

# Diseño y Desarrollo de Sistemas de Información

---

## Tema 2

Modelado conceptual de una  
base de datos y transformación  
al modelo lógico

# Objetivos

- Tener la capacidad de sintetizar un análisis de requisitos mediante un modelo conceptual
- Conocer el modelo Entidad – Relación por ser el modelo de datos conceptual más utilizado
- Estudiar, en detalle, los conceptos del modelo básico y las extensiones añadidas
- Conocer los problemas de semántica que se pueden derivar del modelo y presentar las posibles soluciones
- Conocer el proceso de transformación del modelo conceptual (entidad-relación extendido) al modelo lógico estándar (modelo relacional) y específico (SGBD Oracle)

# Contenidos

## 2.1. Introducción

## 2.2. Elementos del modelo entidad-relación

### 2.2.1. Entidad

### 2.2.2. Relación: restricciones de cardinalidad

### 2.2.3. Dominio y atributo

## 2.3. Transformación al modelo relacional: generalidades

### 2.3.1. Transformación de entidades

### 2.3.2. Transformación de relaciones

- Binarias
- Reflexivas
- Ternarias

# Contenidos

## 2.4. Extensiones del modelo E-R

- 2.4.1. Generalización y especialización
- 2.4.2. Categoría
- 2.4.3. Agregación
- 2.4.4. Relaciones exclusivas

## 2.5 Transformación de las extensiones del modelo E-R

- 2.5.1. Generalización y especialización
- 2.5.2. Categoría
- 2.5.3. Agregación
- 2.5.4. Relaciones exclusivas

## 2.6. Consideraciones de diseño

- 2.6.1. Relaciones con grado mayor que dos
- 2.6.2. Control de redundancias
- 2.6.3. Dimensión temporal

## 2.1. Introducción

- La tercera fase del diseño de una base de datos consiste en realizar un **modelo conceptual** que recoja toda la semántica referente a la información que se ha obtenido en el análisis de requisitos y que deseamos integrar en la base de datos
- Esta etapa se denomina **diseño conceptual de la base de datos**
- Se plantean dos preguntas:

¿qué  
representar?

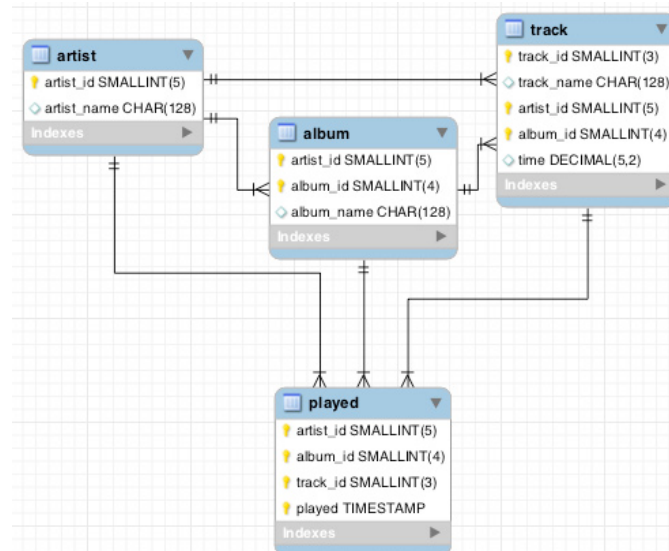
¿cómo  
representarlo?

- A la primera pregunta responde:
  - el estudio de las reglas de la empresa (que proveen el marco para el análisis del sistema). **Reglas de negocio**
  - entrevistas a los usuarios de los diferentes niveles de la organización (que proveen los detalles sobre los datos)
- En la realización del esquema conceptual de cualquier BD es fundamental el **conocimiento del problema a modelar**
- En esta fase se utiliza el lenguaje natural para recoger toda la información (por ejemplo mediante entrevistas), ya que los usuarios pueden expresar, de esa forma, todas sus necesidades

- A la segunda pregunta da respuesta el **modelo Entidad-Relación**:
  - representación normalizada que permite ver la información con todo su contenido semántico pero sin describir los aspectos ligados a la instrumentación del modelo en un SGBD, y que es fácil de traducir al esquema lógico
- La construcción de esquemas conceptuales es una labor creativa que se realiza en sucesivos pasos de refinamiento
  - no todos los diseñadores obtienen el mismo esquema E-R
  - Fue propuesto por **Peter Chen** entre los años 1976 y 1977
  - Según el autor: ***“El modelo E-R puede ser usado como una base para una vista unificada de los datos”***, adoptando “el enfoque más natural del mundo real que consiste en **entidades** y **relaciones**”

- 
- ```

    erDiagram
        SOCIO ||--o{ EJEMPLAR : préstamo
        préstamo {
            string f_entrega
            string f_devol
        }
    
```





## 2.2. Elementos del modelo entidad-relación

|         |          |          |         |
|---------|----------|----------|---------|
| Entidad | Relación | Atributo | Dominio |
|---------|----------|----------|---------|

### 2.2.1. Entidad

*"Objeto del mundo (real o abstracto) acerca del cual queremos almacenar información en la base de datos y que es distinguible de otros objetos"*

*"Algo con realidad objetiva que existe o que puede ser pensado" (Hall, 1976)*

*"Una persona, lugar, cosa, concepto o suceso, real o abstracto, de interés para la empresa" (ANSI, 1977)*

- Ejemplos:
  - La persona Anselmo Martín Sañudo con D.N.I. 29.123.321
  - El libro “Bases de Datos” con isbn 0-07-044754-3
  - El segundo curso del Grado en Ing. Informática
- Un **tipo de entidad** (Chen lo denominó ***entity set***) es un conjunto de entidades de un mismo tipo que comparten las mismas propiedades
  - Todas las personas matriculadas en la universidad pueden formar el tipo de entidad ALUMNO. El tipo de entidad PAS puede representar a los trabajadores de administración y servicios de la universidad
- En este curso denominaremos **entidad** al conjunto de entidades y **ocurrencia** o **ejemplar** a un determinado elemento de la entidad

- En general, existen 2 tipos de entidades:
  - **Regulares o fuertes.** Aquellas que tienen existencia por sí mismas (EMPLEADO, PROYECTO, etc.)
  - **Débiles.** Aquellas cuya existencia depende de la existencia de otra entidad (una entidad FAMILIAR dependerá de la entidad EMPLEADO, una entidad EXPEDIENTE dependerá de una entidad ALUMNO). La desaparición de una ocurrencia en la entidad ALUMNO implica la desaparición de las ocurrencias dependientes que se encuentran en la entidad débil EXPEDIENTE

- Representación (diagrama de Chen):



- Representación (UML):

