



Universidad de Granada

decsai.ugr.es

Fundamentos de Bases de Datos

Grado en Ingeniería Informática

Tema 3: Modelos de datos. El modelo Relacional



DECSAI

**Departamento de Ciencias de la
Computación e Inteligencia Artificial**

- 1. Definición de modelos de datos**
- 2. El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional**
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad**
- 3. Otros modelos de datos**
 - 3.1. Modelo jerárquico**
 - 3.2. Modelo en red**



- 1. Definición de modelos de datos**
- 2. El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
- 3. Otros modelos de datos**
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



Modelización semántica es la estructuración de la información.

Problema tipo

- Mundo real:

- Delimitación objetivos.
- Selección de datos.
- Hipótesis semánticas.
- Organización de los datos a almacenar.

- Esquema inicial:

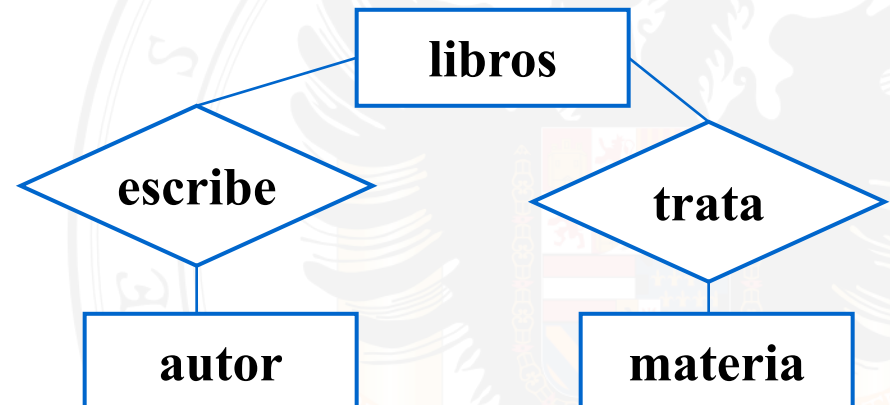
- Datos operativos.
- Atributos.
- Conexiones.
- Restricciones.

- Biblioteca



- Resultado

- Libros: título, isbn, editorial....
- Autor: nombre, nacionalidad,....
- Materia: código, descripción....



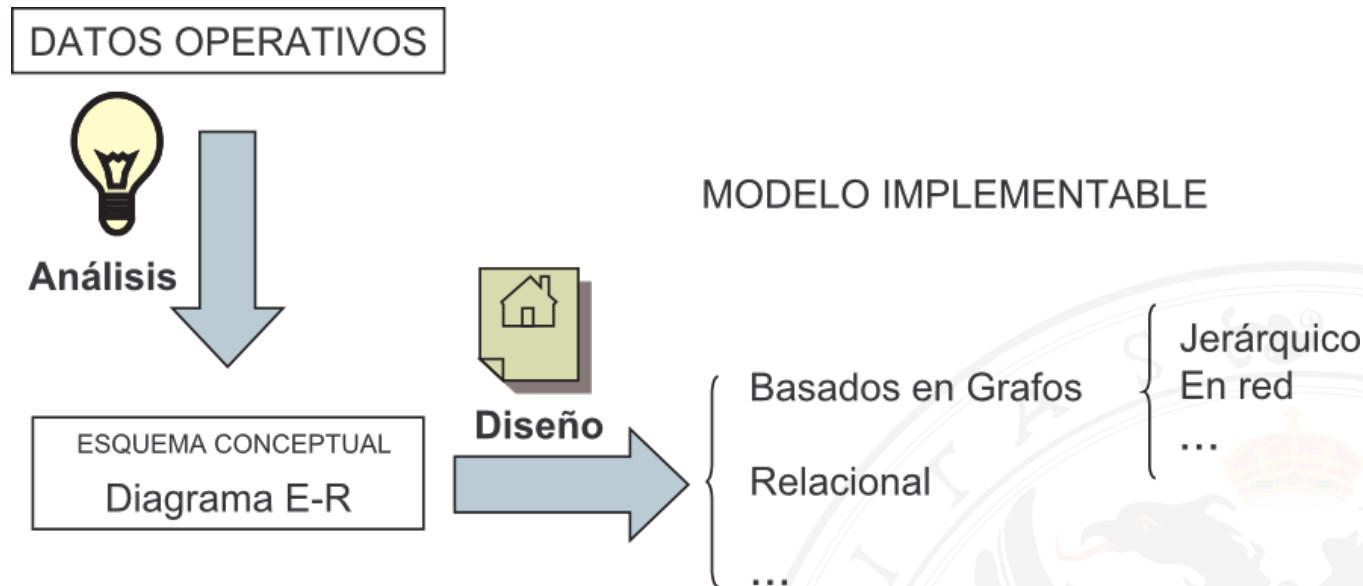
Definición formal (de modelo de datos)

- **Mecanismo formal** para representar y manipular información de manera general y sistemática.
- Debe constar de:
 - 1.- Notación para describir **datos**.
 - 2.- Notación para describir **operaciones**.
 - 3.- Notación para describir **reglas de integridad**.

