valida los tipos de datos.

# ¿Cómo y por qué utilizar SQLite?



## SQLite se utiliza especialmente:

- Para pruebas, desarrollo y en cualquier otro escenario en el que tenga sentido que la base de datos esté en el mismo disco que el código de la aplicación.
- Por aquellas personas que quieren poner en práctica sus conocimientos de programación o probar soluciones antes de realizar una alta inversión.
- Para evitar todas las complejidades involucradas de un cliente/servidor.

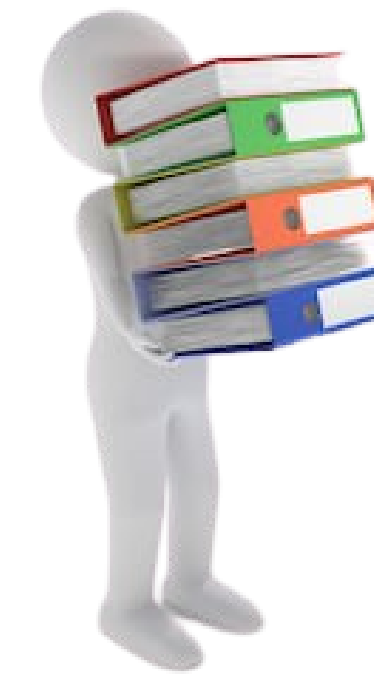
## Algunos de los usos más comunes son:

- Programas de escritorio y aplicaciones móviles. Además, es la opción ideal para los dispositivos integrados destinados al Internet de las Cosas (**IOT** por las siglas en inglés de *Internet Of Things*).
- Sitios web muy ligeros, como páginas estáticas.
- Sitios o sistemas que cuenten con un número relativamente bajo de usuarios (en torno a 100.000 accesos diarios).

## **5. ARQUITECTURA DE NIVELES**



# Volumen de datos



¿Cómo almacenarías un listado de los compañeros de este módulo?

- a) Bloc de notas
- b) Procesador de textos (LibreOffice Writer, MS Word, etc.)
- c) Hoja de cálculo (LibreOffice Calc, MS Excel, etc.)

¿Y un listado con todos los millones de usuarios de una tienda multinacional de venta *online*?

- a) Bloc de notas ¿**seguro**?
- b) Procesador de textos (LibreOffice Writer, MS Word, etc.) ¿**seguro**?
- c) Hoja de cálculo (LibreOffice Calc, MS Excel, etc.) ¿**seguro**?

# Volumen y complejidad de datos



¿Y cómo almacenarías estos datos?

Información de **PEDIDOS**. Hasta el momento se registran los siguientes datos en distintos documentos:

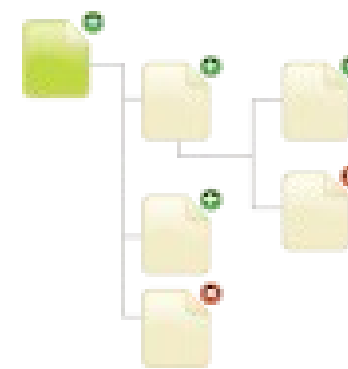
- Para cada **cliente**: número de cliente (único), direcciones de envío (varias por cliente), saldo, límite de crédito (depende del cliente, pero en ningún caso debe superar los 18.000 euros), descuento.
- Para cada **artículo**: código de artículo (único), fábricas que lo distribuyen, existencias de ese artículo en cada fábrica, descripción del artículo.
- Para cada **pedido**: cada pedido tiene una cabecera y el cuerpo del pedido. La cabecera está formada por el número de cliente, dirección de envío y fecha del pedido. El cuerpo del pedido son varias líneas y en cada línea se especifica el número del artículo pedido y la cantidad.
- Para cada **fábrica**: número de la fábrica (único) y teléfono de contacto. Y se desean ver cuántos artículos (en total) provee la fábrica. También, por información estratégica, se podría incluir información de fábricas alternativas respecto de las que ya fabrican artículos para esta empresa.