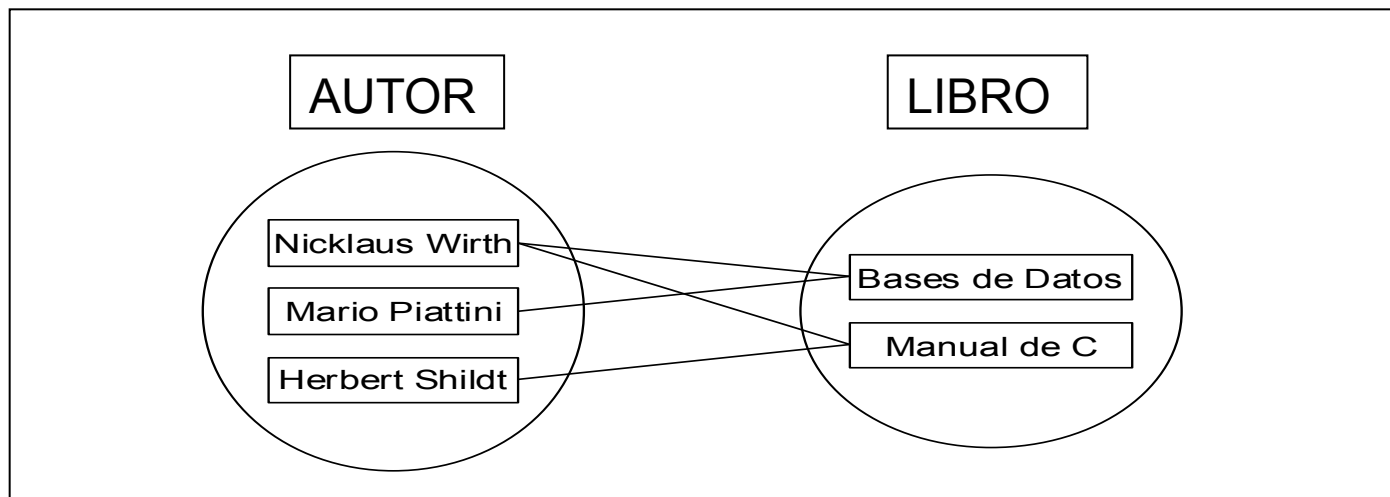


- Algunas veces resulta difícil decidir si un determinado objeto o concepto se modela como una entidad
  - **Ejemplo:** el color es habitualmente una propiedad de una entidad (el color de un coche), pero en una fábrica de pinturas probablemente sería apropiado modelar el color como una entidad con sus propias propiedades
- Algunos autores han intentado precisar el concepto de entidad
- TARDIEU et al. (1979) proponen tres reglas generales que debe cumplir una entidad:
  - tiene que tener **existencia propia**
  - cada ejemplar de un tipo de entidad debe poder **distinguirse** de las demás
  - todos los ejemplares de un tipo de entidad deben tener las **mismas propiedades**

## 2.2.2. Relación: restricciones de cardinalidad

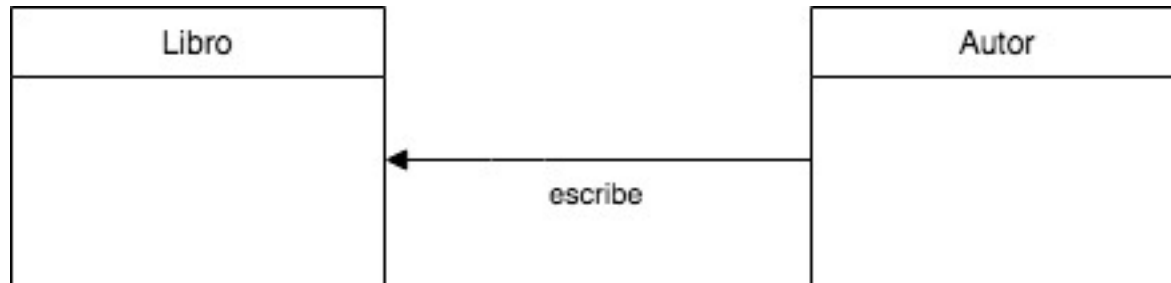
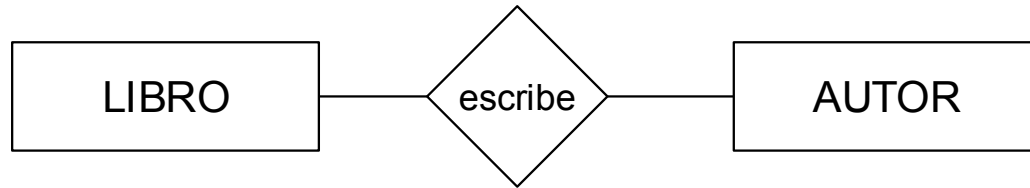
- Asociación o correspondencia entre ejemplares de entidades
  - **Ejemplo:** Correspondencia entre el autor de un libro con el libro que ha escrito
- Un **tipo de relación** es el conjunto de las relaciones de un mismo tipo.
  - **Ejemplo:** Un tipo de relación ESCRIBE relaciona los tipos de entidades LIBRO y AUTOR
- De la misma forma que con las entidades, al tipo de relación lo denominaremos simplemente **relación**

- El **grado** de una relación es el número de entidades que participan en dicha relación
- Cuando la relación se produce entre dos entidades (suele ser el caso más habitual), se denomina **relación binaria**



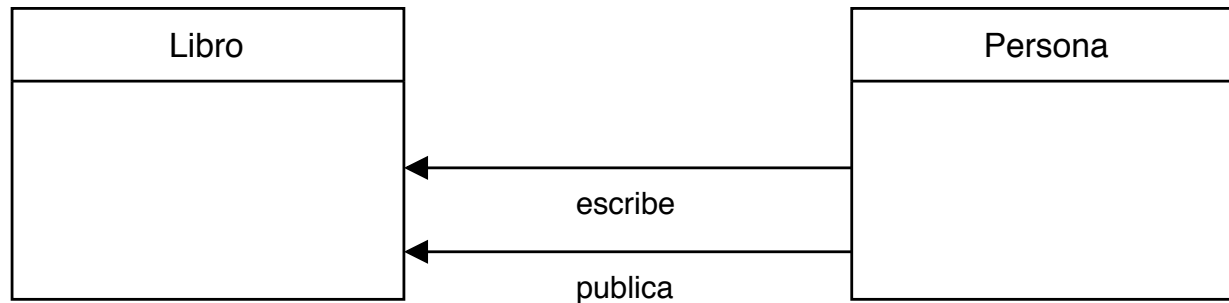
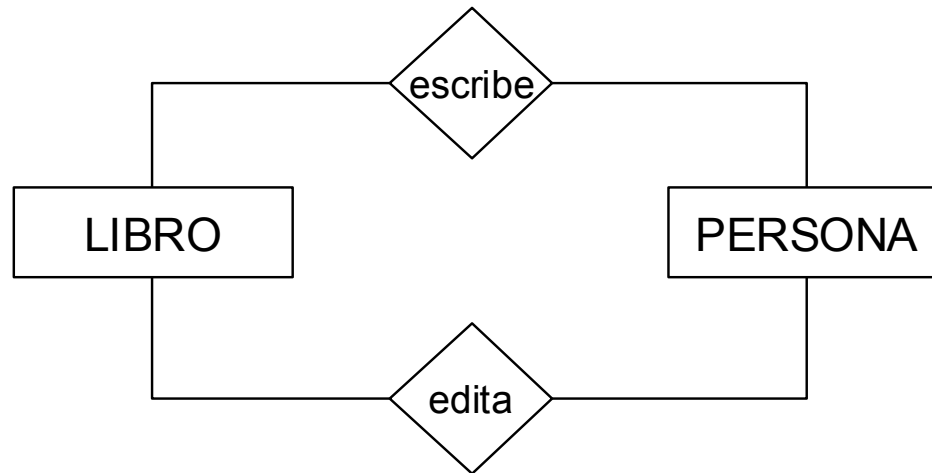
Relación entre las entidades AUTOR y LIBRO

- Representación:

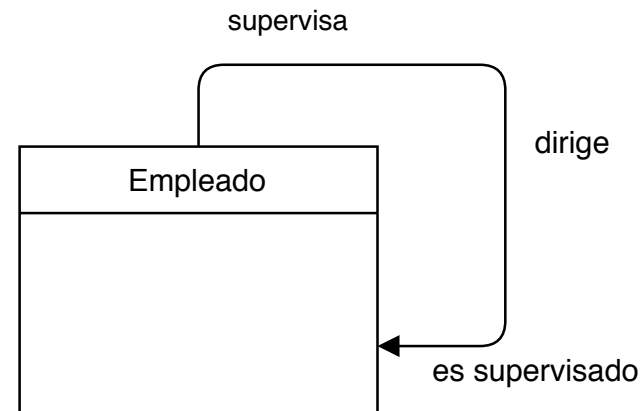
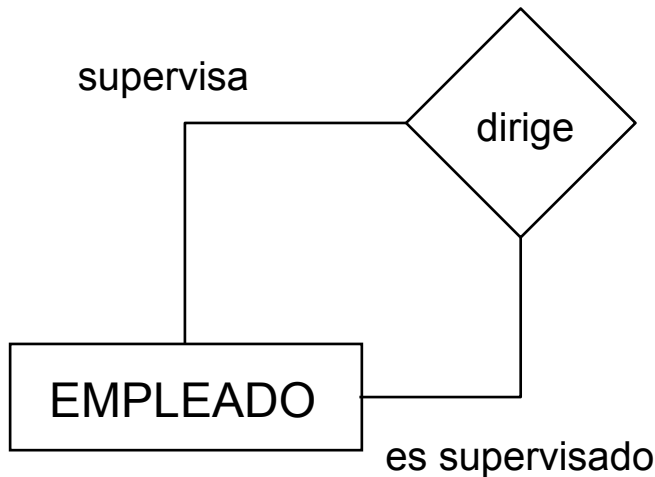




- Entre dos entidades puede existir más de una relación



- Las ocurrencias de una entidad se pueden relacionar con ocurrencias de la propia entidad. A este tipo de relaciones se denominan **relaciones reflexivas** (grado 1)
- El **papel** o **rol** en una relación es la función que desempeña cada una de las entidades en la relación. Esta característica adquiere importancia cuando la relación es reflexiva

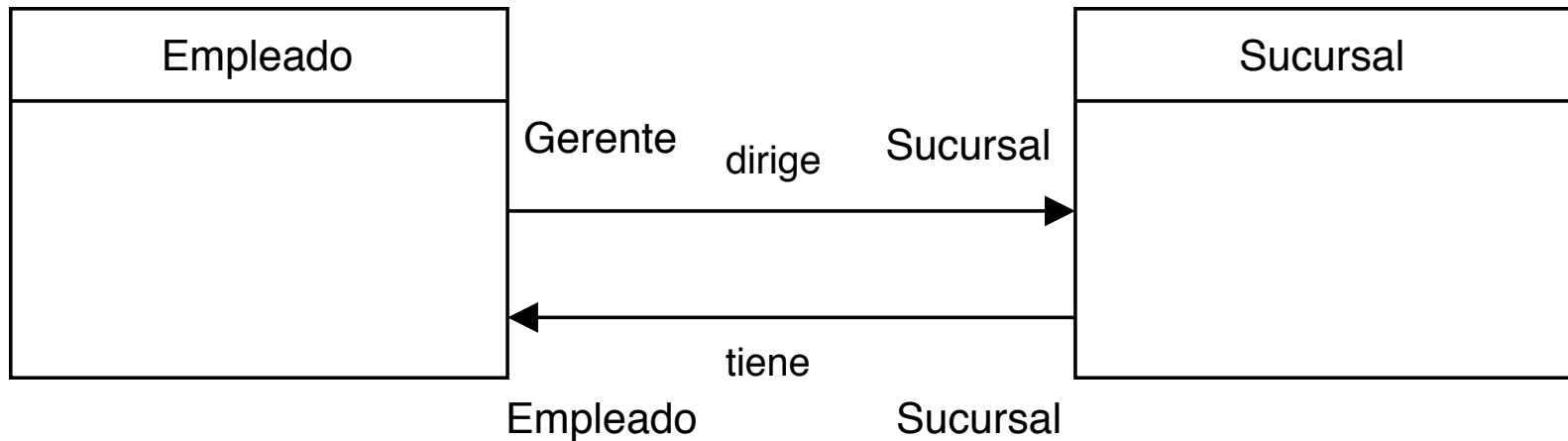


En este caso, una ocurrencia de la misma entidad participa dos veces en la relación, una como director y otra como empleado normal.

Es necesario distinguir el papel en cada sentido de la relación, ya que no tiene el mismo significado los pares formados por los valores (director, trabajador) que los pares formados por los valores (trabajador, director)

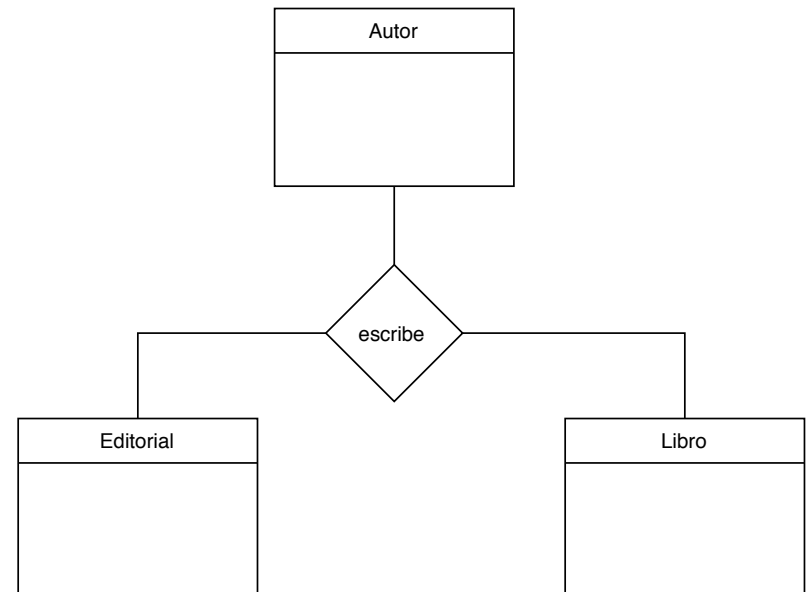
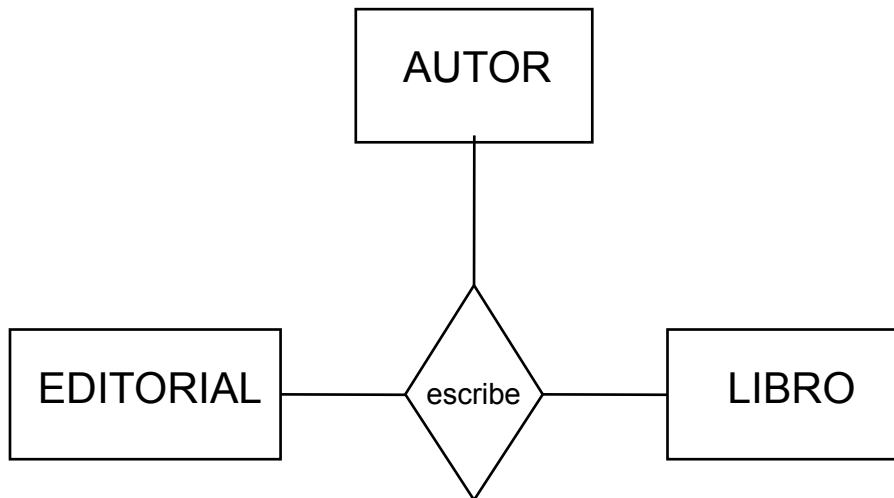
Los nombres de rol también se pueden usar cuando dos entidades están asociadas a través de más de una relación

En general, los nombres de rol no resultan necesarios si la función de las entidades participantes en la relación no es ambigua



