

# Unidad 2: Modelos de BBDD

BBDD01, Sesión 2:  
Modelo Entidad / Relación  
Diagramas E/R

Los Profesores  
De la asignatura

# INDICE

- Conceptos básicos
- Restricciones
- Claves
- Cuestiones de diseño
- Conjuntos de entidades débiles
- Diagrama Entidad-Relación
- Diseño de un esquema de bases de datos E-R.

Referencias: Silberschatz 4ª Ed. pp 19-52

# Conceptos básicos

- Una base de datos se modela con un conjunto de
  - Esquemas de **entidades** y
  - **Relaciones** entre entidades
- Una entidad  $\Rightarrow$  un objeto del mundo real que existe y se distingue de otros objetos. Ejemplo: empleado, compañía
- Entidad puede ser concreta (libro) o abstracta (vacación)
- Entidad tiene atributos  $\Rightarrow$  propiedades de esa entidad
  - Ejemplo: nombre del empleado y dirección
  - Algunos atributos permiten identificar a la entidad de forma unívoca
- Entidad **tipo** (o simplemente entidad)  $\Rightarrow$  conjunto de **ocurrencias** entidades del mismo tipo, que comparten las mismas propiedades o atributos y participan en las mismas relaciones
  - Ejemplo: conjunto de todas las personas, compañías.
  - No tienen por qué ser disjuntos. Ej: clientes y empleados pueden ser de la entidad personas.

# Conceptos básicos

Ej: parte de una BD de un banco

| Nombre     | DNI        | Calle    | Ciudad     |
|------------|------------|----------|------------|
| Santos     | 32.112.312 | Mayor    | Peguerinos |
| Gómez      | 01.928.374 | Carretas | Cerceda    |
| López      | 67.789.901 | Mayor    | Peguerinos |
| Sotoca     | 55.555.555 | Real     | Cádiz      |
| Pérez      | 24.466.880 | Carretas | Cerceda    |
| Valdivieso | 96.396.396 | Goya     | Vigo       |
| Fernández  | 33.557.799 | Jazmín   | León       |

*cliente*

| Numero | Valor |
|--------|-------|
| P-17   | 1.000 |
| P-23   | 2.000 |
| P-15   | 1.500 |
| P-14   | 1.500 |
| P-19   | 500   |
| P-11   | 900   |
| P-16   | 1.300 |

*préstamo*

# Conceptos básicos

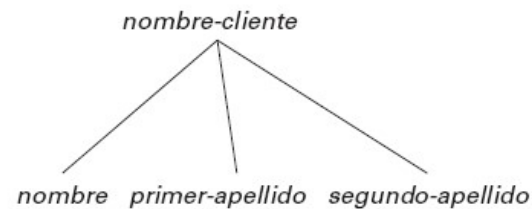
- Una entidad se representa por un conjunto de atributos (relevantes para el problema a resolver)  $\Rightarrow$  propiedades que posee cada miembro de un tipo de entidad
  - Ej: cliente (nombre, DNI, calle, ciudad)  
prestarlo (numero, valor)
- Cada ocurrencia entidad tiene un valor concreto para cada uno de sus atributos
- Dominio  $\Rightarrow$  conjunto de valores permitidos para cada atributo
  - Ejemplo: mes 1..12
- Cada ocurrencia entidad se puede describir como un conjunto de pares (atributo, valor).
  - Ej: instancia cliente como { (nombre, López), (DNI, 67.789.901), (calle, Mayor), (ciudad, Peguerinos) }
- Atributo  $\Rightarrow$  una función que asigna al conjunto de entidades un dominio
- Una base de datos incluye una colección de entidades tipo, cada una de las cuales incluye un número de entidades instancia, de ese tipo.

# Conceptos básicos

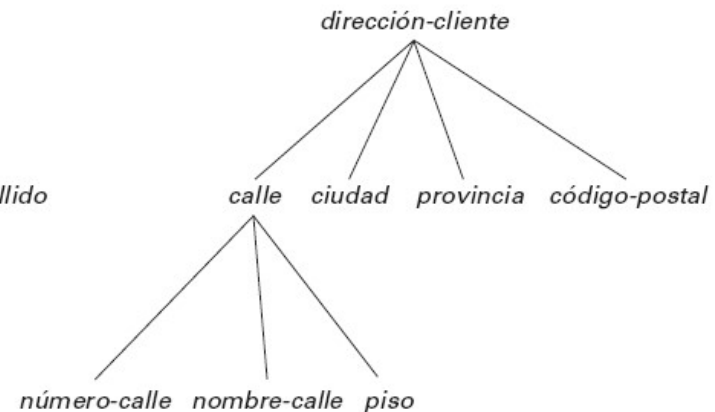
## ○ Tipos de atributos:

- Simples y compuestos
  - Compuestos permiten agrupar los atributos  $\Rightarrow$  modelos más claros

Atributos  
compuestos



Atributos  
componentes



- Monovalorados y multivalorados, ejemplo: Sueldo y número\_telefono
- Derivados:
  - Se pueden obtener de otros atributos o entidades relacionados
  - Ejemplo: edad se puede obtener de la fecha de nacimiento
  - No se almacena, se calcula cuando es necesario.

# Conceptos básicos

## ○ Valor nulo

- Un atributo toma el valor nulo cuando una instancia entidad no tiene valor para ese atributo
- Puede indicar no aplicable  $\Rightarrow$  valor no existe para esa entidad  
Ej: Número\_Embarazos para instancia de paciente con sexo=varón
- Puede designar que el valor es desconocido
  - Puede ser perdido  $\Rightarrow$  valor existe pero no se dispone de la información
    - Ejemplo: nombre del paciente que ha ingresado en coma
  - Desconocido  $\Rightarrow$  no se conoce si el valor existe realmente o no
    - Ejemplo: piso en una dirección del cliente.

