

# Los datos: conceptos introductorios

# Índice

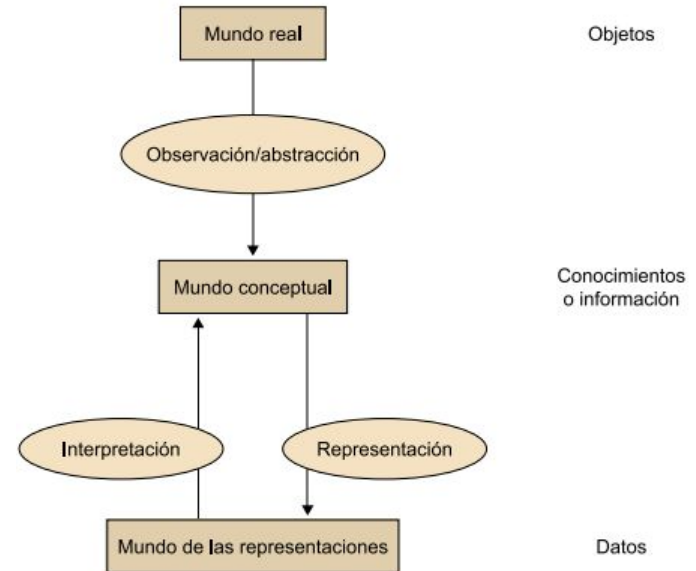
1. Los tres mundos: el real, el conceptual y el de las representaciones
2. El mundo conceptual: entidades y atributos
3. El mundo de las representaciones
4. La memoria persistente

# 1. Los tres mundos: el real, el conceptual y el de las representaciones

# Los tres mundos

Para tener un marco donde situar los términos y los conceptos que explicaremos en la asignatura, distinguiremos tres ámbitos diferentes:

- El mundo **real** con los **objetos** de nuestro interés.
- El mundo de las **conceptualizaciones** lógicas.
- El mundo de las **representaciones** informáticas.



## La realidad: los objetos

El mundo **real**, la parte de la realidad que nos interesa, es lo que percibimos con nuestros sentidos y está compuesto por objetos concretos, físicos o no.

Ejemplos de objetos concretos:

- El enfermo Joan Garcia
- La cama 34 de la segunda planta
- El almacén de Soria
- El camión B-3452-AG
- La alumna María Pi
- La asignatura Química I
- La devolución de un pedido concreto

## Las concepciones: la información

El conjunto de los conocimientos obtenidos a partir de la observación de un mundo real se denomina **mundo conceptual** o **mundo de las concepciones**.

En la esfera de las concepciones construimos un modelo abstracto, conceptual, del mundo real, y esto nos ayuda a razonar y a expresarnos.

El proceso de **observación/abstracción** es básicamente un proceso para modelizar la estructura, las propiedades y el funcionamiento de la realidad.

La **información** es un **conocimiento transmisible**, es decir, que se puede representar.

## Las representaciones: los datos

Para poder trabajar con conocimientos (que pertenecen al mundo mental) y poderlos comunicar, necesitamos proyectar los pensamientos en el exterior representante físicamente de alguna manera. Este es el mundo de las **representaciones**.

Damos el nombre de **datos** a las representaciones físicas de los conocimientos que tenemos de los objetos del mundo real.

El paso de los conocimientos a los datos, o de una concepción a una representación informática, no es automático. Es un proceso humano: un proceso de diseño.

Acabamos de ver el camino que nos lleva de la realidad a los conocimientos, y de éstos a los datos o representaciones. Pero nos hará falta poder interpretar la representación. El proceso inverso al de representación se conoce como **interpretación**.