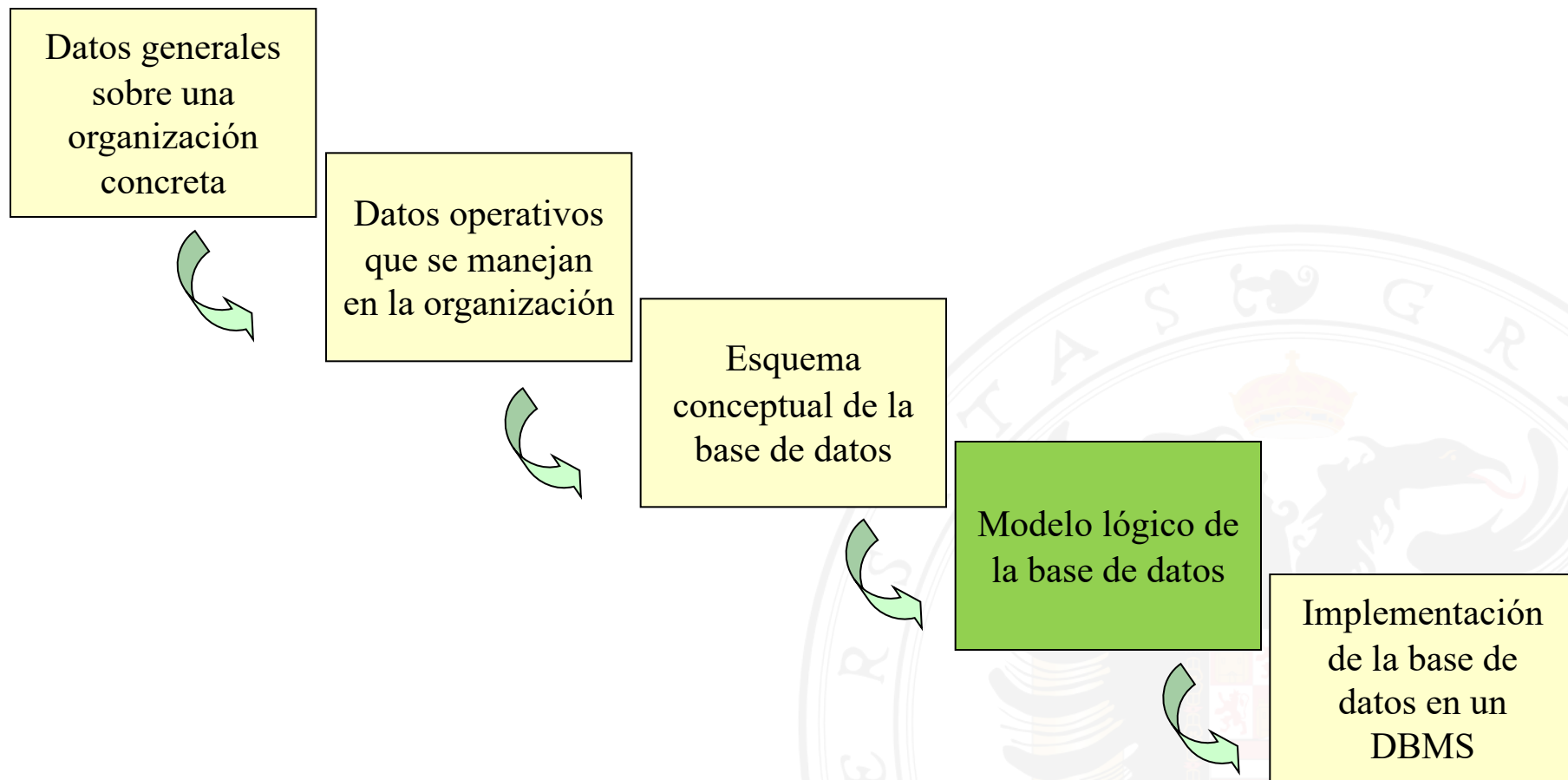
Historia

- 1^{er} modelo es el relacional (Codd, 1970).
- Se recuperan los modelos basados en grafos (1974).
- El modelo E/R (Chen, 1975), otros modelos semánticos.
- Modelos orientados a objetos (1983, 1986,...).
- Modelos lógicos (1986...).

Proceso de análisis y diseño de una BD



- Una vez hemos llevado a cabo el proceso de análisis de datos y obtenido el esquema conceptual y lógico de nuestra BD, es necesario implantarla en un sistema a través de un **proceso de diseño**, que nos permitirá trasladar la estructura actual a un modelo de datos implementable.



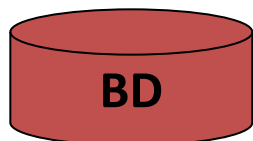
Datos generales sobre una organización concreta

Datos operativos que se manejan en la organización

Esquema conceptual de la base de datos

Modelo lógico de la base de datos

Implementación de la base de datos en un DBMS



- **Modelado lógico:** Trasladamos el esquema conceptual a un **esquema lógico** en función de una estructura implementable.
- **Ejemplo:** Tabla LIBROS

Título	ISBN	Editorial	...
Introducción a las BD	1234-1234	Thomson	...
Cálculo para todos	4321-4321	Delta	...
...

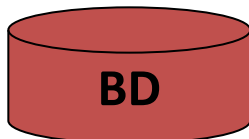
Datos generales sobre una organización concreta

Datos operativos que se manejan en la organización

Esquema conceptual de la base de datos

Modelo lógico de la base de datos

Implementación de la base de datos en un DBMS



- Lo **implementamos** en un sistema comercial:

```
CREATE TABLE LIBROS (
    titulo char(45) NOT NULL,
    ISBN char(10) PRIMARY KEY,
    editorial char(30) REFERENCES ...
);
```

Necesidad de modelos de datos

- Cada esquema se describe utilizando un **lenguaje de definición de datos**.
- Este lenguaje es de muy bajo nivel: muy ligado al SGBD.
- Hacen falta otros mecanismos de más alto nivel que permitan describir los datos de una forma no ambigua y entendible por los usuarios implicados en cada paso del proceso de implantación.

Objetivo

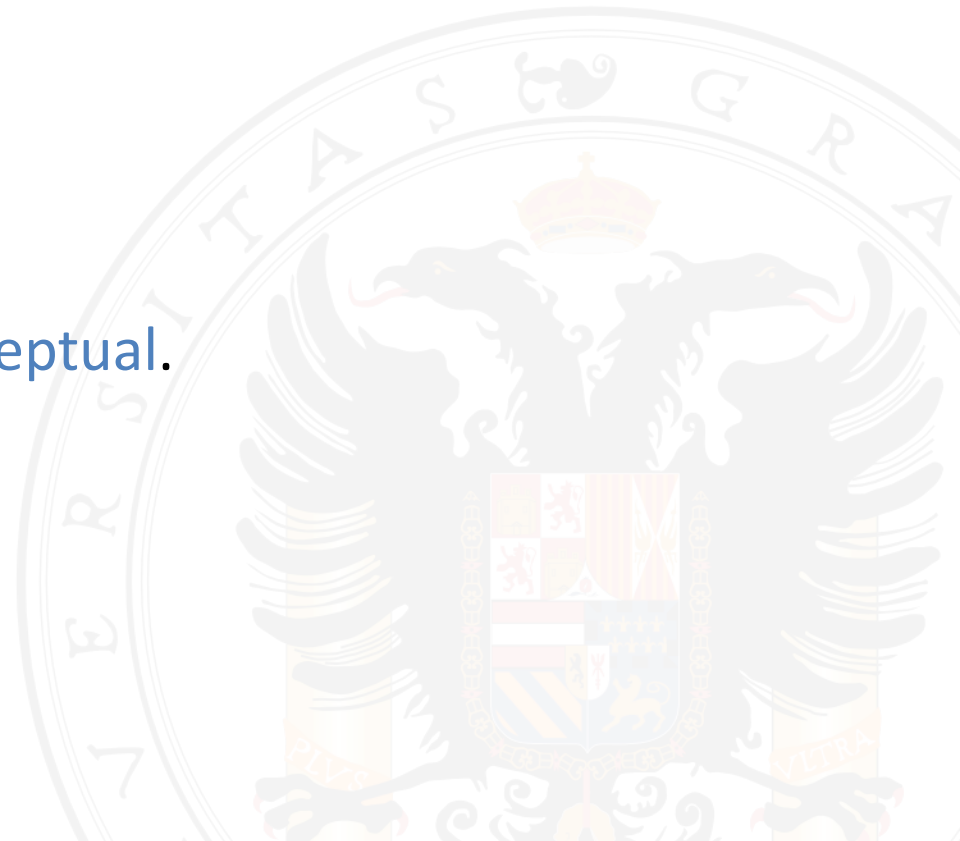
- Describir modelos que representen los datos y que los describan de una forma entendible y manipulable.
- En relación con la Arquitectura ANSI/SPARC:
 - Nivel externo:
 - Modelo de datos externo.
 - Nivel conceptual:
 - Modelo de datos conceptual.
 - Nivel interno:
 - Modelo de datos interno.