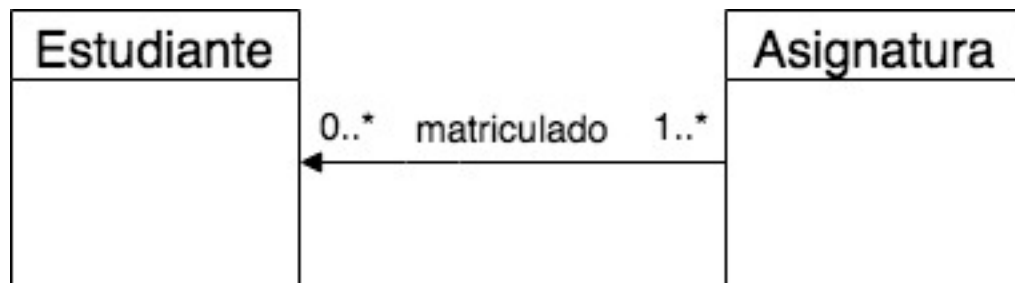
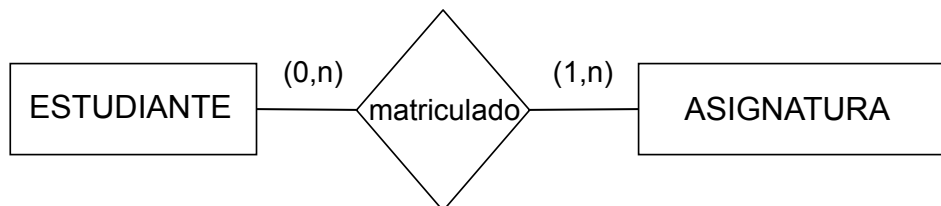
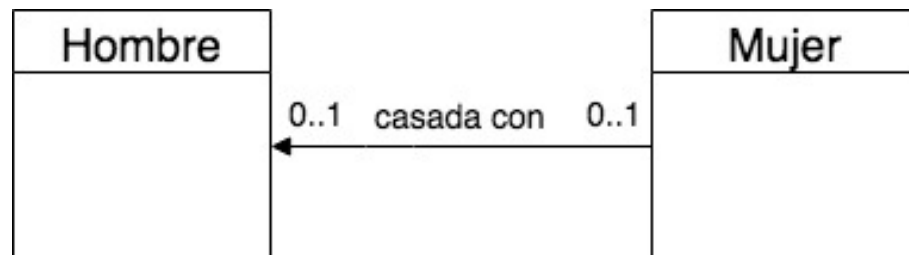
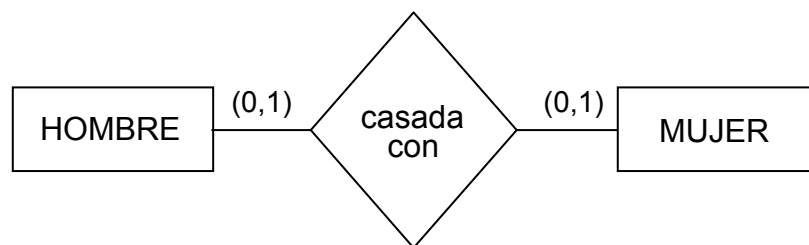
En el ejemplo se muestran dos entidades asociadas mediante dos relaciones distintas, junto con los nombres del rol. Se quiere representar que un gerente dirige una sucursal y una sucursal tiene empleados

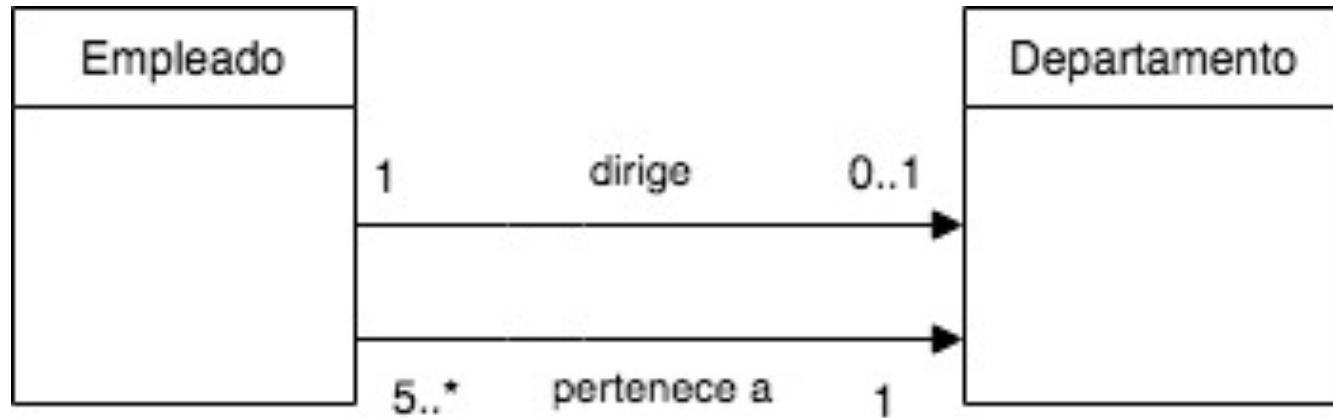
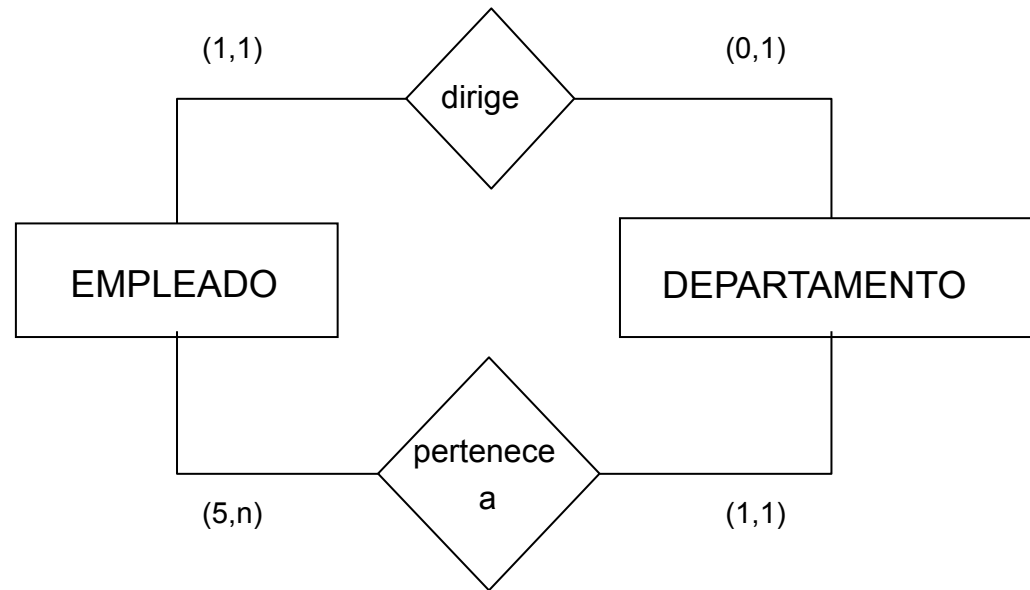
- Aunque las relaciones binarias son las más habituales, hay situaciones en las que es necesario que la relación esté formada por 3 entidades (**relaciones ternarias**)
- Las relaciones de grado mayor que tres se dan con muy poca frecuencia, pero también se pueden dar