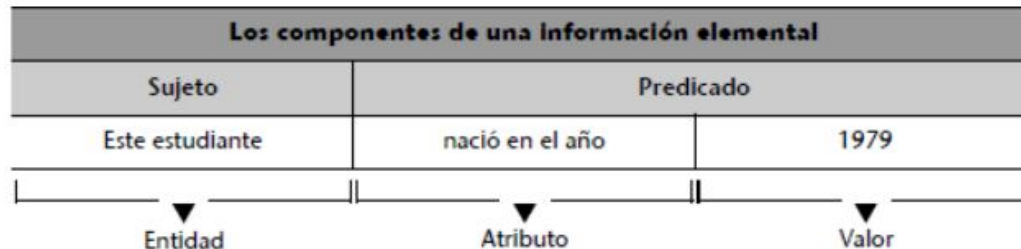
## 2. El mundo conceptual: entidades y atributos



# Entidades, atributos y valores

En términos lingüísticos: una **información** (un conocimiento elemental) se puede expresar con un **sujeto** (el estudiante concreto) y un **predicado** ("nació en 1979"). El predicado es formado por el **verbo** y el **complemento**.

Desde un punto de vista informático: llamamos **entidades** los objetos que conceptualizamos como distinguibles unos de otros (es decir, que son identificables) y de los que nos interesan algunas propiedades. El *predicado* es la propiedad descrita, y sus dos partes, *verbo* y *complemento*, las llamamos **atributo** (año de nacimiento) y **valor** (1979), respectivamente.

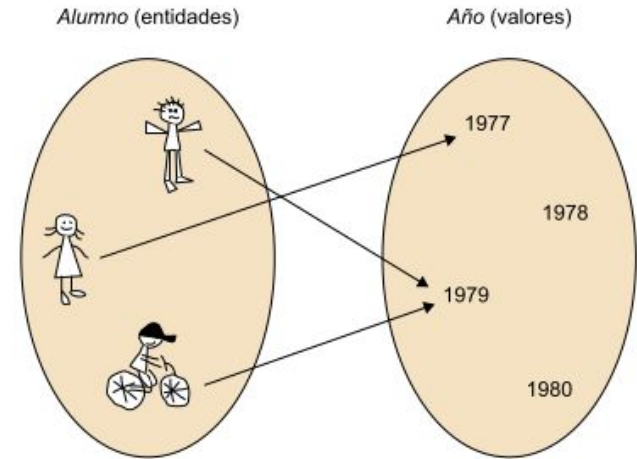


## El marco de la teoría de conjuntos

Usando la teoría de conjuntos, podemos ver la correspondencia entre los estudiantes y los años como una **aplicación** (en el sentido de las matemáticas) del conjunto de los estudiantes sobre el conjunto de los años:

- Cada estudiante tiene un solo año de nacimiento.
- Diferentes estudiantes pueden tener el mismo año de nacimiento.

El atributo *año de nacimiento*



## Dominios, identificadores y claves

El conjunto de todos los valores válidos, o legales, que puede llegar a tener un atributo, recibe el nombre de **dominio** del atributo.

Puede ocurrir que el valor de un atributo determinado de alguna entidad individual sea desconocido o no exista. Entonces diremos que el dominio de ese atributo **acepta el valor nulo**.

Los atributos que concebimos como aplicaciones inyectivas<sup>1</sup> se denominan **identificadores**.

Todo atributo o conjunto de atributos que permite identificar las entidades individuales recibe el nombre de **clave**.

Dado que el atributo es una aplicación entre conjuntos, a cada entidad le puede corresponder como máximo un solo valor. En consecuencia, **un atributo no podrá ser multivalor** (o multivaluado).

<sup>1</sup> Una aplicación inyectiva es aquella en la que cada elemento del conjunto imagen le corresponde un elemento del conjunto origen como máximo.

