

# Bases de Datos

## TEMA 4. ANEXO. UML

Introducción al **lenguaje de modelado unificado** (*Unified Modeling Language*) y su uso para el diseño conceptual de bases de datos

# Lenguaje de Modelado Unificado (UML)

2

- Lenguaje de modelado ampliamente utilizado para modelar y documentar sistemas de información
- Proporciona un vocabulario común (*símbolos*) y distintos tipos de *vistas, diagramas y reglas*, que permiten realizar todas las etapas de análisis y diseño de sistemas de información
- Modelado de datos
  - **Diagrama de clases:** Expresa las clases que modelan el sistema y las relaciones (asociaciones) entre ellas
    - Un *tipo de entidad* del MER se representa como una **clase**
    - Una instancia de clase es un **objeto**, y corresponde a una *entidad concreta (una instancia)* del MER

# Diagrama de clases UML

3

- Componentes:
  - ▣ **Clases**
    - **Atributos**
  - ▣ **Relaciones:**
    - **Asociaciones**
    - **Composiciones *Identifying***
    - **Generalizaciones**
    - ...
  - ▣ **Clases Asociativas**
  - ▣ ...

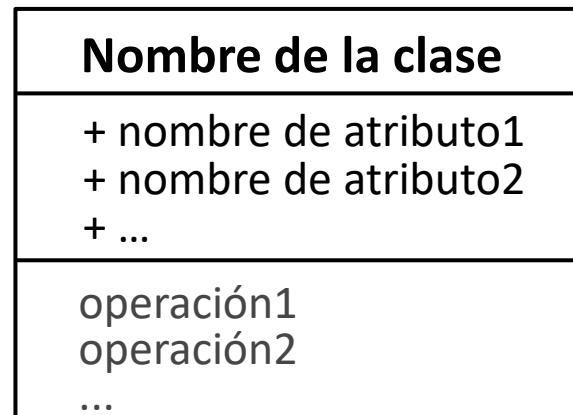
# Clase

4

- La clase equivale al *tipo de entidad* en el MER
- Se representa con un **rectángulo** con 3 secciones:
  - ▣ **Nombre** de la clase
    - Primera letra en mayúscula, el resto en minúscula
  - ▣ **Atributos** que tienen las instancias de la clase (objetos)
  - ▣ Operaciones que pueden realizar los objetos

(\*\* No lo usaremos\*\*)

Así que utilizaremos  
rectángulos con sólo  
2 secciones



# Clase

5

## □ Tipos de atributo. Notación:

- ▣ Por defecto, un atributo se considera **almacenado, monovalorado y obligatorio**, igual que en el MER
- ▣ Si es **opcional** (admite el NULL) se indica con la etiqueta [0..max], donde  $\max \geq 1$
- ▣ Si es **multivalorado** se indica con la etiqueta [min..max] , donde min es 0 ó 1 y  $\max > 1$  y puede ser n
- ▣ Si es **derivado** (calculado) se indica con la etiqueta [derived]
- ▣ Si es **clave principal** (o **discriminante** de una clase débil) se antepone PK
- ▣ Si es **clave alternativa** se antepone AK
- ▣ No se puede representar atributos **compuestos**
  - Modelaremos todos como simples

Scout
PK <b>codigo</b> AK <b>DNI</b> [0..1] + <b>nombre_completo</b> + <b>telefono</b> [1..3] + <b>fecha_nacim</b> + <b>edad</b> [derived]