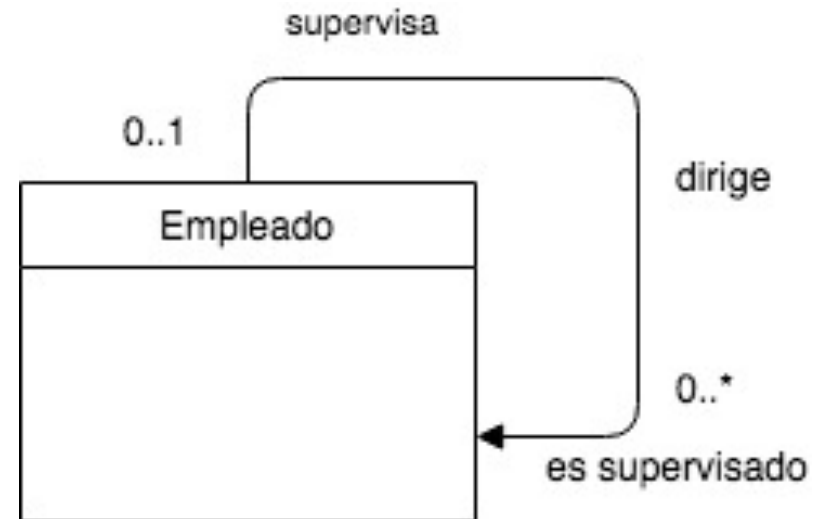
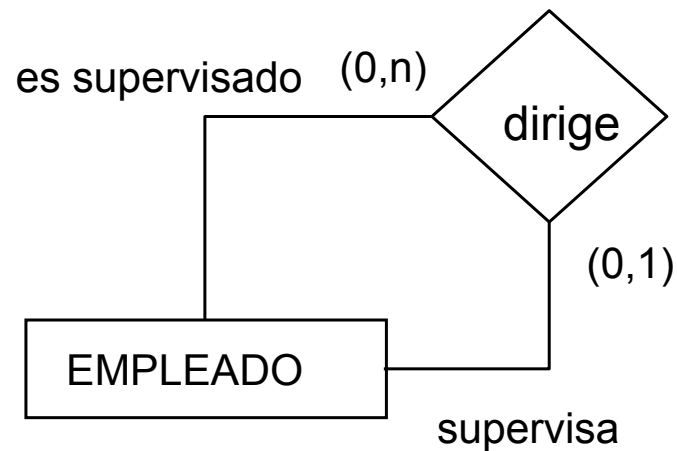
- En su modelo original, Chen definió la **cardinalidad de las relaciones** como el número máximo de ocurrencias de cada entidad que pueden intervenir en la relación
- Para recoger mayor semántica, se puede representar la **cardinalidad máxima** y la **cardinalidad mínima**, que se definen como el número máximo y mínimo de ocurrencias de una entidad que pueden estar relacionadas con una ocurrencia de otra u otras entidades que participan en la relación
- Generalmente se representan con etiquetas del tipo **(0,1)**, **(1,1)**, **(0, n)** o **(1,n)**
  - Para aumentar el contenido semántico, se pueden utilizar etiquetas del tipo **(min, max)**, con  $0 \leq \min \leq \max$  y  $\max \geq 1$

- Ejemplos de cardinalidades (multiplicidad) en relaciones binarias



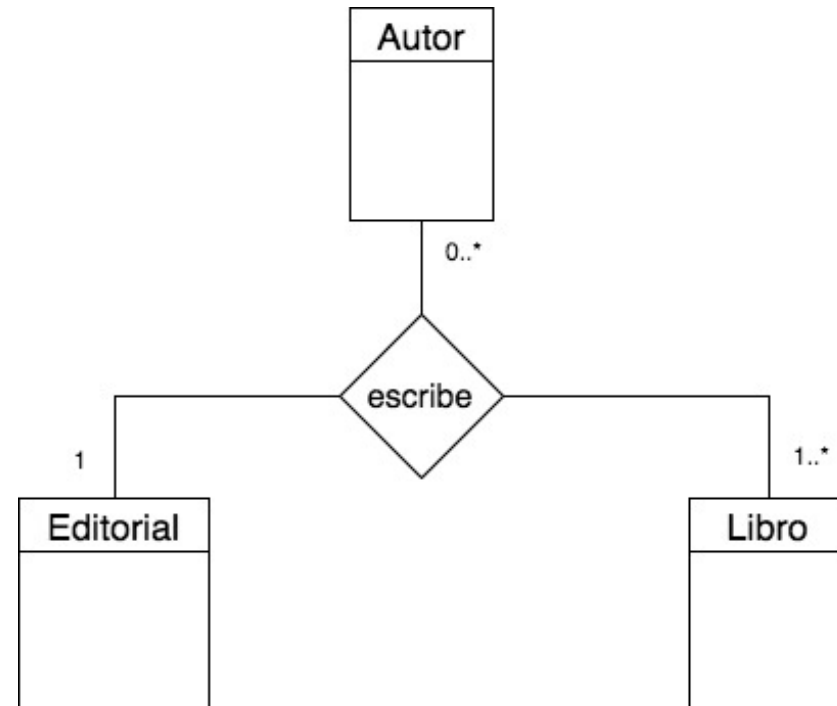
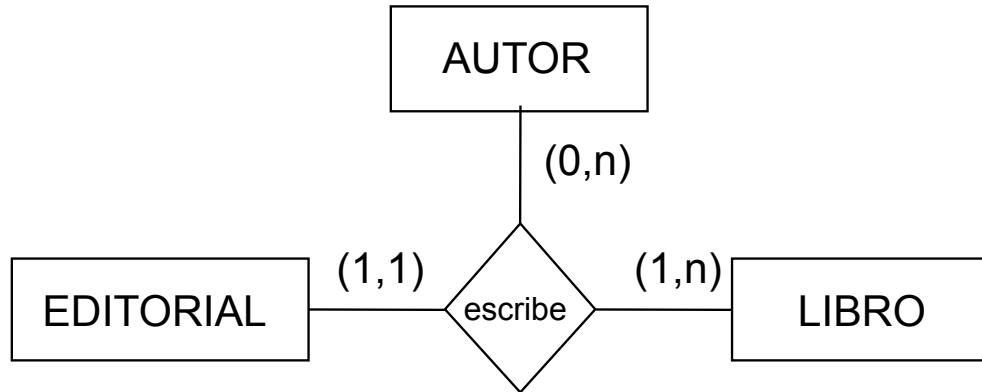