Clasificación

- Basados en registros.
- Basados en objetos.
- Físicos.

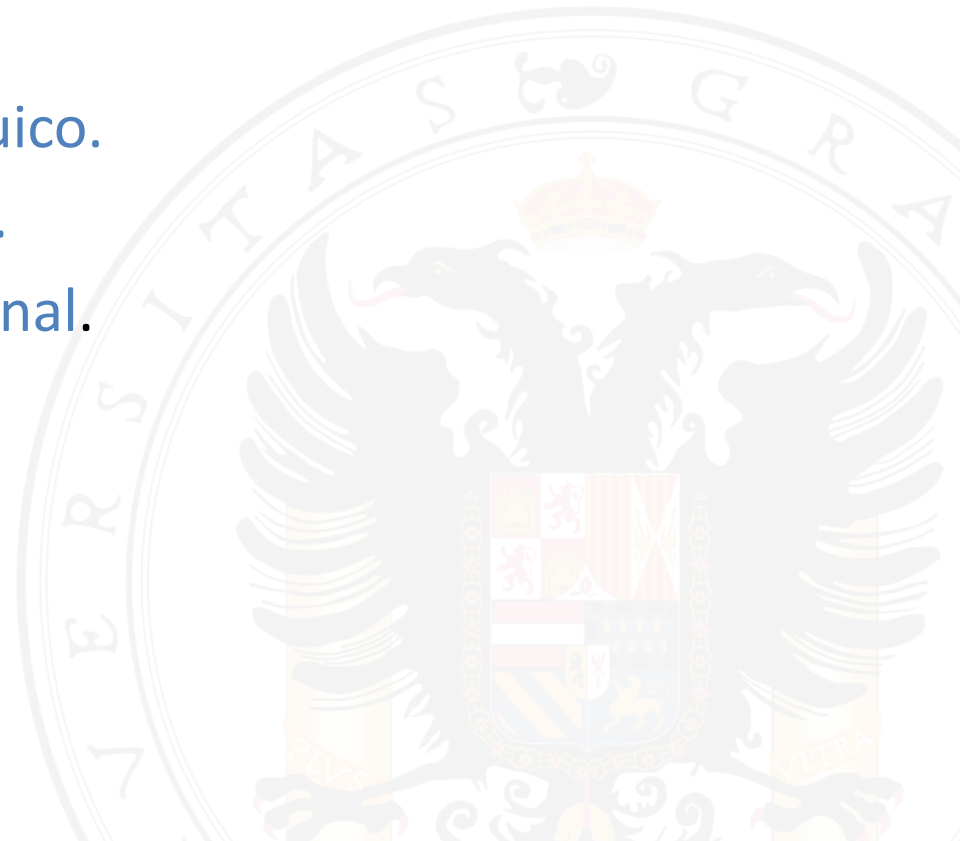
Utilización

- Los dos primeros:
 - Nivel externo y conceptual.
- Físicos:
 - Nivel interno.



Modelos de datos basados en registros

- Modelo de datos **jerárquico**.
- Modelo de datos **en red**.
- Modelo de datos **relacional**.



1. Definición de modelos de datos
2. **El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
3. Otros modelos de datos
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



1. Definición de modelos de datos
2. **El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
3. Otros modelos de datos
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



Introducido por Codd

- El modelo relacional **abarca** tres ámbitos distintos de los datos:
 1. **Estructuras para almacenarlos:** El usuario percibe la información de la base de datos estructurada en tablas.
 2. **Integridad:** Las tablas deben satisfacer ciertas condiciones que preservan la integridad y la coherencia de la información que contienen.
 3. **Consulta y manipulación:** Los operadores empleados por el modelo se aplican sobre tablas y devuelven tablas.
- La **tabla** es la estructura lógica de un sistema relacional. A nivel físico, el sistema es libre de almacenar los datos en el formato más adecuado (archivo secuencial, archivo indexado, listas con punteros, etc.).