# Asociación

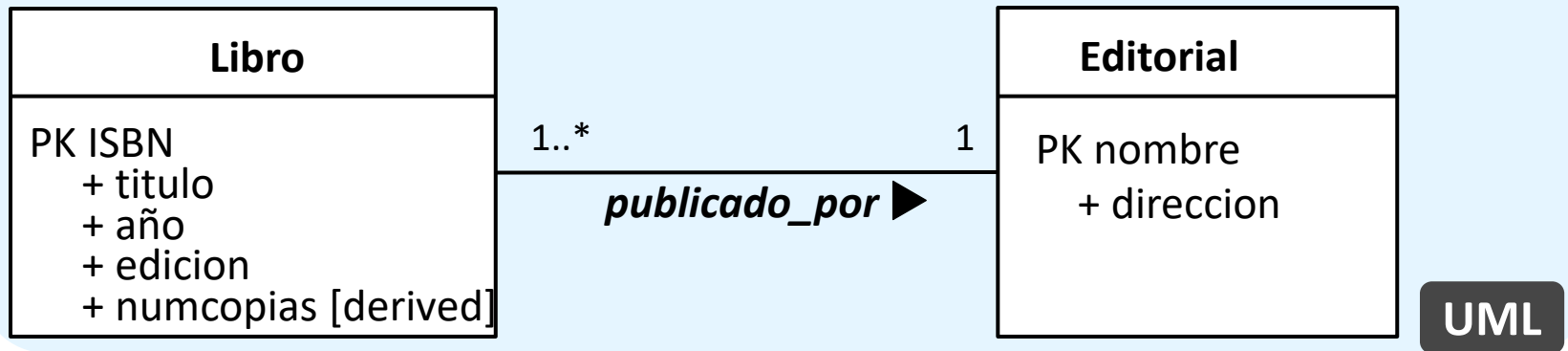
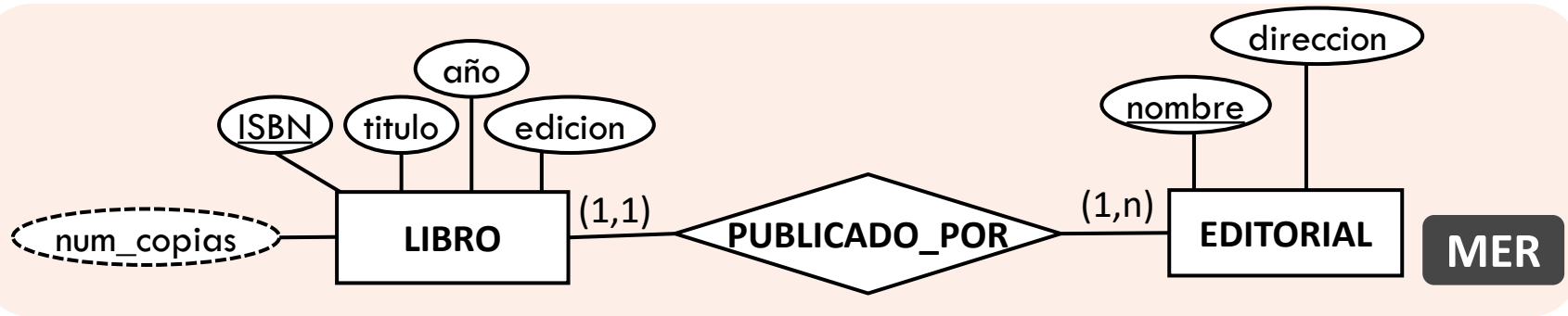
6

- Una asociación equivale a un *tipo de relación* MER
- Representada mediante líneas que unen dos clases
- **Multiplicidad** (equivale a la cardinalidad)
  - ▣ Notación *min..max*
  - ▣ Un \* indica que no hay un límite máximo en la participación



# Notación MER vs UML

7



- ❑ Importante: las cardinalidades se indican “al revés” que en el MER

# Equivalencia cardinalidad MER vs UML

8

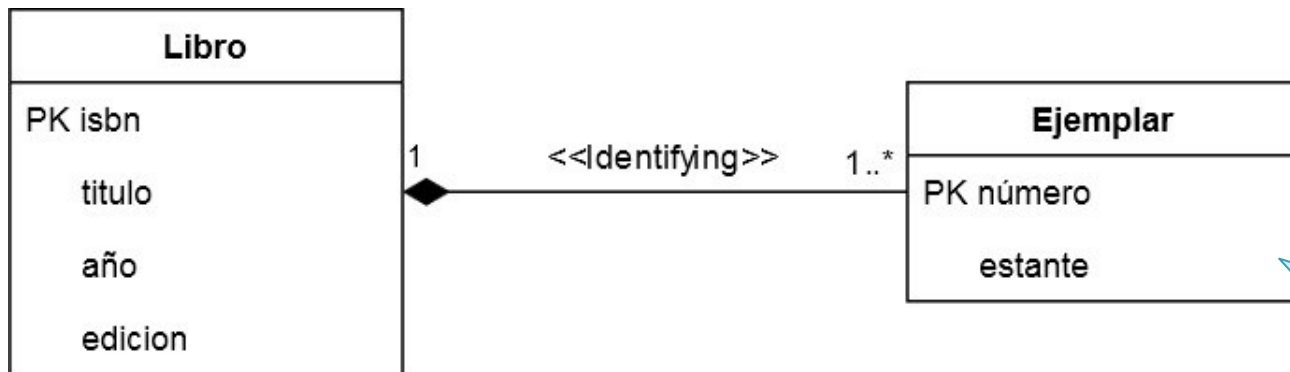
MER	UML
(1,1)	1
(0,1)	0..1
(0,n)	0..*
(1,n)	1..*