- Ejemplos de cardinalidades (multiplicidad) en una relación reflexiva





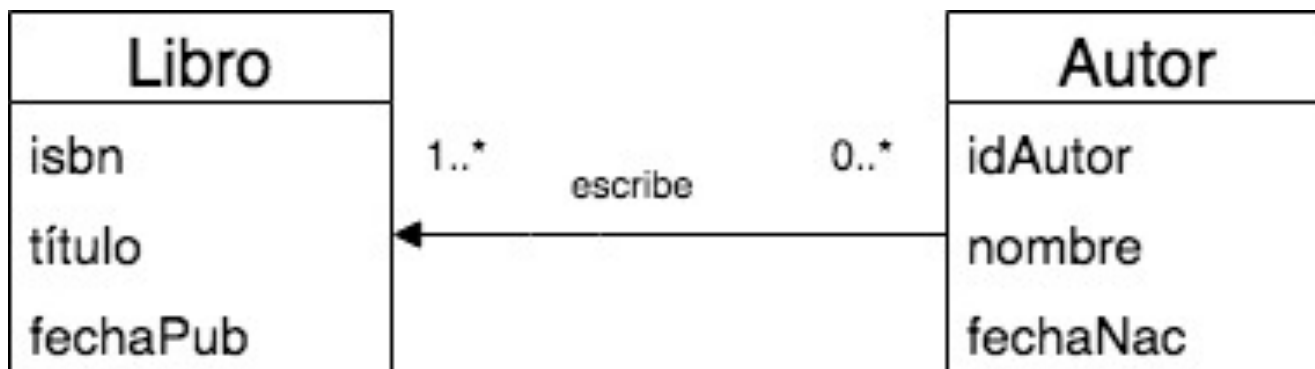
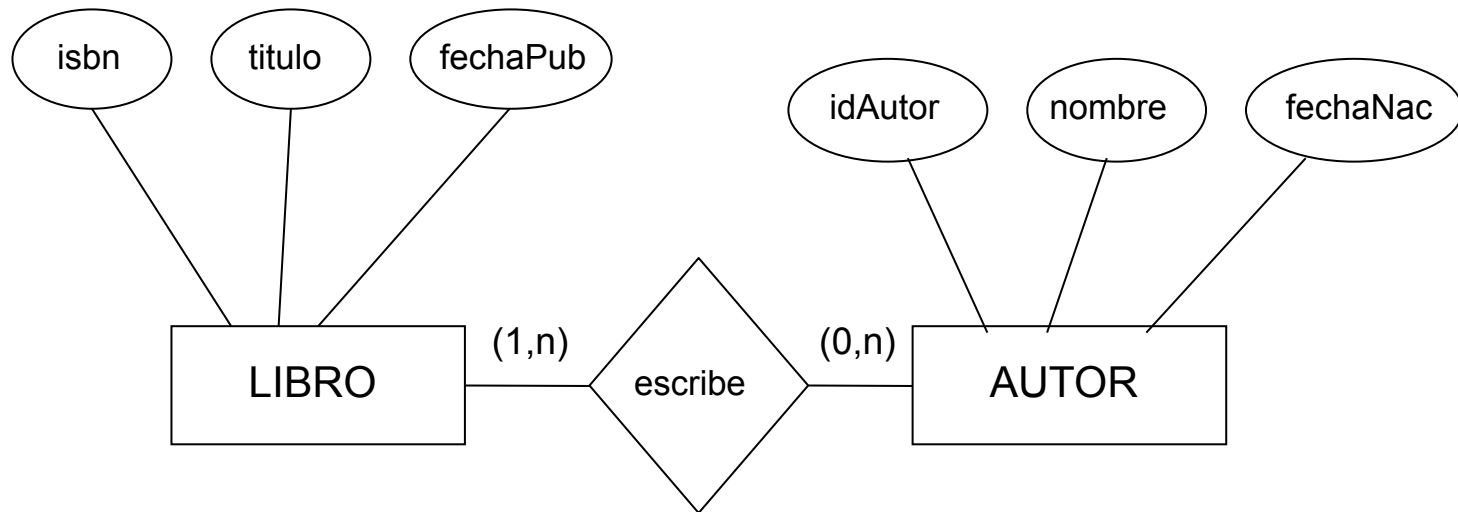
- Ejemplos de cardinalidades (multiplicidad) en una relación ternaria



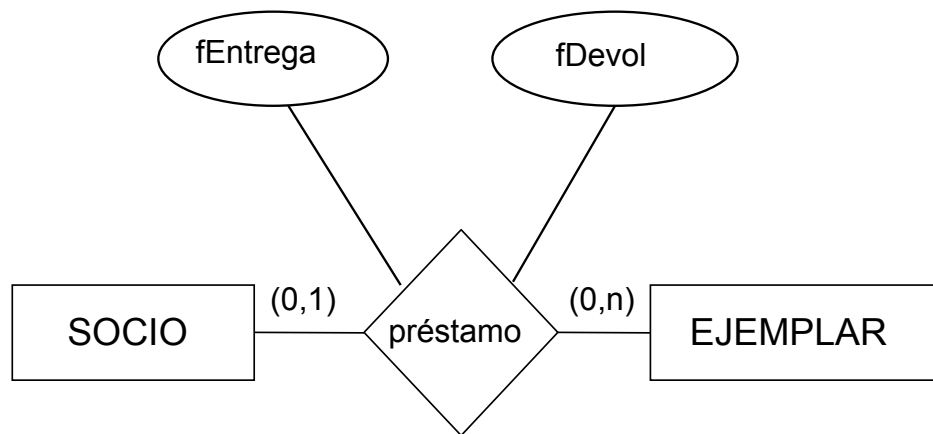
## 2.2.3. Dominio y atributo

- Un **dominio** se define como un conjunto de valores homogéneos con un nombre que lo identifica
- Se conoce como **atributo** a cada una de las propiedades o características que definen una entidad o una relación
- Una característica de una entidad o relación toma valores que pertenecen a un determinado dominio
  - la entidad EMPLEADO puede estar formada por los atributos *dni, nombre, apellidos, dirección, puesto, fechaIngreso*, etc.
  - la entidad LIBRO puede tener los atributos *isbn, título, numPáginas, fechaEdición*, etc.

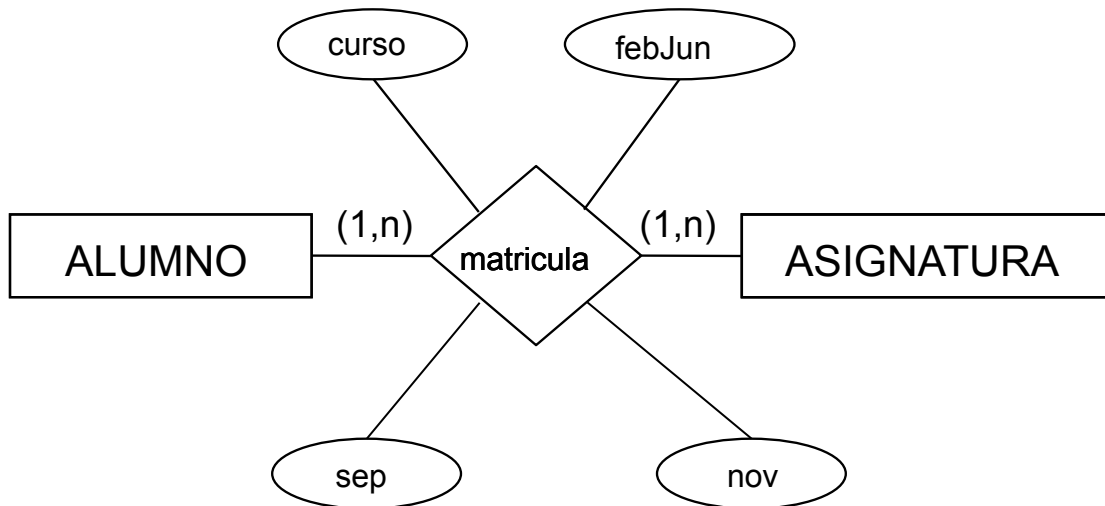
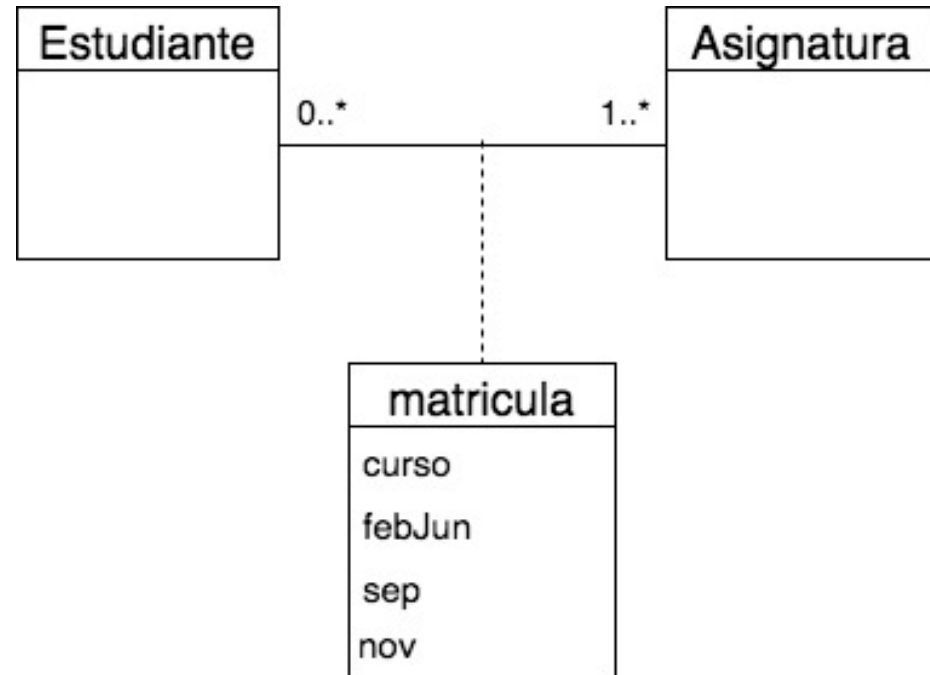
- Representación:



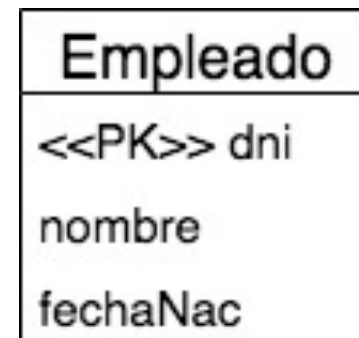
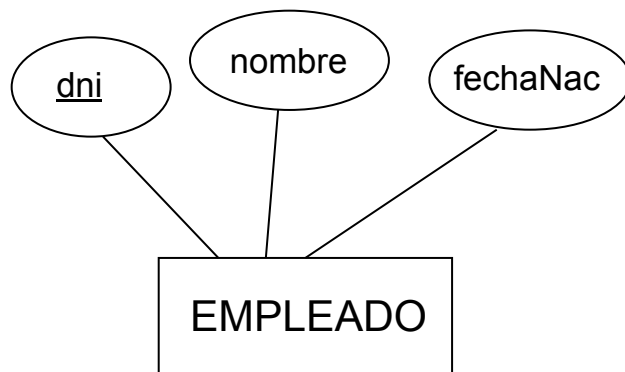
- Cuando la cardinalidad máxima de las relaciones es **uno a uno** o **uno a muchos**, los atributos de las relaciones pueden trasladarse a una de las entidades:
  - uno a muchos → a la parte *muchos* de la relación
  - uno a uno → a cualquiera de ellas



- Si la cardinalidad de la relación es muchos a muchos, los atributos deben especificarse en la relación, pues están determinados por la combinación de las entidades participantes

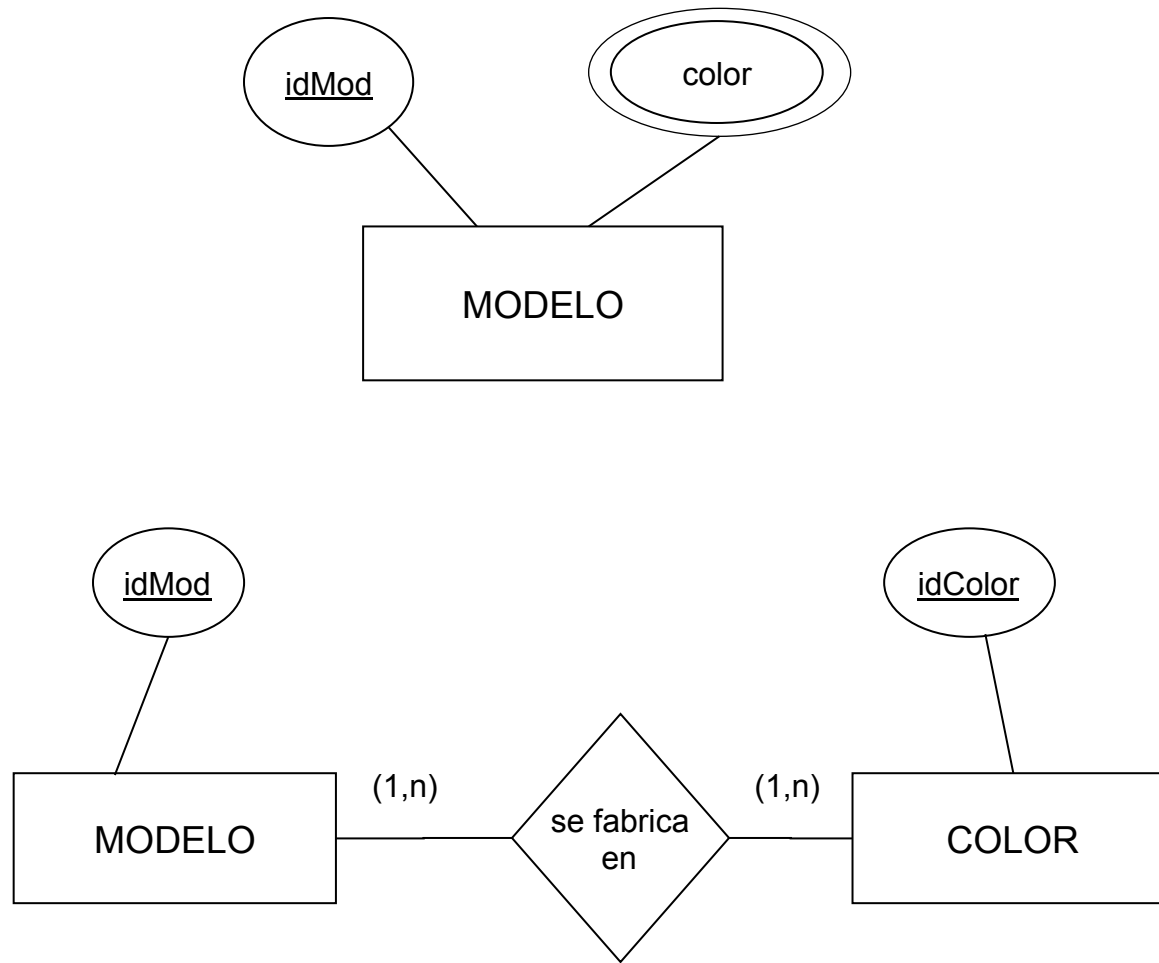


- La restricción de **unicidad** del modelo E-R obliga a que una ocurrencia de una entidad debe distinguirse de las demás
- Los atributos que sirven para identificar de manera única las ocurrencias dentro de una entidad se denominan **atributos clave**
- Se representan subrayando con una línea continua los nombres de los atributos que forman la clave
- PK = primary key