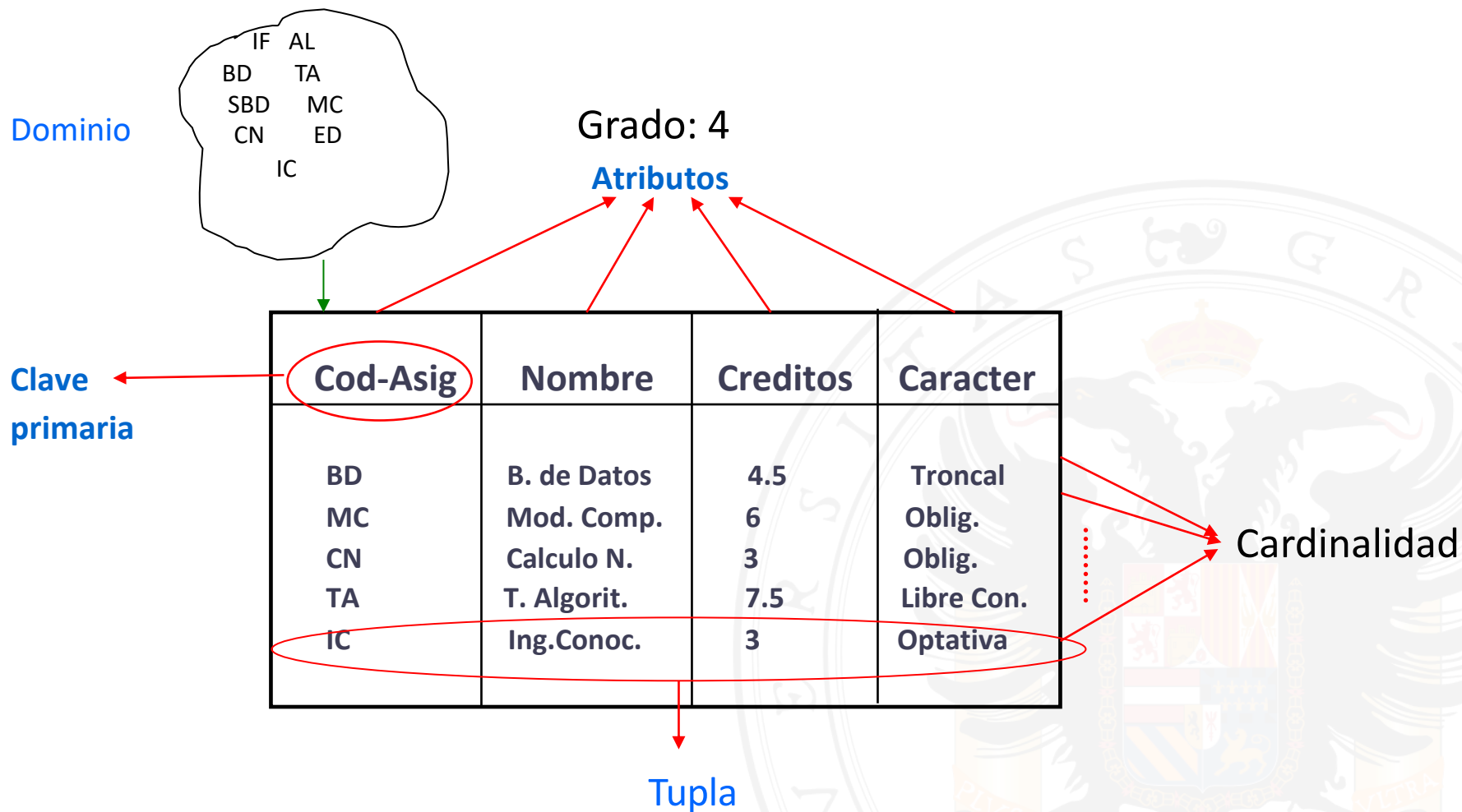
- 1.** Definición de modelos de datos
- 2.** El modelo de datos relacional. Estructura de datos. Integridad
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
- 3.** Otros modelos de datos
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



- **Atributo:** Cualquier elemento de información susceptible de tomar valores. Notación: A_i , $i=1,2,\dots$
- **Dominio:** Rango de valores donde toma sus datos un atributo. Se considera finito. Notación: D_i , $i=1,2,\dots$
- **Relación:** Dados los atributos A_i , $i=1,2,\dots,n$ con dominios D_i , $i=1,2,\dots,n$, no necesariamente distintos, definimos relación asociada a $A_1..A_n$, y lo notaremos por $R(A_1..A_n)$, a cualquier subconjunto del producto cartesiano $D_1 \times D_2 \times \dots \times D_n$.

- **Tupla:** Cada una de las filas de una relación.
- **Cardinalidad de una relación:** Número de tuplas que contiene. Es variable en el tiempo.
- **Esquema de una relación R :** Atributos de la relación junto con su dominio, $A_1: D_1, \dots, A_n: D_n$
- **Grado de una relación:** Número de atributos de su esquema (n). Invariable en el tiempo.
- **Instancia de una relación:** Conjunto de tuplas $\{(x_1, x_2, \dots, x_n)\} \subseteq D_1 \times D_2 \times \dots \times D_n$ que la componen en cada momento.

Ejemplo: Asignaturas de una titulación



1. Definición de modelos de datos
2. **El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - **Propiedades de la estructura de datos relacional**
 - Notación
 - 2.2. Restricciones o reglas de integridad
3. Otros modelos de datos
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red

Condición de normalización

- Todos los **valores de los atributos** de una relación son **atómicos**.
- Valor atómico es un valor no estructurado.
- Cuando una relación cumple la condición de normalización se dice que está en **Primera Forma Normal**.

Consecuencias

- No hay valores tipo conjunto.
- No hay valores tipo registro.
- No hay valores tipo tablas.

Problema

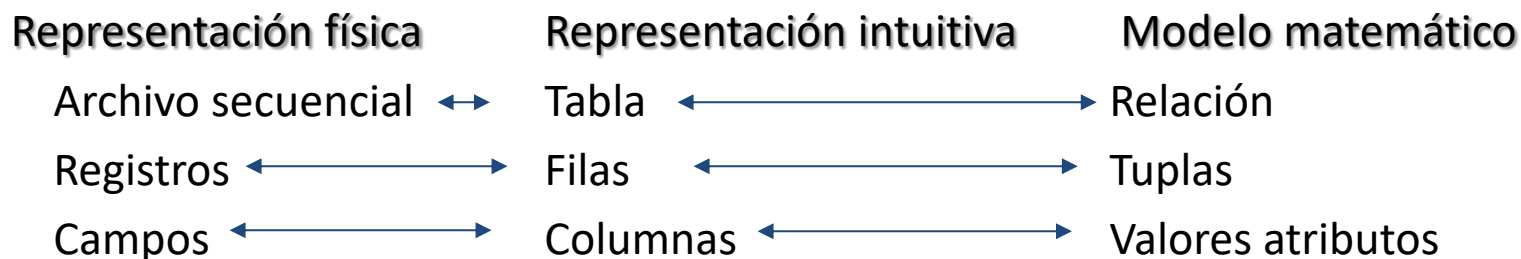
Todas las **representaciones** son **extensivas** → no se puede representar información del tipo “el valor del atributo asignaturas de un alumno es: FBD, ALG, LD”.

Consecuencias de la definición

- No hay tuplas duplicadas:
 - Por la definición conjuntista de relación.
- No hay orden en las filas ni en los atributos:
 - Al no estar ordenados ni los atributos ni las filas (conjuntos) el acceso es por nombre de atributo y valor.
- Varias instancias representan la misma relación.