# Conceptos básicos

- Una relación es una asociación entre varias entidades

- Ejemplo:

|                 |                           |                  |
|-----------------|---------------------------|------------------|
| <u>López</u>    | <u>Prestatario</u>        | <u>P-15</u>      |
| Entidad cliente | relación cliente-préstamo | entidad préstamo |

- Relación ocurrencia es la asociación entre  $n \geq 2$  entidades ocurrencia, tomadas de sus respectivas entidades tipo

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

donde  $(e_1, e_2, \dots, e_n)$  es una relación ocurrencia

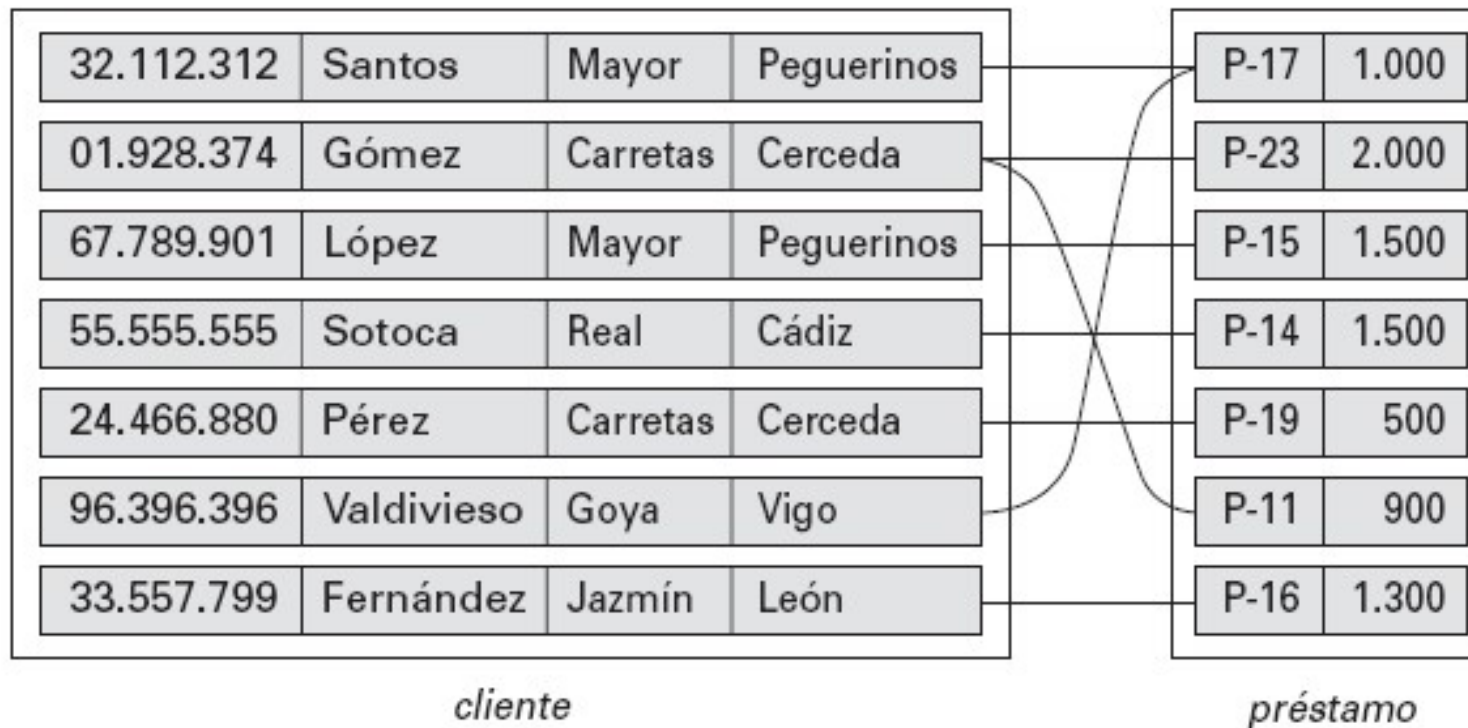
- Ejemplo: (López, P-15) es una ocurrencia de la relación Prestatario