Nota: Una dirección se entenderá como número, calle, ciudad y código postal. Una fecha incluye hora.

# Arquitectura de niveles

Todo proceso **sencillo** se puede hacer **sin planificación** pero, si tratamos muchos datos o éstos se relacionan de manera compleja, necesitamos una **metodología para poder resolverlo de forma eficaz y eficiente**.

Esto, aplicado a las bases de datos, nos lleva a tratar la creación de la base de datos mediante **NIVELES DE ABSTRACCIÓN**. El comité ANSI/SPARC define en 1975 una arquitectura para los sistemas gestores de bases de datos.



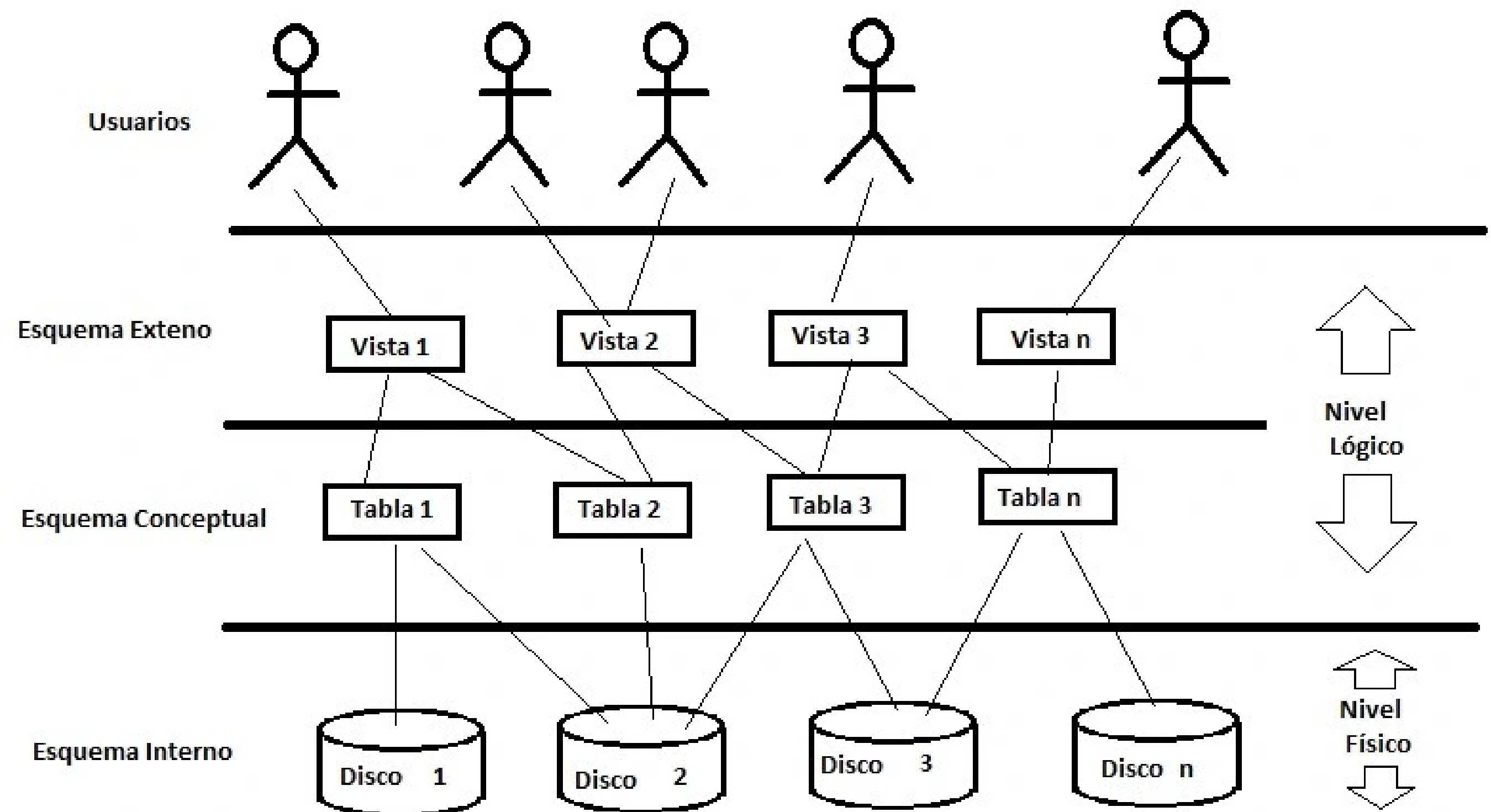
Consta de tres niveles:

- **Nivel externo o de visión.** Se compone de las distintas aplicaciones basadas en vistas de la base de datos. Es lo que ven los usuarios finales.
- **Nivel conceptual.** Se compone de las distintas tablas con sus atributos. Es el nivel que conocen los programadores.
- **Nivel interno o físico.** Define qué discos y archivos componen la base de datos y qué hay en cada uno de ellos. Solo acceden a este nivel los administradores.

# Arquitectura de niveles

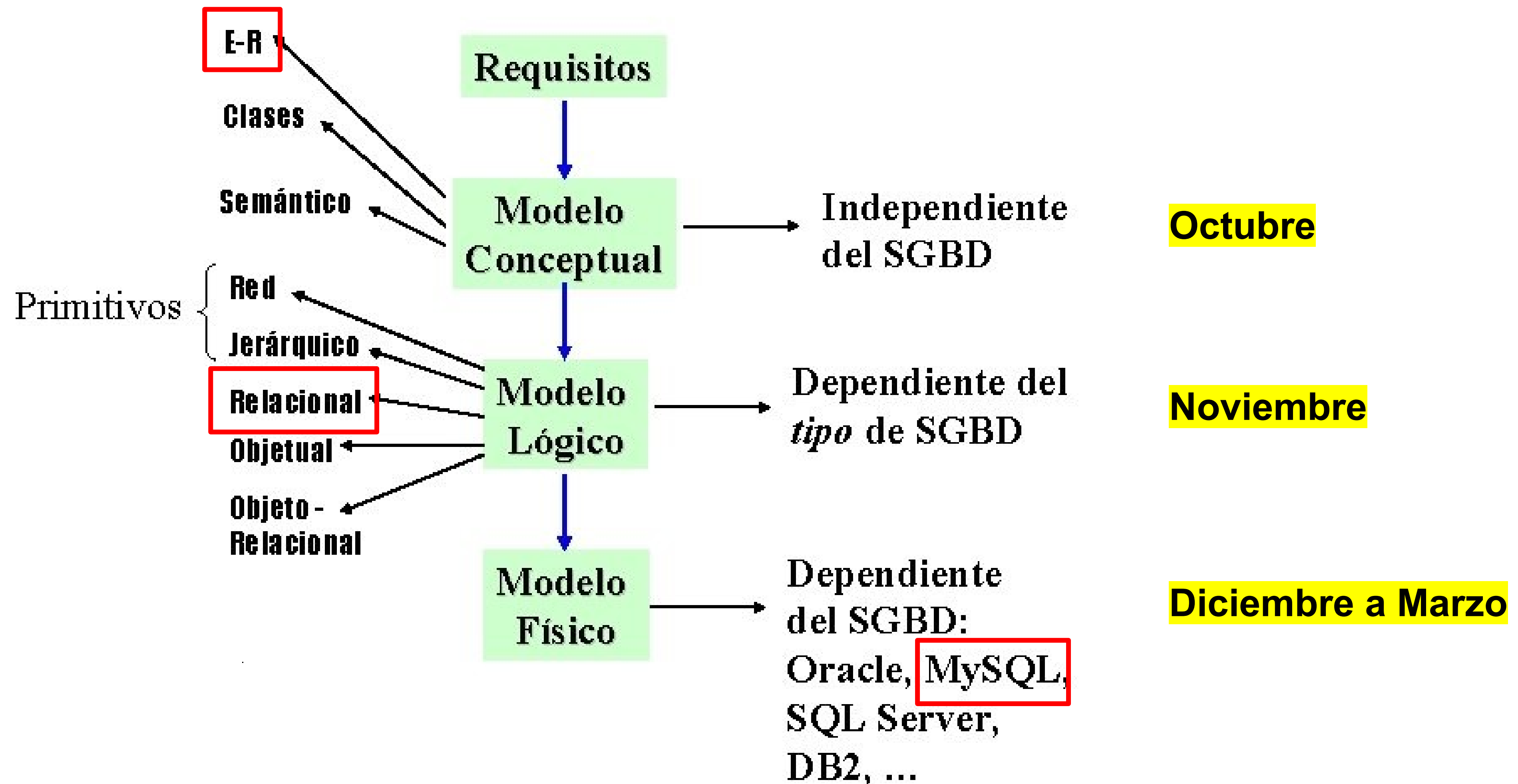
La ventaja de esta arquitectura en niveles es que proporciona independencia lógica y física de los datos respecto a las aplicaciones:

- **Independencia lógica:** Se pueden realizar cambios en el nivel conceptual (añadir tablas o atributos) sin que sea necesario reescribir todas las aplicaciones.
- **Independencia física:** Es posible modificar la ubicación de los ficheros que contienen los datos sin que se vean afectadas las aplicaciones.



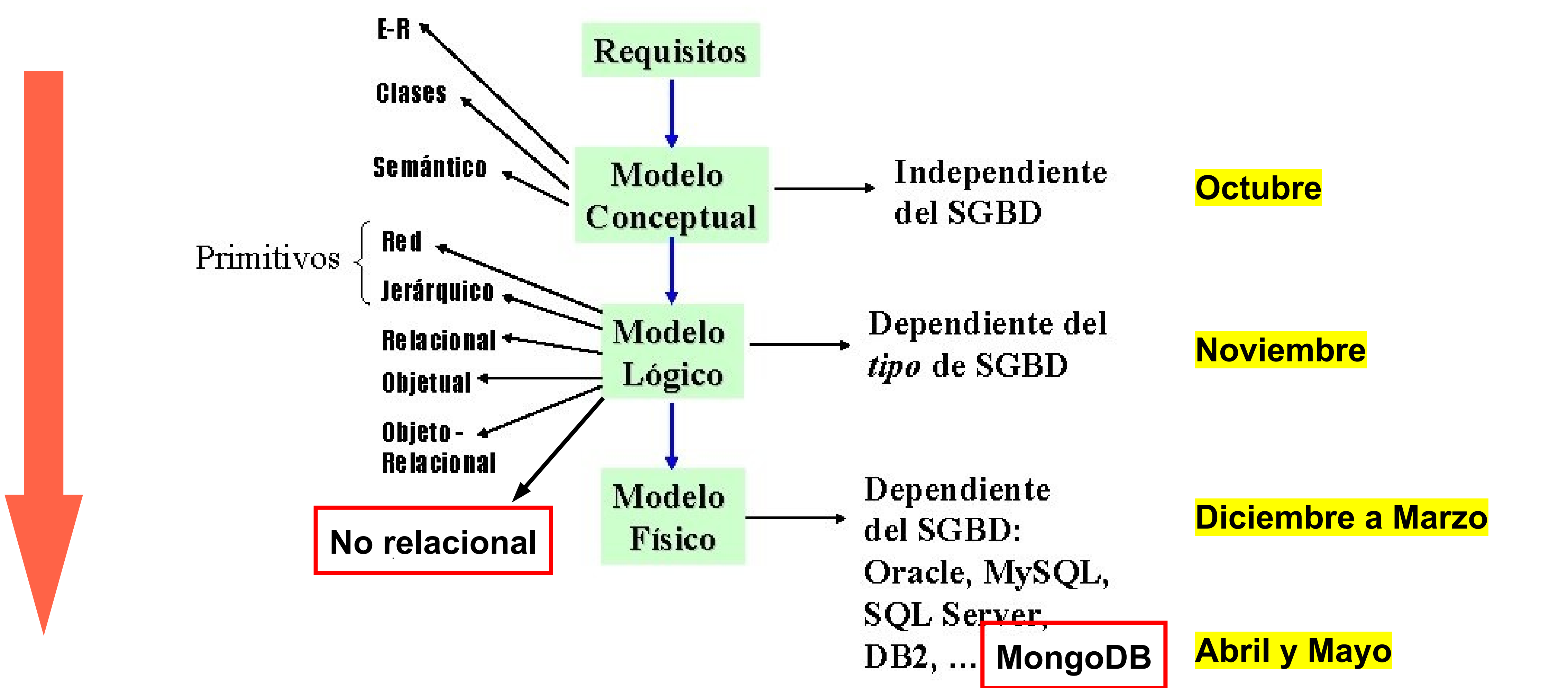
# Proceso de creación de una BD relacional