A	B	C	D	E
<i>a1</i>	<i>b1</i>	<i>c1</i>	<i>d1</i>	<i>e1</i>
<i>a1</i>	<i>b2</i>	<i>c2</i>	<i>d2</i>	<i>e1</i>
<i>a2</i>	<i>b1</i>	<i>c3</i>	<i>d3</i>	<i>e1</i>
<i>a2</i>	<i>b1</i>	<i>c4</i>	<i>d3</i>	<i>e1</i>
<i>a3</i>	<i>b2</i>	<i>c5</i>	<i>d1</i>	<i>e1</i>

A	B	C	D	E
<i>a2</i>	<i>b1</i>	<i>c4</i>	<i>d3</i>	<i>e1</i>
<i>a2</i>	<i>b1</i>	<i>c3</i>	<i>d3</i>	<i>e1</i>
<i>a1</i>	<i>b2</i>	<i>c2</i>	<i>d2</i>	<i>e1</i>
<i>a3</i>	<i>b2</i>	<i>c5</i>	<i>d1</i>	<i>e1</i>
<i>a1</i>	<i>b1</i>	<i>c1</i>	<i>d1</i>	<i>e1</i>



- **Esquema de una base de datos relacional**
 - Colección de esquemas de relaciones junto con sus restricciones de integridad.
- **Instancia o estado de una base de datos**
 - Colección de instancias de relaciones que verifican las restricciones de integridad.
- **Base de datos relacional**
 - Instancia de una base de datos junto con su esquema.

1. Definición de modelos de datos
2. **El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - **Notación** hasta aquí
 - 2.2. Restricciones o reglas de integridad
3. Otros modelos de datos
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red

Notación a utilizar

- **Relación:** R, S, T, \dots
- **Atributos:** A, B, \dots
- **Esquema** de relación: $R[A_1, A_2, \dots, A_n]$
- **Instancia** de relación R : r, \dots
- **Tuplas** de una instancia: $x_1, x_2, \dots \in r$
- **Valor** de un atributo A_i en una tupla x_j : $x_j[A_i]$ ó A_{ij}

Valores nulos

- Algunas veces no se conoce el valor de un atributo para una determinada tupla. En esos casos a ese atributo de esa tupla se le asigna un valor nulo (NULL).
- Un valor nulo puede ser:
 - Un **valor desconocido**.
 - Un **atributo no aplicable**.
- En cualquier caso, ese valor es un valor más de todos los dominios de la base de datos.

1. Definición de modelos de datos
2. **El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
3. Otros modelos de datos



Restricciones o reglas de integridad

- Son condiciones para preservar la semántica de una base de datos.
- Asociadas a tablas:
 - $0 \leq \text{edad} \leq 100$
 - $\text{créditos} > 0$
 - $\text{carácter} \in (\text{'troncal'}, \text{'obligatoria'}, \text{'optativa'}, \dots)$
- Asociadas a la base de datos:
 - $\text{imparte.NRP} \in \text{profesor.NRP}$
(un profesor inexistente no puede impartir una asignatura)
 - $\text{cod_asig} \neq \text{nulo}$
(siempre debe conocerse el código de una asignatura)

- **Superclave:** Cualquier conjunto de atributos que identifica unívocamente a cada tupla de una relación.
- **Clave de una relación:** superclave minimal.
- Por **ejemplo**, en la relación Asignaturas:
 - El conjunto de atributos {**Cod_Asig**, **Nombre**} identifica unívocamente cada tupla.
 - Sin embargo, no es minimal y no puede considerarse como una clave.
 - **Cod_Asig** por sí sola, es una clave.

- En una relación dada puede que más de un conjunto de atributos puedan ser elegidos como clave. Estos conjuntos de atributos se llaman **claves candidatas**.
- Cuando hay más de una clave candidata, hay que seleccionar una como principal. Esta clave recibe el nombre de **clave primaria** de la tabla.

Clave candidata (CC) y primaria (CP o PK) (definición formal)

- Sea $R[A_1, A_2, \dots, A_n]$, $PK \subseteq \{A_1, A_2, \dots, A_n\}$ se denomina clave candidata sii:
 - **Unicidad:** $\forall r$ instancia de R y $\forall t_1, t_2 \in r \ t_1 \neq t_2 \Rightarrow t_1[PK] \neq t_2[PK]$
 - **Minimalidad:** No existe $PK' \subset PK$ que verifique la unicidad.
- Una **clave candidata** es un atributo o conjunto de atributos que identifican a cada tupla en la relación y que, además, no existe un subconjunto de ellos que también identifiquen a cada tupla de la relación.
- Una **clave primaria** es una clave candidata elegida por el diseñador.
- Si PK verifica la unicidad pero no la minimalidad se denomina **superclave**.

Conceptos generales

- Condiciones de **integridad**:
 - Normas que mantienen la corrección semántica de una base de datos.
- Nos centramos en **integridad genérica**: depende del papel que juegue un atributo en el diseño de la tabla:
 - Son metarreglas: generan reglas de integridad aplicadas a una base de datos concreta.
 - Existen la **integridad de entidad** y la **integridad referencial**.

Integridad de entidad

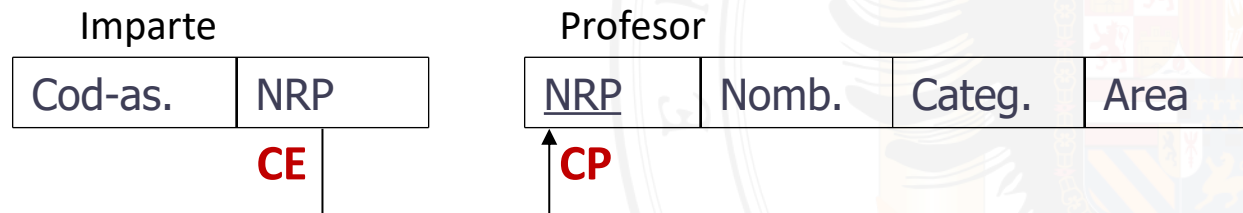
- No se debe permitir que una entidad sea representada en la base de datos si no se tiene una información completa de los atributos que son claves de la entidad.
- Un atributo que forma parte de la **clave primaria** de una tupla en una relación, **no puede tener un valor nulo**.

Clave externa

- Conjunto de atributos en una relación que es una clave en otra (o incluso en la misma) relación.
- Podemos ver una clave externa como un conjunto de atributos de una relación cuyos valores en las tuplas **deben coincidir con valores de la clave primaria** de las tuplas de la otra relación.

Clave externa (definición formal)

- Sean $R[A_1, A_2, \dots, A_n]$ y $PK \subseteq \{A_1, A_2, \dots, A_n\}$ su clave primaria, sea $S[B_1, B_2, \dots, B_n]$ y $FK \subseteq \{B_1, B_2, \dots, B_n\}$ de manera que $\text{grado}(PK) = \text{grado}(FK)$.
- FK es clave externa** de S con respecto a R si se verifica:
 - $\forall r$ instancia de R y $\forall s$ instancia de S, $\forall x \in s \Rightarrow \exists y \in r / x[FK] = y[PK]$.
 - Es decir, el dominio activo de FK debe estar incluido en el dominio activo de PK para cualquier instancia de la base de datos.