- La relación tipo es el conjunto de ocurrencias de relaciones del mismo tipo

- Ejemplo: prestatario



## Conceptos básicos

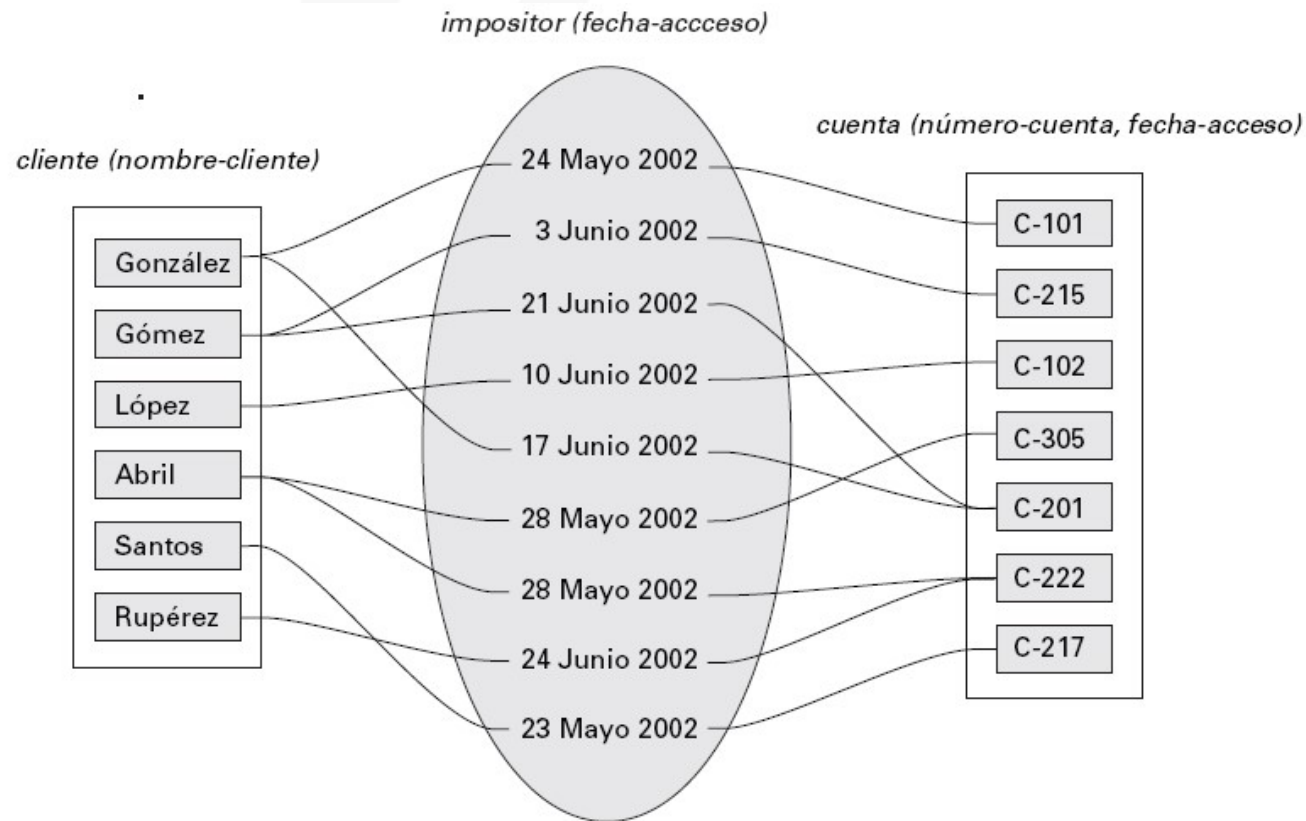


## Conceptos básicos

- La asociación entre entidades se conoce como participación  $\Rightarrow$  entidades  $E_1, E_2, \dots, E_n$  **participan** en la relación
  - El hecho de que el cliente López posee el préstamo P-15 es un ejemplar (ocurrencia) de la relación prestatario
- La función que desempeña una entidad en una relación se llama **papel (role)** de la entidad
- El papel no se suele especificar si es entre entidades tipo diferentes
- Sin embargo, son útiles cuando el significado de una relación necesita aclaración  $\Rightarrow$  entre instancias del mismo tipo de entidad (relación tipo recursivo)
  - Ejemplo entidad empleado y relación trabaja\_para: distinguir entre (empleado, jefe)

# Conceptos básicos

- Las relaciones pueden tener atributos descriptivos



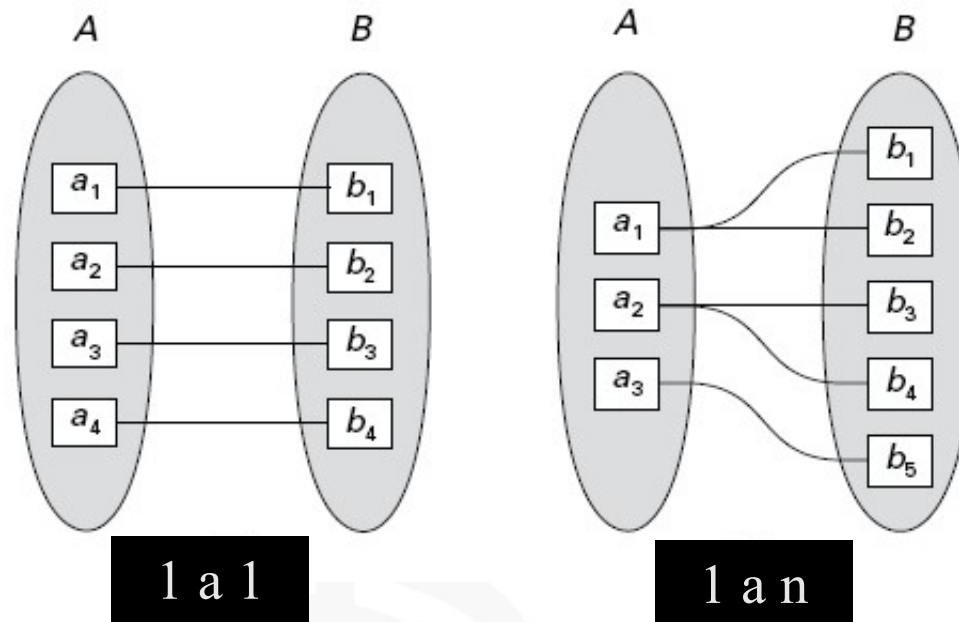
## Conceptos básicos

- Grado del esquema de la relación  $\Rightarrow$  número de entidades que participan
- Relaciones tipo que involucran dos entidades son binarias (grado 2)
- Puede haber más de dos conjuntos de entidades en una R:
  - Por ejemplo: conjunto de entidades empleado, sucursal y trabajo, donde (Santos, Navacerrada y director) es una ocurrencia
- La mayoría las de relaciones en sistemas de bases de datos son binarias
- Evitar relaciones ternarias si es posible.

# Restricciones

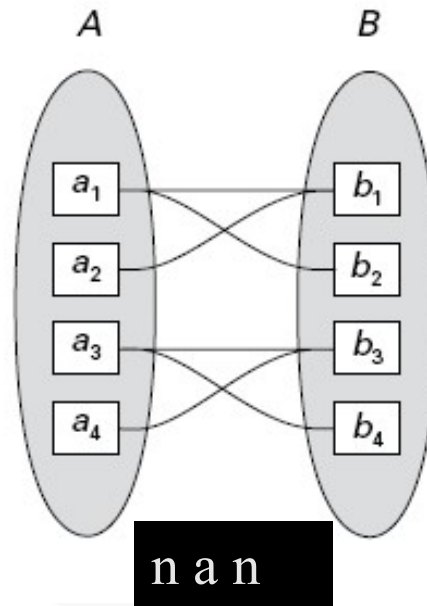
- Definir ciertas restricciones en el modelo E-R a las que el contenido de la BD se tiene que adaptar
  - Correspondencias de cardinalidades
  - Restricciones de participación
- Correspondencia de cardinalidad o Razón de cardinalidad  $\Rightarrow$  expresa número de entidades a las que la otra entidad puede estar asociada por medio de la relación. Posibles
  - Uno a uno  $\Rightarrow$  Una entidad A se asocia, a lo sumo, con una entidad B
  - Uno a varios  $\Rightarrow$  Una entidad A con varias entidades B, una B con una A
  - Varios a varios  $\Rightarrow$  Una entidad A con varias de B y viceversa
- Importante elegir la cardinalidad apropiada para modelar correctamente el mundo real
  - Ejemplo: cliente-préstamo

## Restricciones



- Nota: algunos elementos de  $A$  y  $B$  puede que no estén mapeados a ningún elemento del otro conjunto.

## Restricciones



- Nota: algunos elementos de A y B puede que no estén mapeados a ningún elemento del otro conjunto.