El módulo de BD consiste en recorrer este esquema de arriba a abajo, paso por paso, para **diseñar una BD de manera correcta desde cero**.



# Proceso de creación de una BD no relacional

El módulo de BD consiste en recorrer este esquema de arriba a abajo, paso por paso, para **diseñar una BD de manera correcta desde cero**.





## **6. ACTIVIDADES PROPUESTAS**

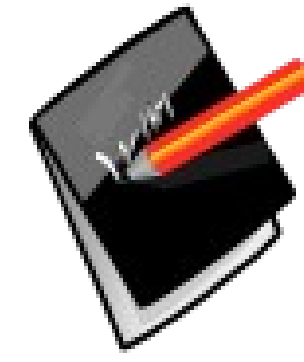
# Actividades propuestas



Estas actividades son opcionales y no evaluables pero es recomendable hacerlas para un mejor aprendizaje de la asignatura.

Mira los ejercicios propuestos que encontrarás en el “Aula Virtual”.

## **7. BIBLIOGRAFÍA**



# Recursos

- Desarrolloweb. <https://desarrolloweb.com/articulos/introduccion-base-datos.html>
- Tema 0: Introducción a bases de datos (2014/2015). Universidad de Sevilla. <https://www.cs.us.es/cursos/bd-2014/BD-Tema0.pdf>
- 14748 - Bases de Datos – Biblioteconomía. 2003-2004. Tema 1: Introducción a las Bases de Datos. Universidad de Valencia (UV). <http://informatica.uv.es/docencia/biblioguia/BD/ficheros/tema1.pdf>
- Bases de datos y sistemas de información (11548). Universidad Politécnica de Valencia (UPV). <https://www.studocu.com/es/course/universitat-politecnica-de-valencia/bases-de-datos-y-sistemas-de-informacion/1267405>
- Carlos Manuel Martí Hernández. Bases de dades. Desenvolupament d'aplicacions multiplataforma i Desenvolupament d'aplicacions web. Creative Commons. Departament d'Ensenyament, Institut Obert de Catalunya. Dipòsit legal: B. 12715-2016. <https://ioc.xtec.cat/educacio/recursos>
- SQLite. <https://www.sqlite.org/index.html>
- Softzone. Conoce SQLite. <https://www.softzone.es/programas/lenguajes/que-es-sqlite/>
- Hostgator. <https://www.hostgator.mx/blog/sqlite-que-es-y-diferencias-con-mysql/>