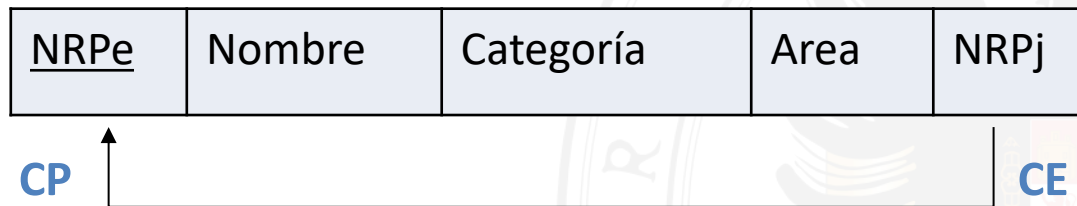


Integridad referencial

- Una base de datos en la que **todos los valores no nulos de una clave externa referencian valores reales de la clave referenciada** en la otra relación, cumple la regla de integridad referencial.
- Si una relación incluye una clave externa conectada a una clave primaria, el valor de la clave externa debe ser, o bien igual a un valor ya existente en el dominio activo de la clave primaria, o bien completamente nulo (si la semántica lo permite).
- La integridad referencial **mantiene las conexiones** en las bases de datos relacionales.

Integridad referencial

- Puede haber **más de una clave externa** en una relación.
- Puede haber una **clave externa** a la clave primaria de la **propia relación**.
- Ejemplo:



EL SGBD debe encargarse de mantener las siguientes restricciones

- La unicidad de la clave primaria y de las claves candidatas:
 - Frente a operaciones de inserción y actualización, el SGBD debe rechazar los valores introducidos que sean iguales a los presentes en la BD para los atributos que el diseñador ha definido como clave primaria y como claves candidatas.

EL SGBD debe encargarse de mantener las siguientes restricciones

- La **restricción de integridad de identidad**:

Frente a operaciones de inserción y actualización, el SGBD debe rechazar las modificaciones que vulneren la unicidad en la clave primaria y/o que asignen un valor NULO a algún atributo de la clave primaria.

EL SGBD debe encargarse de mantener las siguientes restricciones

- **Integridad referencial en inserciones:**
 - Rechazar la tupla insertada si el valor de la clave externa no concuerda en la relación referenciada para alguna tupla en el valor de su clave primaria.
 - Si el valor para la clave externa es NULO y el diseño no lo permite, habrá que rechazar también esa inserción.

EL SGBD debe encargarse de mantener las siguientes restricciones

- **Integridad referencial en actualizaciones:**
 - Si se actualiza la clave externa, rechazar la modificación si se produce alguna de las circunstancias descritas en el punto anterior.
 - Si se actualiza la clave primaria de la relación referenciada, actualizar en cadena las claves externas que la referencien (o impedir la actualización mientras existan referencias al valor anterior).
- **Integridad referencial en borrados:**
 - Si se borra la clave primaria en la relación referenciada, borrar en cascada todas las tuplas que la referencian o poner valor nulo en la clave externa de todas esas tuplas.

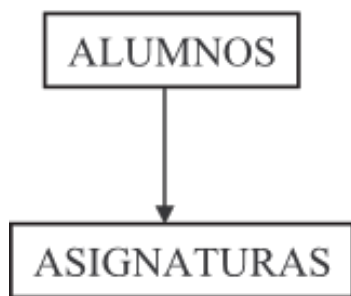
1. Definición de modelos de datos
2. El modelo de datos relacional. Estructura de datos. Integridad
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
3. **Otros modelos de datos**
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



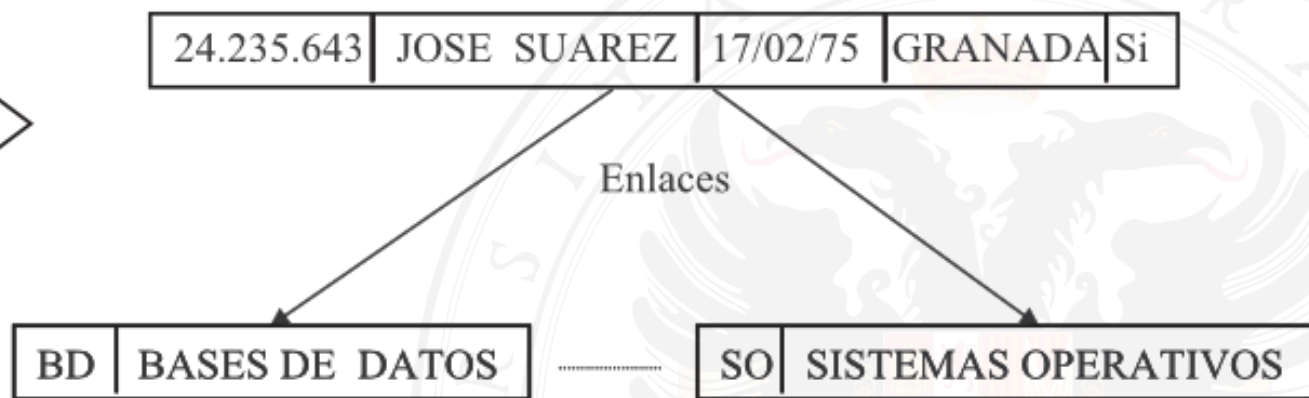
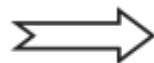
- Fue el **primero en implementarse físicamente**:
 - Nivel externo: aplicaciones Cobol o lenguaje del sistema.
 - No había interactividad:
 - Carecía de un lenguaje de consulta.
- **Estructura de datos básica**:
 - **Árbol**:
 - Registro maestro: raíz.
 - Registros secundarios: dependen de los anteriores.
- La BD es una **colección de instancias de árboles**.