# Restricciones

## ○ Restricción de participación $\Rightarrow$ cómo es en la relación

- **Total**  $\Rightarrow$  cada entidad de un conjunto de entidades E participa, al menos, en una relación de R
- **Parcial**  $\Rightarrow$  sólo alguna entidad de E participa en la relación R
- Ejemplo: R=Cliente-Préstamo: Préstamo es total, si cada préstamo tiene asociado un cliente, y Cliente es parcial ya que un cliente del banco puede no tener un préstamo solicitado. Cliente-Cuenta es una relación total, todo cliente (del banco) tiene por lo menos una cuenta en el banco. Toda cuenta es de un cliente.



# Claves

- ¿Cómo distinguir una entidad ocurrencia dentro de una entidad tipo?
- ¿Cómo distinguir una relación ocurrencia dentro de una relación tipo?
- Se hace mediante el concepto de clave:
  - Superclave  $\Rightarrow$  conjunto de uno o más atributos que, tomados colectivamente, permiten identificar de forma única una ocurrencia de una entidad dentro del conjunto de ocurrencias
    - Ejemplo: DNI del conjunto cliente es superclave
    - Nombre\_cliente no es, porque varios clientes pueden tener el mismo nombre.

# Claves

- La superclave no es mínima  $\Rightarrow$  puede contener atributos innecesarios
- Cualquier superconjunto de una superclave K es superclave también
  - Ejemplo: DNI,nombre\_cliente es superclave
- Interesan las superclaves tales que los subconjuntos propios de ellas no son superclave. Tales superclaves mínimas se llaman **claves candidatas**
  - Por ejemplo: Si en la porción del mundo real que se esta modelando no pueden vivir dos personas con el mismo nombre en la misma casa  $\Rightarrow$  {nombre\_cliente,calle\_cliente } es una clave candidata. (DNI también)
  - {nombre\_cliente,DNI} no lo es, ya que DNI por sí sólo es superclave.

# Claves

- **Clave primaria** denota una clave candidata que es elegida por el diseñador de la base de datos como elemento principal, para identificar las entidades ocurrencia, dentro de una entidad
- Las claves son propiedades del esquema de la entidad
- En una entidad tipo no hay dos entidades ocurrencia iguales. No es una restricción del modelo ER, sino del mundo real
- Las claves candidatas hay que designarlas con cuidado
- La clave primaria hay que elegirla (de entre las claves candidatas) de manera que el valor de sus atributos no cambie. Ej: DNI.

# Claves

- Esquema de una relación
- Sea  $R$  una relación tipo que involucra las entidades tipo  $E_1, E_2, \dots, E_n$
- Sea  $k(E_i)$  el conjunto de atributos que forma la clave primaria de la entidad  $E_i$
- La clave primaria de  $R$  depende de la estructura de los atributos asociados a  $R$
- Si  $R$  no tiene atributos asociados:
  - Esquema  $R = k(E_1) \cup k(E_2) \cup \dots \cup k(E_n)$
- Si  $R$  tiene atributos  $a_1, a_2, \dots, a_m$  asociados,
  - $k(E_1) \cup k(E_2) \cup \dots \cup k(E_n) \cup \{a_1, a_2, \dots, a_m\}$
- En ambos casos
  - $k(E_1) \cup k(E_2) \cup \dots \cup k(E_n)$  es una superclave para  $R$

# Claves

- La estructura de la clave primaria para el conjunto de relaciones depende de la correspondencia de cardinalidades asociada al conjunto de relaciones
- Ejemplo: cliente, cuenta y atributo fecha\_acceso
  - Uno a uno  $\Rightarrow$  ¿Clave?
  - Uno a varios  $\Rightarrow$  ¿Clave?
  - Varios a varios  $\Rightarrow$  ¿Clave?
- Con relaciones no binarias es más complicado elegir la clave primaria
  - Sin restricciones de cardinalidad  $\Rightarrow$  La superclave se elige como antes  $\Rightarrow$  única clave candidata  $\Rightarrow$  clave primaria
  - Con restricciones, más complicado.

# Cuestiones de diseño

- Uso entidad o atributo
- ¿qué constituye un atributo? ¿qué constituye una entidad?
- Depende exclusivamente del problema del mundo real que se esté modelando y de la semántica asociada con el atributo en cuestión
  - Ejemplo: entidad: empleado, atributos: nombre\_empleado y número\_telefono
    - número\_teléfono es atributo  $\Rightarrow$  sólo 1 teléfono por cliente
    - es entidad  $\Rightarrow$  si un cliente más de un teléfono o  
existen teléfonos no asociados a empleados, o  
hay que guardar características del teléfono
    - nombre\_empleado  $\Rightarrow$  más difícil argumentar que sea entidad
  - Ejemplo: proyecto
    - Si solo necesito especificar en qué proyecto trabaja cada empleado  $\Rightarrow$  es un atributo de empleado
    - Si necesito propiedades del proyecto  $\Rightarrow$  es una entidad y se establece una relación empleado-proyecto

## Cuestiones de diseño

- Error  $\Rightarrow$  usar la clave primaria de una entidad como atributo de otra entidad  $\Rightarrow$  usar relación
- Error  $\Rightarrow$  designar atributos de la clave primaria de las entidades relacionadas como atributos de la relación. Ya están implícitos en la relación
- Uso de entidad o de relación
- No siempre está claro si es mejor expresar un objeto mediante entidades o mediante relaciones
- Por ejemplo: préstamo ¿como una entidad o como una relación entre clientes y sucursales, con número\_préstamo e importe como atributos?.

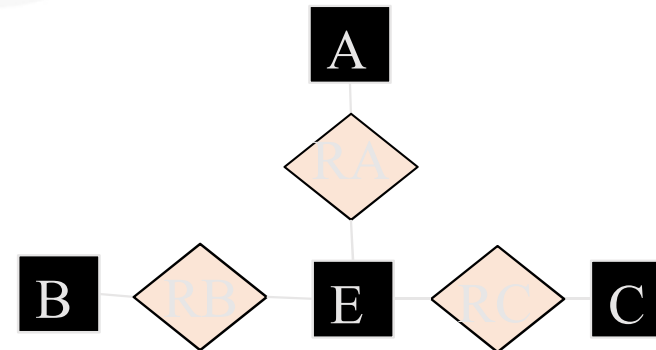
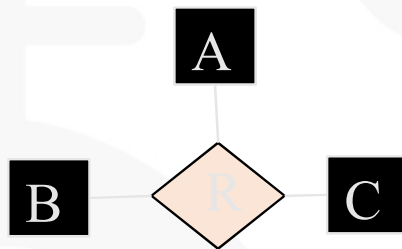
# Cuestiones de diseño