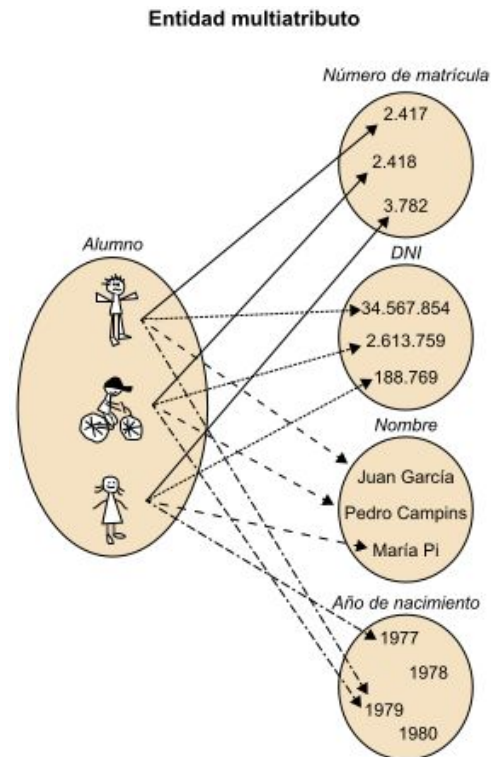
# 3. El mundo de las representaciones

## La representación tabular

La representación informática de una información elemental se llama **dato**.




La figura de la izquierda es una **representación gráfica**, no informatizada, de la información de los estudiantes, utilizando la representación clásica de las aplicaciones en la teoría de conjuntos, para una entidad multiatributo.

Con tantas flechas y conjuntos, no resulta muy cómodo para ser procesada ....



## La representación tabular

Resulta mucho más sencilla una representación **tabular** con una fila para cada entidad individual y una columna para cada atributo.

Representación tabular de la información en la figura 5				
Alumno	Número de matrícula	DNI	Año de nacimiento	Nombre
	2.417	34.567.854	1979	Juan García
	3.782	188.769	1977	María Pi
	2.418	2.613.759	1979	Pedro Campins

# Ficheros, registros y campos

Un **archivo de datos** es una representación informática equivalente a la representación tabular:

- La representación de una entidad, el equivalente a una *fila* de la tabla, recibe el nombre de **registro**.
- La representación del *valor* de un atributo de una entidad se denomina **campo**.

El conjunto de campos constituye el registro, y el conjunto de registros constituye el archivo.

Fichero de estudiantes				
<i>número de matrícula</i>	<i>DNI</i>	<i>año de nacimiento</i>	<i>nombre</i>	Nombre o cabecera de los campos
2.417	34.567.854	1979	Juan García	Registros
3.782	188.769	1977	María Pi	
2.418	2.613.759	1979	Pedro Campins	

# Ficheros interrelacionados

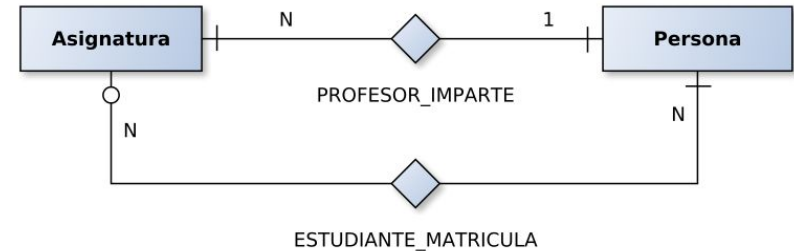
De momento, en este módulo introductorio daremos el nombre de **base de datos (BD)** a un conjunto de ficheros de datos interrelacionados.

Supongamos que los tipos de objetos de nuestro interés son estudiantes, asignaturas y profesores.

Podremos representar estas entidades mediante tres ficheros, uno para cada entidad, con los campos correspondientes a los atributos. Pero falta la información que permite interrelacionar las entidades entre sí.

- 1) Todo estudiante puede cursar más de una asignatura y toda asignatura puede ser cursada por muchos estudiantes.
- 2) Toda asignatura es dada por un solo profesor, pero cada profesor puede dar varias asignaturas.

Interrelaciones entre entidades



Los softwares especializados en estos conjuntos complejos de datos reciben el nombre de **Database Management Systems (DBMS)** o Sistemas de gestión de BD (SGBD).