Ejemplo

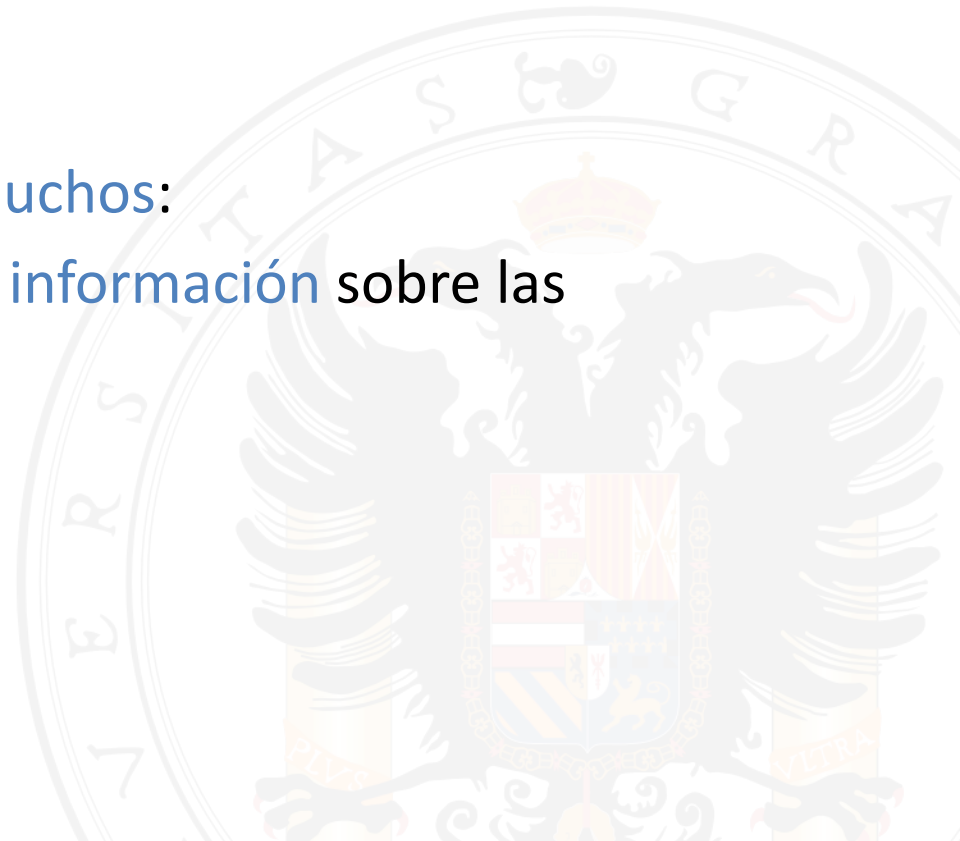
Esquemas



Instancias

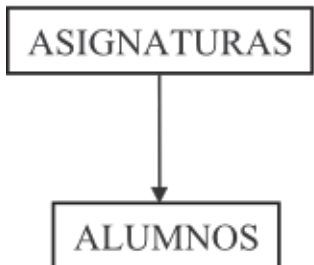
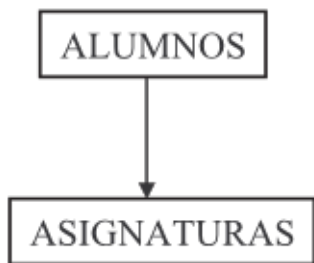


- Esta estructura plasma de forma muy directa:
 - Relaciones muchos a uno.
 - Relaciones uno a uno.
- Para relaciones muchos a muchos:
 - Hay que duplicar toda la información sobre las entidades involucradas.