- Cada préstamo asocia un cliente y una sucursal  $\Rightarrow$  razonable una relación
- ¿Si varios clientes comparten el mismo préstamo?
  - Entonces habría que replicar el valor para los atributos descriptivos número-préstamo e importe en cada una de estas relaciones
  - Cada una de estas relaciones debe, por supuesto, tener el mismo valor número-préstamo e importe
  - Problemas:
    - Los datos se almacenan varias veces (desperdicio espacio y tiempo)
    - Podría producir **inconsistencia** de datos
- Teoría de normalización para evitar esta réplica de datos
- Si préstamo es entidad no hay réplica (**redundancia**)
- Una posible guía para determinar si diseñar una entidad o un relación es elegir relación para describir una acción que ocurre entre entidades
- Este enfoque puede también ser útil para decidir si ciertos atributos se pueden expresar más apropiadamente como relaciones



## Cuestiones de diseño

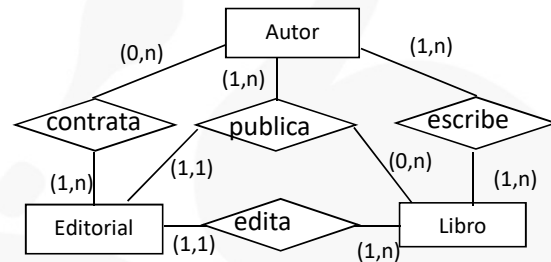
- Conjuntos de relaciones binarias frente a n-arias
- Siempre que se pueda, evitad ternarias
  - Ejemplo: ternaria padres (madre, padre, hijo)  
Dos binarias (padre, hijo), (madre, hijo)
- Sea R abstracto, ternario ( $n = 3$ ), y las entidades A, B, y C. Se sustituye la relación R por una entidad E y tres relaciones:
  - RA, relacionando E y A
  - RB, relacionando E y B
  - RC, relacionando E y C



## Cuestiones de diseño

- Si R tiene atributos, se asignan a E; por otra parte se crea un atributo de identificación especial para E
- Para cada relación  $(a_i, b_i, c_i)$  de R, se crea una nueva entidad  $e_i$  en E
- Se inserta un nuevo miembro:
  - $(e_i, a_i)$  en RA
  - $(e_i, b_i)$  en RB
  - $(e_i, c_i)$  en RC
- Se puede generalizar el proceso para relaciones n-arias
- No siempre es deseable:
  - Incrementa la complejidad del diseño.
  - Queda más clara la participación de varias entidades en una relación
  - Podría no haber forma de traducir restricciones ternarias en restricciones de las relaciones binarias.

# Descomposición de relaciones ternarias



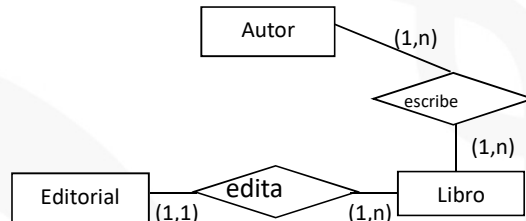
Editorial publica libros de autores

1) ¿Cardinalidades?

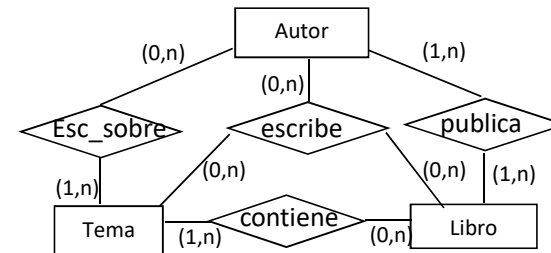
- Un libro se publica en una editorial por uno o varios autores.
- Un autor, en una editorial publica o no libros
- Un libro de un autor es publicado en una única editorial

3) ¿Detectando relaciones binarias?

4) ¿Alguna de las tres es redundante?



Una editorial edita libros escritos por autores



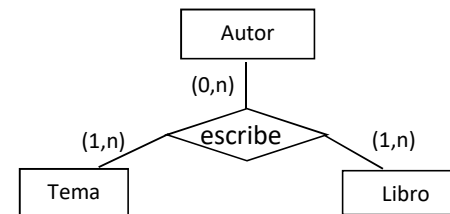
¿Sobre qué tema ha escrito un determinado autor en un determinado libro?

2) ¿Cardinalidades?

- ¿Sobre cuantos temas ha escrito un autor en un libro? (0,n)
- ¿En cuantos libros puede escribir un autor sobre un tema? (0,n)
- ¿Cuántos autores han escrito sobre un tema en un libro? (0,n)

5) ¿Detectando relaciones binarias?

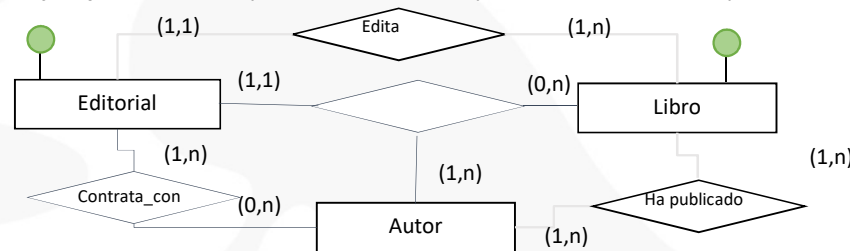
6) ¿Tiene la misma semántica?