# Composición *identifying*

9

- ❑ Para representar la relación entre una “clase débil” y su “clase fuerte”
- ❑ Se denota con una **relación de composición**: rombo negro ◆ junto a la clase “fuerte”
  - ▣ No tiene sentido que exista la débil si no existe la fuerte
- ❑ La etiqueta «Identifying» indica que la clave de la clase débil se forma con la combinación de la PK de la fuerte junto con la PK de la débil (si tiene)



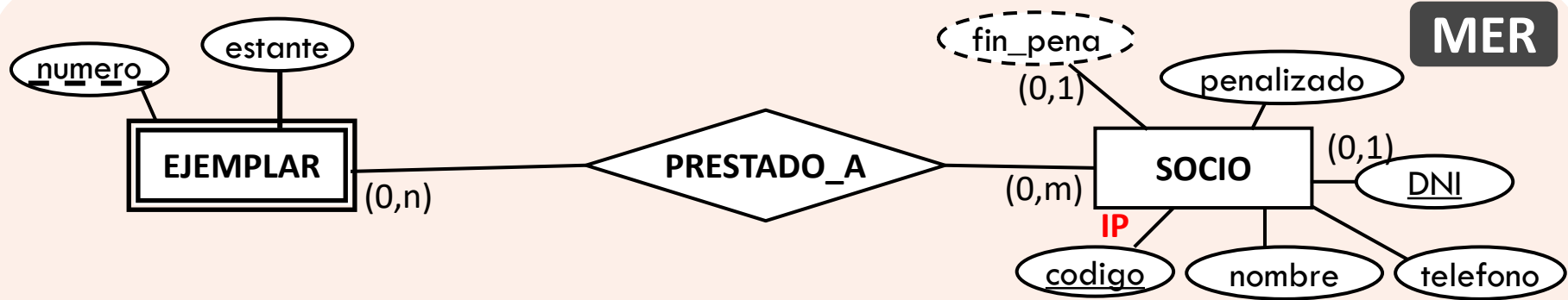
En UML **no** existe la noción de **atributo discriminante** de la entidad débil

Cada ejemplar se identifica mediante (isbn, número)