## Acceso a los datos

Hay dos formas básicas de acceso a los datos:

- Acceso **secuencial**: el acceso a un registro presupone el acceso previo a todos los registros anteriores.
- Acceso **directo**: el acceso a un registro no requiere el acceso previo a todos los registros anteriores.

Adicionalmente, distinguimos entre:

- Acceso por **valor**: el acceso por valor nos lleva al registro en función del valor de alguno de sus atributos, sin tener en cuenta la posición que ocupa el registro.
- Acceso por **posición**: el acceso por posición nos lleva a un lugar - una posición - donde hay un registro de datos, sin tener en cuenta el contenido.

## Nivel lógico y nivel físico

En el mundo de las representaciones informáticas, se suele distinguir dos puntos de vista o niveles: el **nivel físico**, cuando es necesario considerar la realización física, y el **nivel lógico**, cuando no hay que conocerla.

- **Nivel lógico:** por ejemplo, el programador de aplicaciones que trabaja con un lenguaje de alto nivel como el C, el C ++, Java, etc., puede ver o imaginar que los archivos están formados simplemente por registros, uno tras el otro, y que contienen campos con letras y números. No ve, ni tiene que conocer, la realización física que quizás constará de encadenamientos de registros físicos (cada uno con varios registros lógicos), marcas separadoras entre campos, compresión de datos, índices, etc.
- **Nivel físico:** entramos en el nivel físico cuando tenemos que considerar la realización física. Un programador de software básico (SGBD, SO, etc.), un diseñador físico de una BD, un técnico de sistemas que administra una BD, etc., tienen que entrar en el nivel físico.

# 4. La memoria persistente

## Justificación de la utilización de la memoria persistente

La necesidad de almacenar los datos nos obliga a utilizar memorias **no volátiles** con soportes permanentes, como los discos magnéticos, los discos ópticos, memorias flash, cintas, etc. Además de la no volatilidad, esta memoria también tiene otras propiedades: **gran capacidad** y el **bajo precio por byte**.

El principal inconveniente de estos periféricos de almacenamiento persistente es el **tiempo de acceso**, que es significativamente más lento que el de la memoria interna.

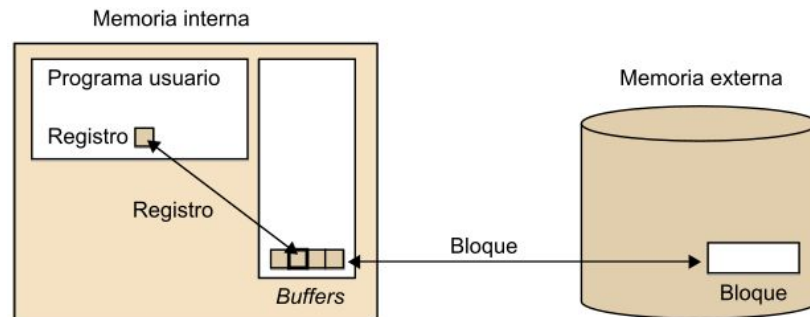
## Esquema de E/S

El **bloque** es lo que se lee o se escribe de vez en una sola operación física de E/S (entrada/salida).

En el mundo de las BD se utiliza a menudo el término **página** como sinónimo de bloque.

Del mismo modo que la unidad de transferencia entre la memoria persistente y los buffers es el bloque, la unidad de transferencia entre éstas y el programa de usuario es el **registro**.

Esquema básico del E/S



## Tiempo de acceso

El tiempo necesario para completar una operación física de lectura o escritura de un bloque a una memoria persistente móvil consta de dos partes:

- El tiempo de **acceso** es el tiempo necesario para que el mecanismo se coloque al inicio del bloque que se debe leer o escribir.
- El tiempo de **transferencia** es el tiempo necesario para leer o escribir el bloque.

---

Universitat Oberta  
de Catalunya

---

Uoc

 UOC.universitat  
 @UOCuniversitat  
 UOCuniversitat

---