Autores escriben sobre temas en libros

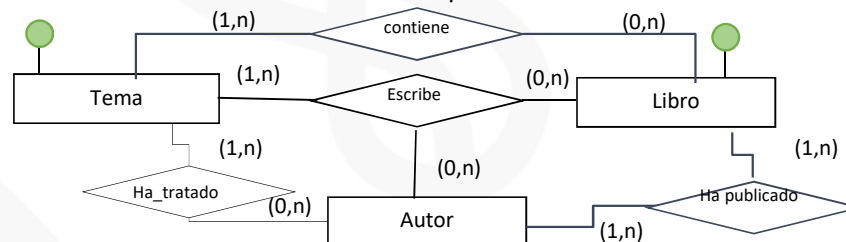
# Descomposición de relaciones ternarias

**Ejemplo1:** Un libro puede estar escrito por varios autores, se publica en una editorial



Representando el resto de relaciones conocidas, apreciamos que, sabiendo que libros ha publicado la editorial, y de que autores es cada libro sabemos si la editorial ha contratado o no a un autor concreto, por tanto, la relación ternaria, y Contrata\_con son redundantes y deben ser eliminadas

**Ejemplo2:** Un autor puede escribir sobre diversos temas, un libro puede contener diversos temas. Un libro es de varios autores. Interesa saber sobre que tema ha escrito un determinado autor en un determinado libro



A diferencia del ejemplo anterior, el saber que un autor ha tratado un determinado tema, y que un libro contiene determinado tema, y que un autor ha publicado un libro, “pierdo” (pudiéramos decir que “gano”) información de si ese autor ha tratado el tema en ese libro en concreto

## Conjuntos de entidades débiles

- Esquema entidad fuerte  $\Rightarrow$  conjunto de entidades que tiene una clave primaria
  - Ejemplo: Préstamo (id\_préstamo, cliente, importe, fecha)
- Un conjunto de entidades puede no tener suficientes atributos para formar una clave primaria  $\Rightarrow$  esquema entidad débil
  - Ejemplo: pago(número\_pago, fecha\_pago, importe\_pago)
- Para que entidad débil tenga sentido  $\Rightarrow$  asociada a entidad fuerte  $\Rightarrow$  entidad identificadora o **propietaria**
- La ocurrencia de una entidad débil **depende existencialmente** de la entidad identificadora
- Relación identificadora:
  - Participación total del conjunto de entidades débiles.

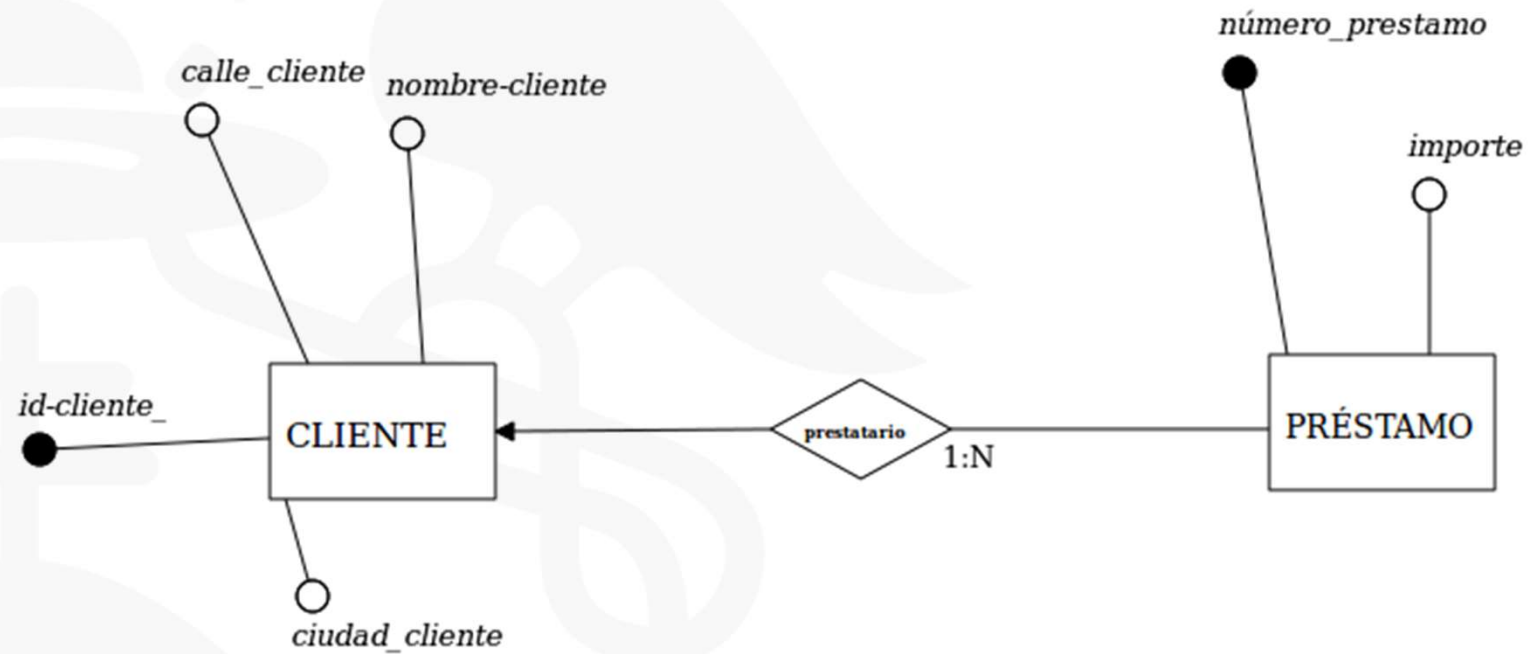
## Conjuntos de entidades débiles

- Se necesita asociar todas las ocurrencias entidades del conjunto de entidades débiles que dependen de una ocurrencia de entidad fuerte particular
- El **discriminante** de un conjunto de entidades débiles es un conjunto de atributos que realiza esta asociación
  - Ejemplo: número\_pago en pago
- El discriminante de un conjunto de entidades débiles  $\Rightarrow$  la clave parcial
- La clave primaria  $\Rightarrow$  clave primaria de la entidad identificadora + discriminante
  - Ejemplo: {id\_préstamo, numero\_pago}

# Diagrama Entidad-Relación

- La estructura lógica general de una base de datos se puede expresar gráficamente mediante un **diagrama E-R**.
- Son simples y claros
- Componentes. Resumen:
  - **Rectángulo**  $\Rightarrow$  entidad tipo
  - **Rombo**  $\Rightarrow$  relación tipo
  - **Conector**  $\Rightarrow$  une entidades a relación
  - **Rectángulo doble**  $\Rightarrow$  entidad débil
  - **Línea continua con círculo**  $\Rightarrow$  atributo obligatorio
  - **Línea discontinua con círculo**  $\Rightarrow$  atributo opcional
  - **Flecha continua con círculo**  $\Rightarrow$  atributos multivalorados
  - **Flecha discontinua con círculo**  $\Rightarrow$  atributo multivalorado opcional