# Clase Asociativa

10

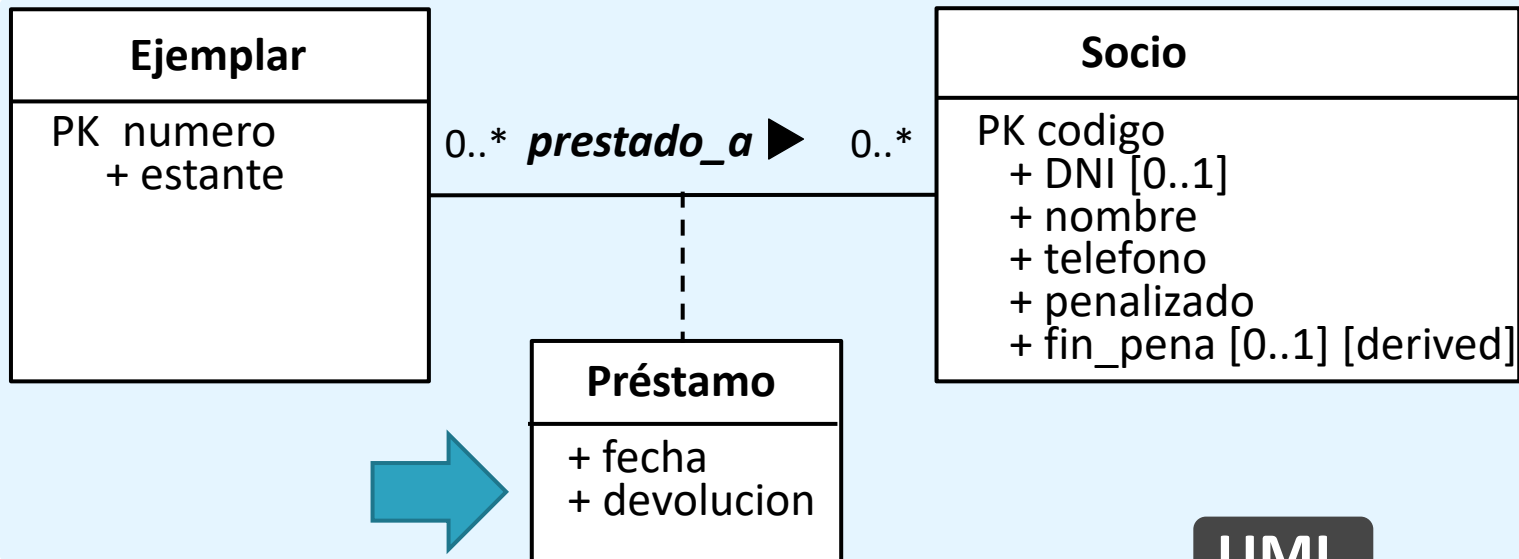
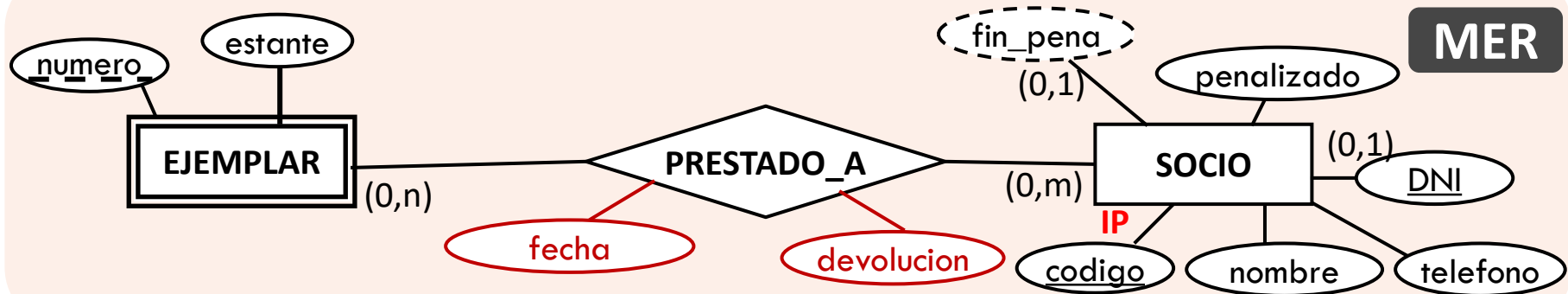
- Permite representar atributos de relación



- Un socio toma en préstamo de “cero a muchos” ejemplares
- Un ejemplar es prestado de “cero a muchas” veces
- Necesitamos *guardar la fecha de cada préstamo y de la devolución. ¿Dónde se guardan?*