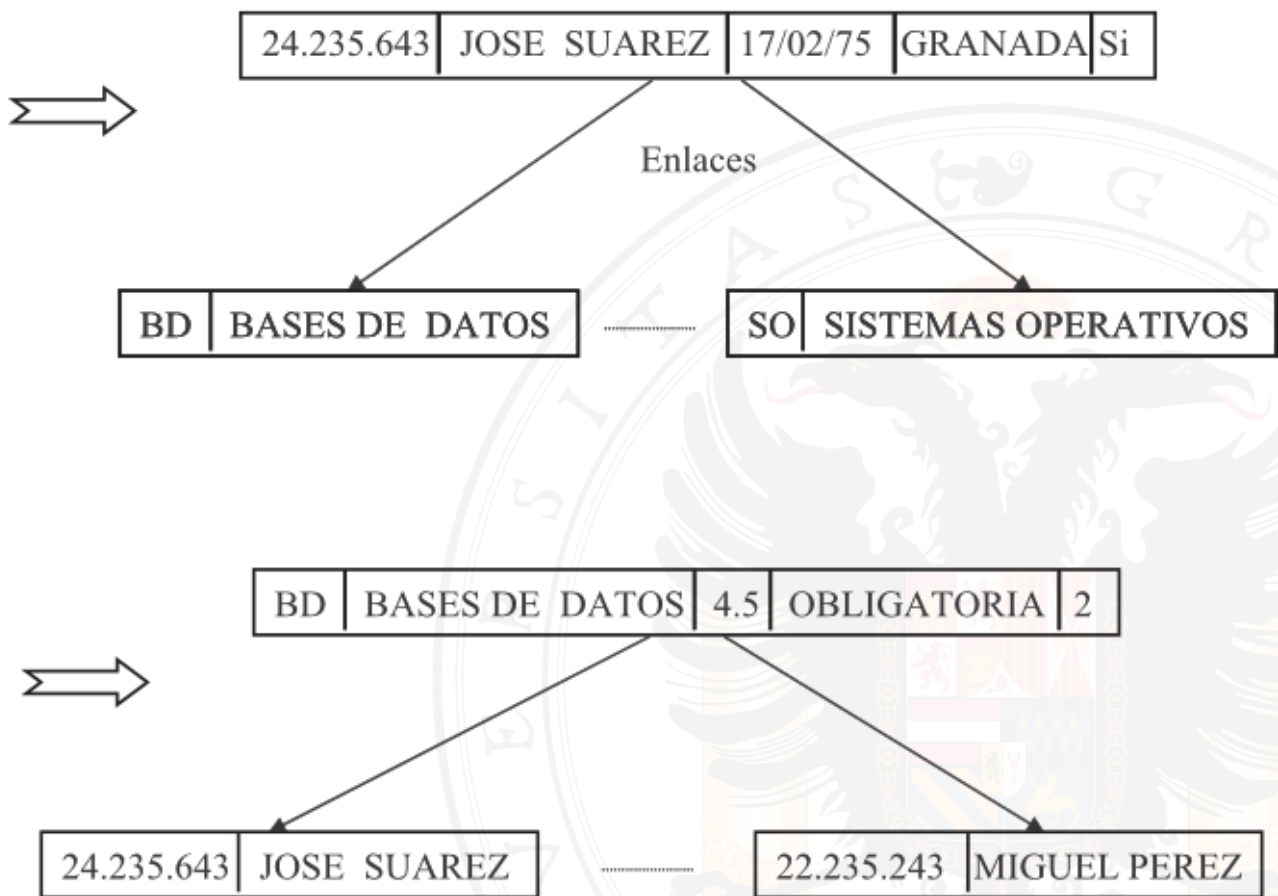


Ejemplo

Esquemas



Instancias



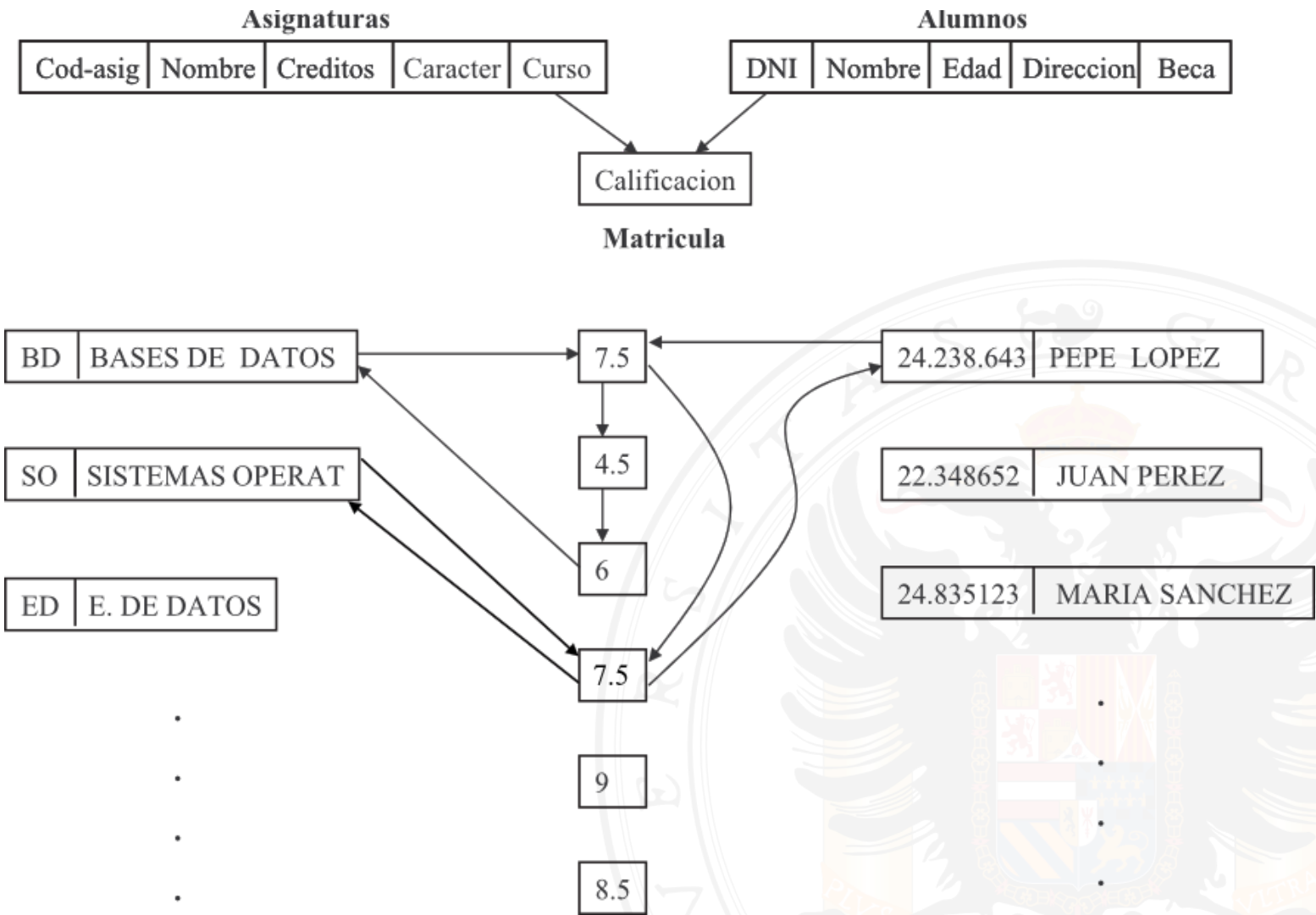
- **Inconvenientes:**
 - Almacenar árboles en ficheros es **complejo**:
 - Varios tipos de registros en el mismo fichero.
 - Punteros que hay que mantener.
 - DML **difícil** de:
 - Implementar.
 - Usar.
 - **Dependencia existencial** obligatoria de los registros de tipo secundario con respecto a los de tipo raíz:
 - No se podrá insertar un registro de tipo secundario mientras no exista uno de tipo raíz con el que enlazar.
 - **Redundancia necesaria** para plasmar relaciones muchos a muchos:
 - La integridad de los datos es costosa de mantener en actualizaciones y borrados.

- 1. Definición de modelos de datos**
- 2. El modelo de datos relacional. Estructura de datos. Integridad**
 - 2.1. La estructura de datos relacional
 - Definiciones iniciales
 - Propiedades de la estructura de datos relacional
 - Notación
 - 2.2. Restricciones o reglas de integridad
- 3. Otros modelos de datos**
 - 3.1. Modelo jerárquico
 - 3.2. Modelo en red



- Estructura de datos:
 - **Grafos** cuya topología depende de las conexiones existentes entre las entidades:
 - **Nodos**: registros.
 - **Arcos**: enlaces entre registros (punteros).
 - **Relaciones** entre conjuntos de entidades:
 - » Conectores: registros especiales constituidos por atributos propios de la relación (si existen).
 - » Cada ocurrencia de un conector representa una asociación distinta.
 - Cualquier registro puede relacionarse con cualquier registro.
- La BD es una **colección de instancias de grafos**.
- La estructura es **muy genérica**:
 - Permite plasmar todo tipo de relaciones.
 - Implementa directamente las relaciones muchos a muchos.

Ejemplo



- **Ventajas:**
 - Estructura algo más homogénea.
 - Permite insertar nuevas entidades en un conjunto de forma independiente.
- **Problemas:**
 - La existencia de enlaces entre los registros hace que las operaciones del DDL y el DML sigan siendo complejas de implementar y utilizar.

Comparativa

- Con respecto a la representación

- Relacional

- Un sólo elemento para la representación (esencialidad).
 - Conexiones lógicas.
 - Representación relaciones n:m simétrica.
 - Identidad por valor.

- Con respecto a la consulta

- Relacional

- Consultas simétricas en jerarquías.
 - Obtención de la consulta como resultado global.



- Lenguajes declarativos.

- Basado en grafos

- Dos elementos para la representación.
 - Conexiones en el modelo físico subyacente.
 - Representación conexiones n:m imposible en modelos jerárquicos, difícil en modelos en red.
 - Identidad por posición.

- Basado en grafos

- Consultas no simétricas en jerarquías.
 - Mecanismo de navegación por punteros.



- Lenguajes procedimentales.



¿Alguna pregunta?