## Diagrama Entidad-Relación



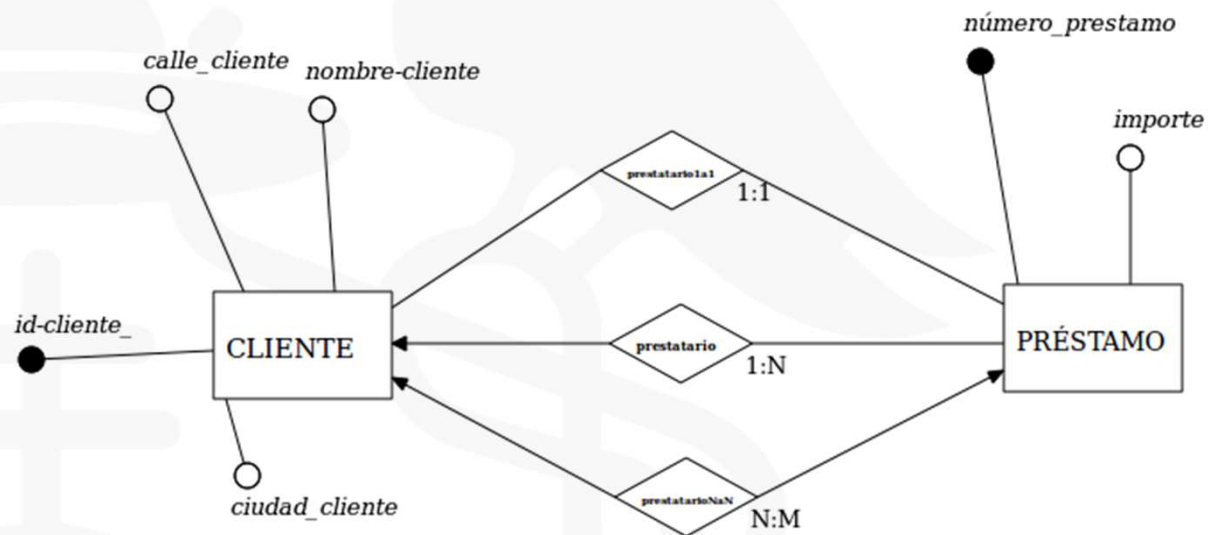


## Diagrama Entidad-Relación

- La política de diferentes bancos podría establecer la relación “prestatario” como varios a varios, uno a varios, varios a uno o uno a uno
- Se dibuja o una línea dirigida ( $\rightarrow$ ) o una línea no dirigida ( $—$ ) entre el conjunto de relaciones y el conjunto de entidades en cuestión, en función de la cardinalidad.

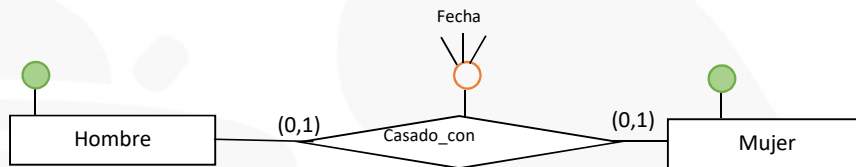
# Diagrama Entidad-Relación

- Relaciones uno a uno, uno a varios y varios a varios

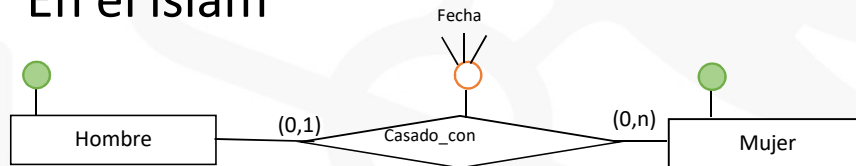


## MER: depende del universo del discurso

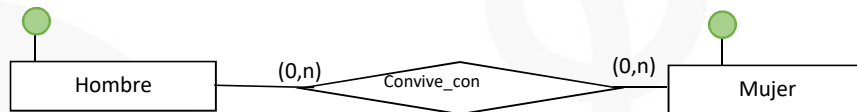
En el catolicismo



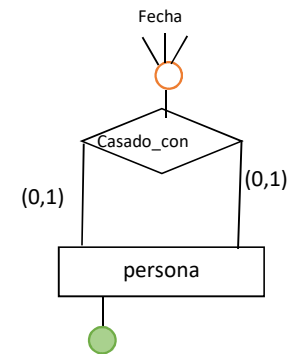
En el islam



En comuna de los años 60



Hoy en España



## Acuerdo de nomenclatura

- Al representar la cardinalidad de la relación mediante flechas, **distintas fuentes utilizan distinto criterio**,
- Adoptamos aquí la de Adoración de Miguel, que utilizan en la uc3m (al contrario que la usada en Silberschatz).

Un coche tiene (0,1) dueño

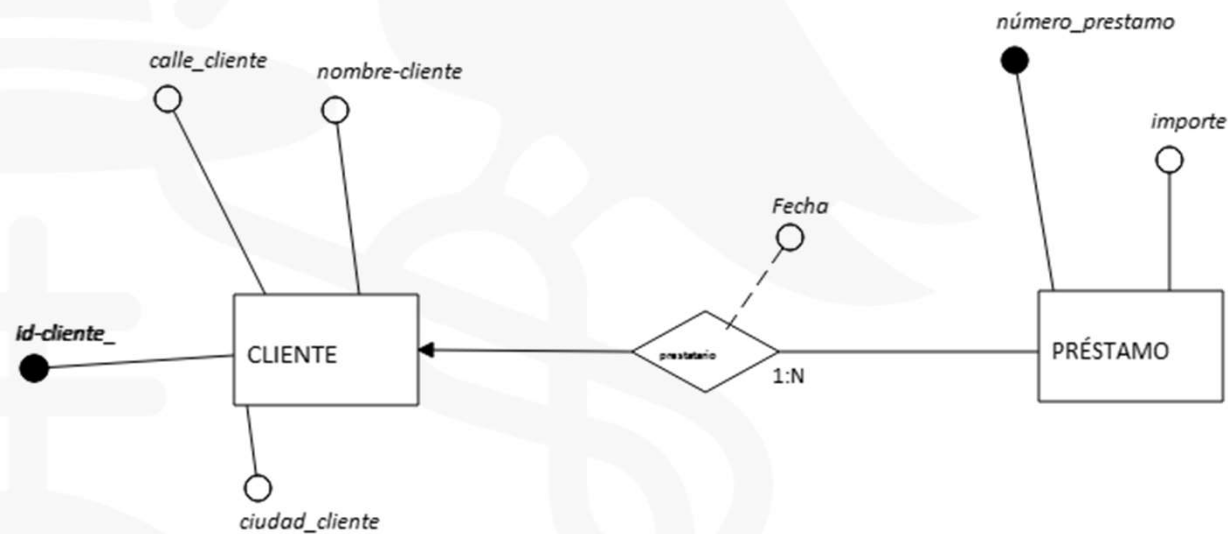
Una persona tiene (0,n) coches

(0,1)      1:N      (0,n)  
PERSONA-----<TIENE>----->COCHE

- La flecha apunta a la entidad de la que hay muchos.