# Clase Asociativa

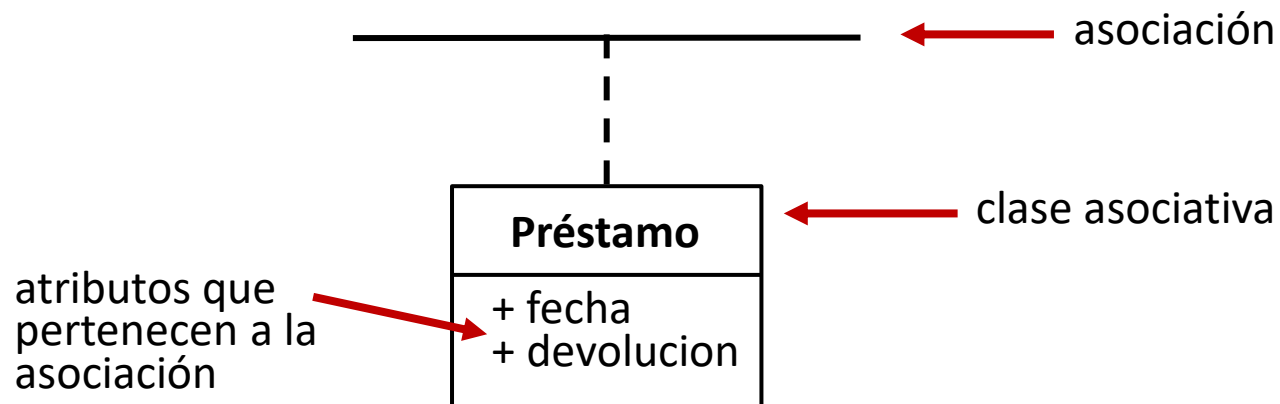
11



# Clase Asociativa

12

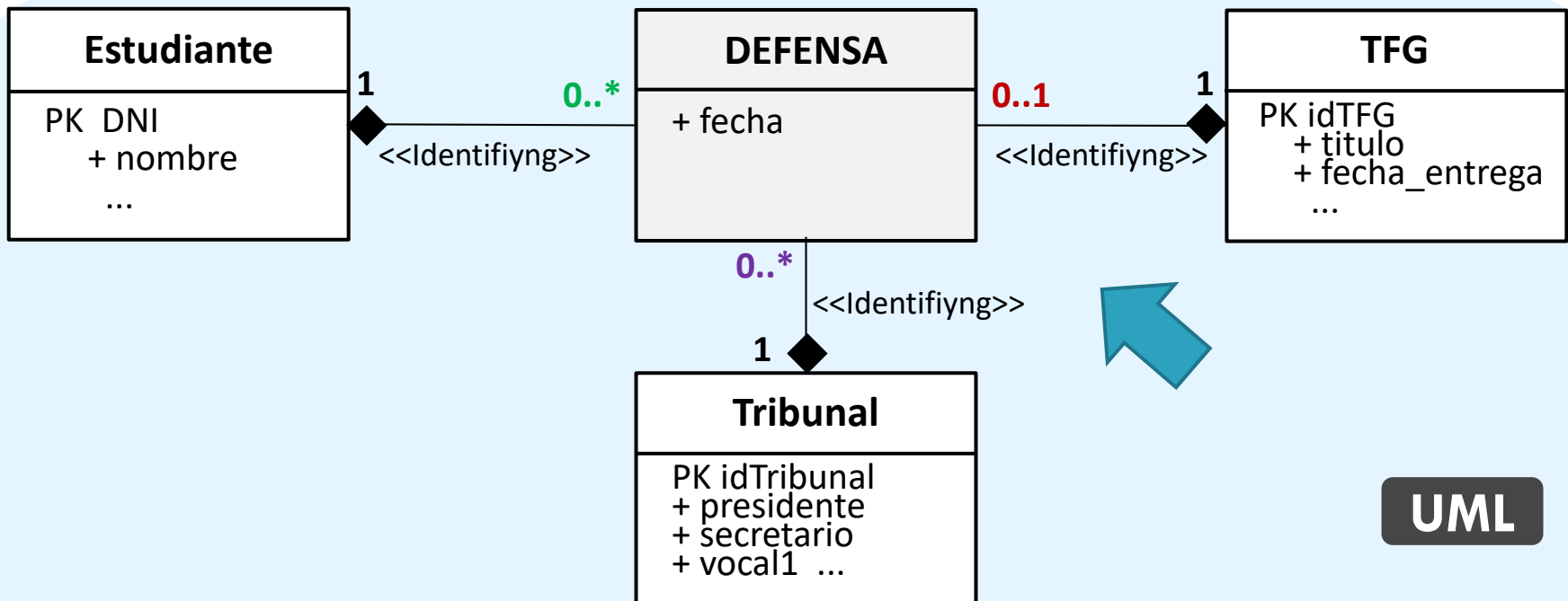
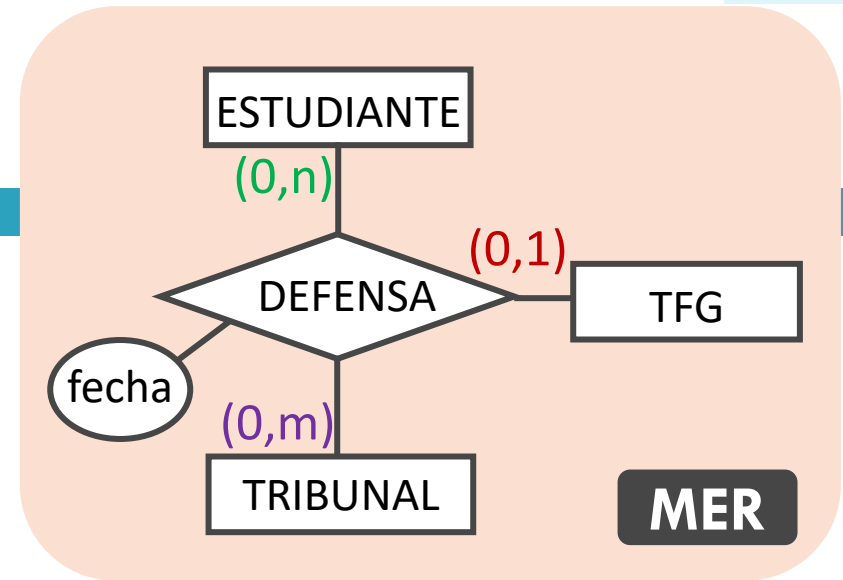
- Una clase asociativa, por tanto...
  - ▣ **Amplía la descripción de una asociación** entre dos clases, porque tal vínculo tiene asociados atributos.
  - ▣ **No está conectada**, por tanto, **con ninguna otra clase**.
  - ▣ En el diagrama de clases aparece **conectada a una asociación mediante una línea discontinua**.