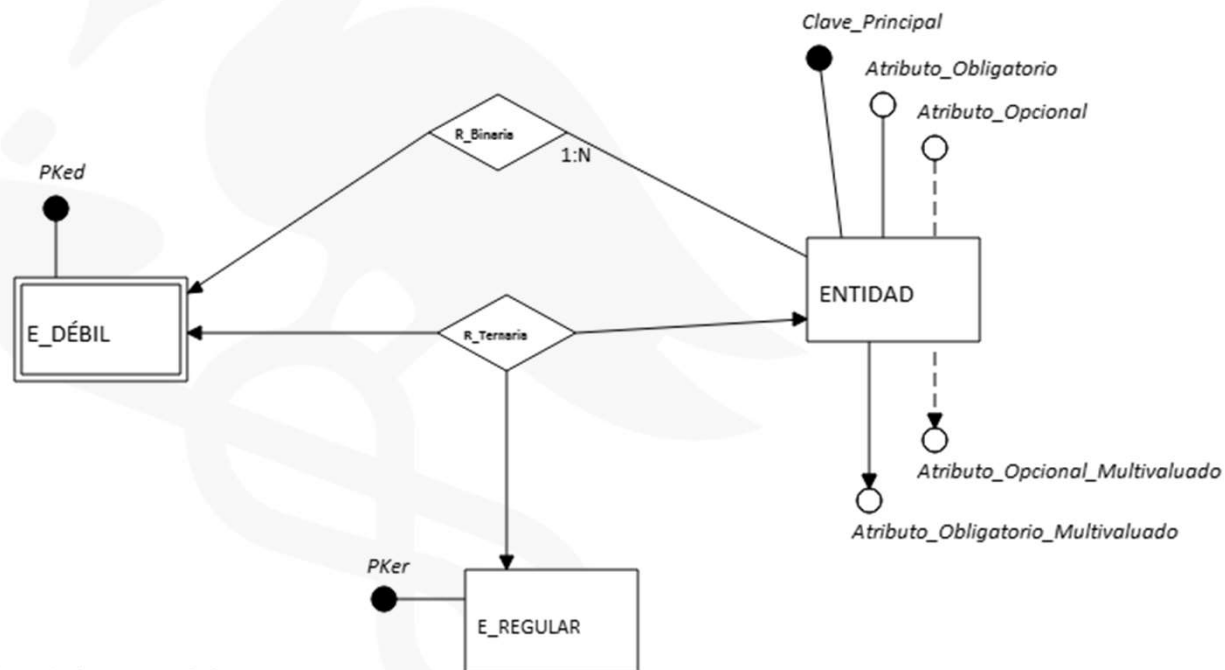
# Diagrama Entidad-Relación

- Si un conjunto de relaciones tiene atributos asociados  $\Rightarrow$  se unen al conjunto de relaciones



# Diagrama Entidad-Relación

- Representación de entidades débiles y relaciones n-arias

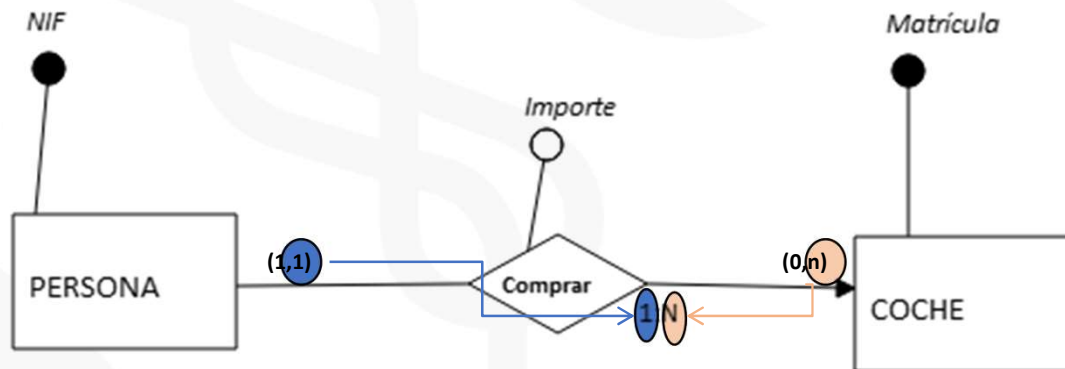


- Por cierto, ¿se podría simplificar?

# Diagrama Entidad-Relación

## ○ También se **debe** representar:

- Restricciones sobre el número de veces que la entidad participa en el conjunto de las relaciones
- Un segmento entre entidades y unas relaciones binarias puede tener una cardinalidad mínima y máxima (mín,máx), donde mín puede tomar valores 0 o 1, y máx puede tomar valores 1 o n
- Ejemplo:
  - Una persona puede tener como mínimo cero coches, como máximo n coches.
  - Un coche es adquirido como mínimo por una persona, y como máximo por una persona.



# Diseño del esquema de BD E-R

- Amplio rango de alternativas para diseñar una base de datos
- Fases de diseño
  - La fase inicial del diseño de la BD  $\Rightarrow$  es caracterizar completamente las necesidades de datos esperadas por los usuarios de la base de datos  $\Rightarrow$  especificación de requisitos del usuario
  - Elección del modelo de datos  $\Rightarrow$  esquema conceptual de la BD (diseño conceptual). Ejemplo. Modelo E-R
    - Hay que revisar el modelo y eliminar conflictos y redundancias
    - Especificación de los requisitos funcionales (operaciones)
  - Fase de diseño lógico  $\Rightarrow$  traducir el modelo conceptual al modelo de datos del SGBD
  - Fase de diseño físico  $\Rightarrow$  características físicas de la base de datos: archivos y estructuras de almacenamiento.
- Ejemplo: diseño de una base de datos para un banco.