# Relaciones n-arias


13

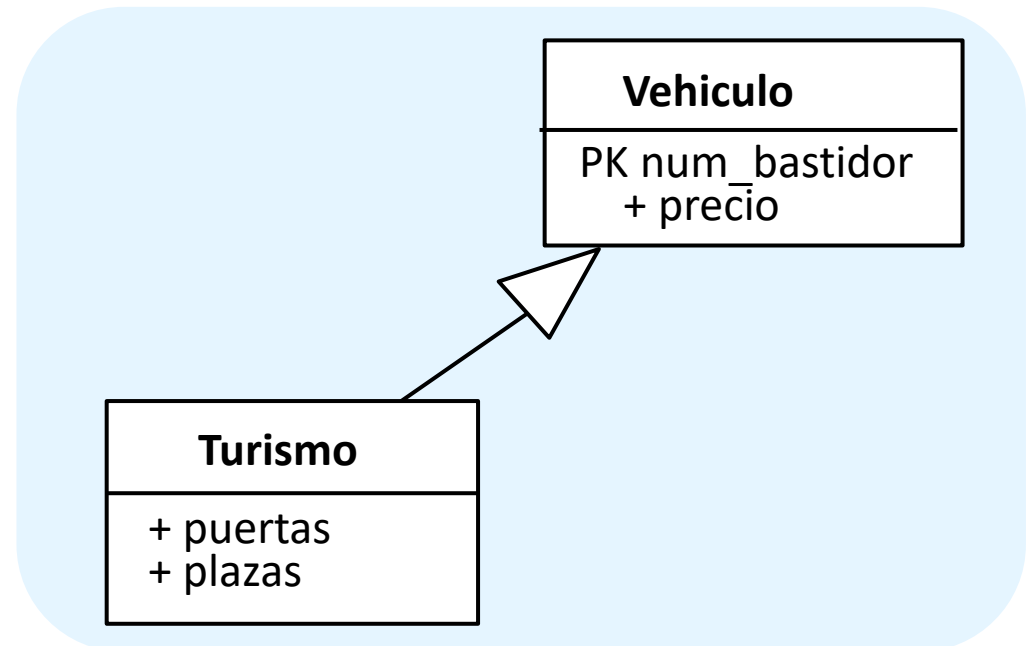
- Las relaciones n-arias en ER se representan en UML como una **clase** (que no es asociativa, ojo)



# Generalización (Jerarquías)

14

- Relación “...**es un**...” entre dos clases
  - ▣ Relación subclase (subtipo) / superclase (supertipo)
- **Herencia**: la subclase hereda atributos y asociaciones de su superclase
- Notación: flecha con la punta pegada a la superclase
-  Sin cardinalidad



# Generalización (Jerarquías)

15