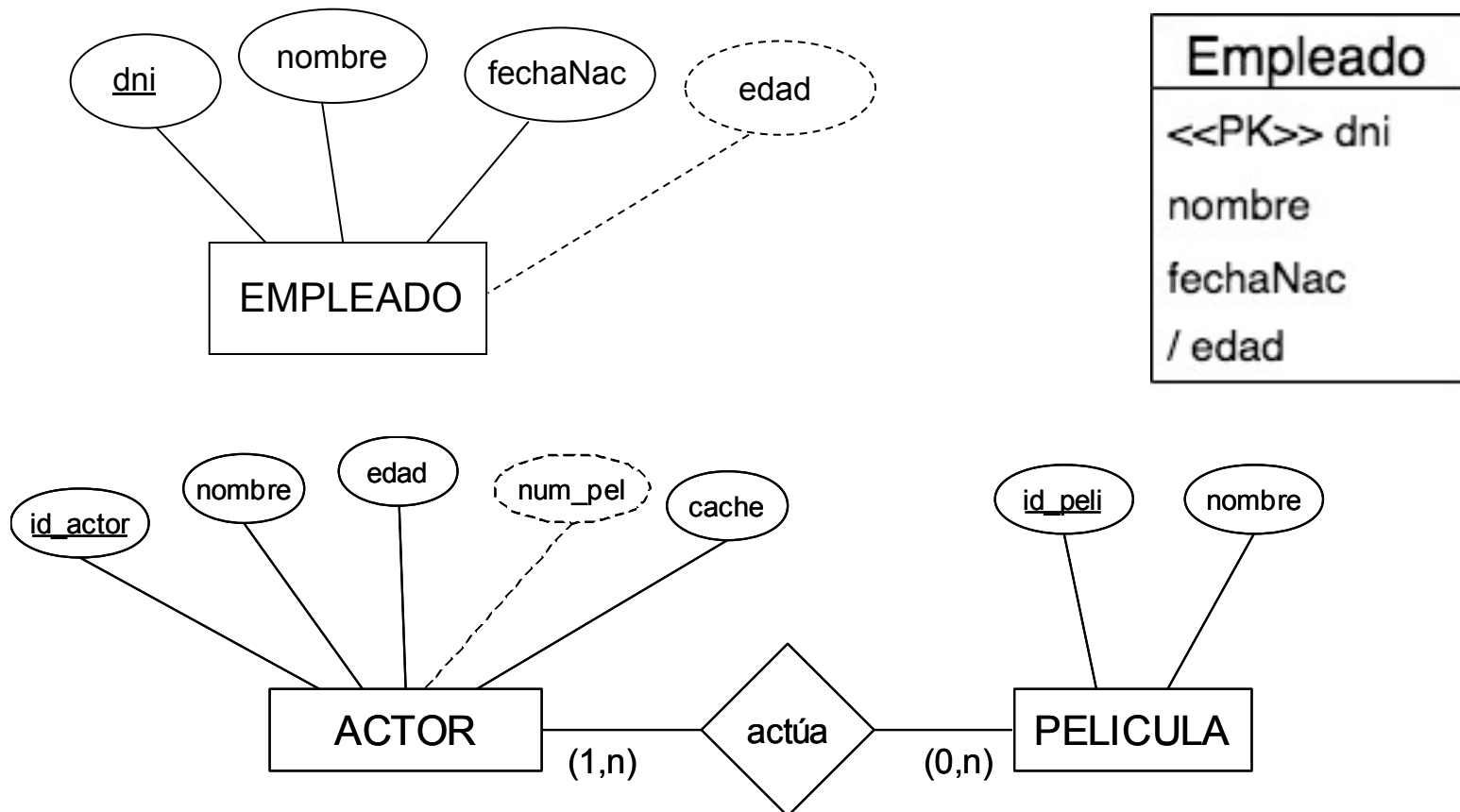
- En general, los atributos tienen un solo valor para una ocurrencia en particular (**atributos univalorados**)
  - el atributo sexo de un determinado empleado puede tomar un único valor
- En algunos casos, los atributos pueden tomar diferentes valores para una misma ocurrencia (**atributos multivalorados**)
- Aunque el modelo E-R admite este tipo de atributos (incluso tiene una figura para representarlo), en este curso no los vamos a considerar en nuestro diseño conceptual

- Ejemplo de transformación del atributo multivalorado **color**

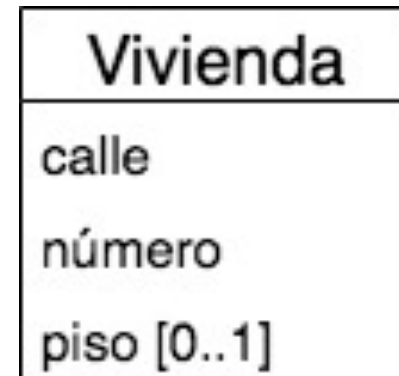
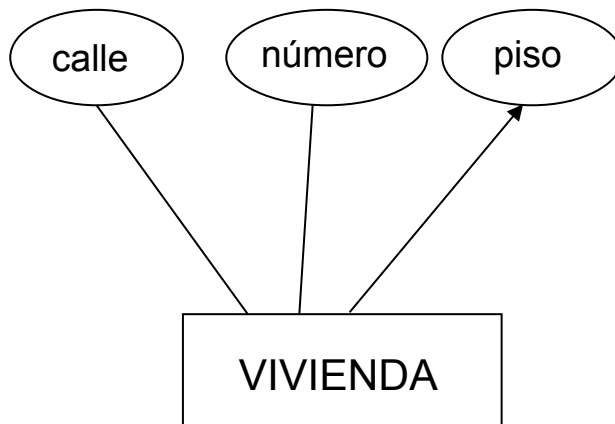




- Un **atributo derivado** es aquél cuyo valor puede obtenerse a partir de los valores de otros atributos o relaciones
- Se representan de la misma forma que los atributos normales pero con el trazo discontinuo

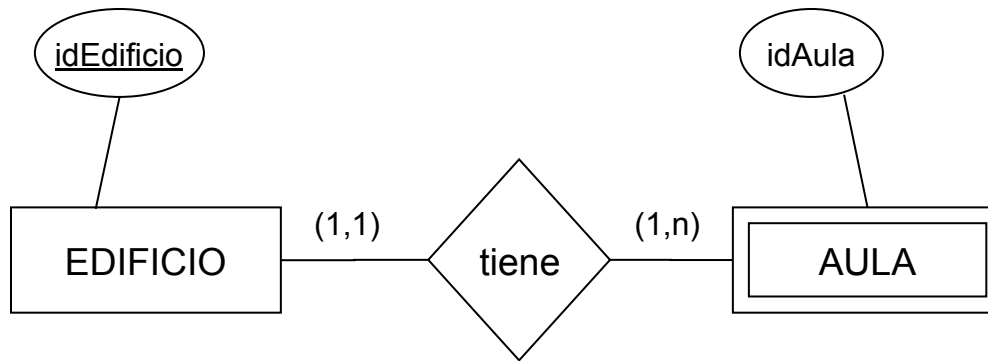


- Un **atributo opcional** es el que puede admitir valores nulos
- Se representa mediante una flecha apuntando al atributo opcional

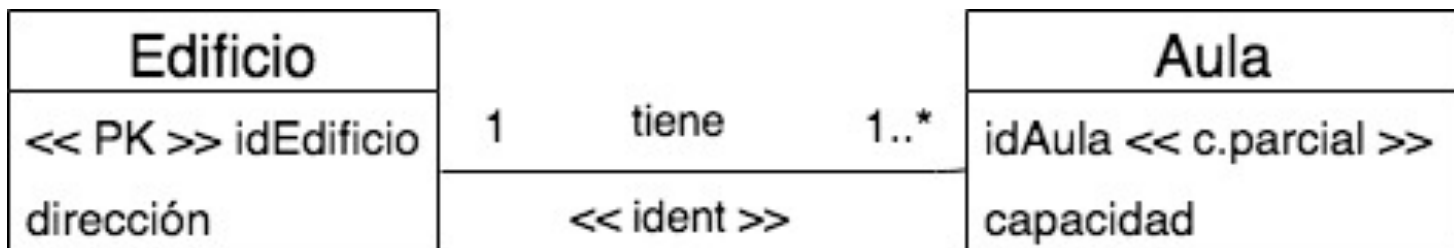


## Entidades fuertes y débiles

- Una entidad débil es una entidad cuyos atributos no la identifican completamente. Solo la identifican de forma parcial
  - Esta entidad debe participar en una relación que ayude a identificarla
- 
- **Ejemplo:** Consideremos las entidades **edificio** y **aula** y supongamos que puede haber aulas con el mismo número en edificios diferentes. En el edificio E30 pueden estar las aulas A1.1 y la A2.1, y en el edificio E21 pueden estar las aulas A1.1 y la A1.2
    - En este caso, el identificador **aula** no identifica completamente un aula. Para identificar unívocamente un aula, es necesario tener en cuenta en qué edificio está situada. De hecho, podemos identificar un aula mediante la relación que la asocia a un único edificio. El identificador del edificio donde está situado junto con el número de aula lo identifican completa y unívocamente



- UML no dispone de una notación especial
- Usaremos el estereotipo **<<c.parcial>>** para indicar la clave parcial y el estereotipo **<<ident>>** para indicar la composición identificadora
- Otros autores representan las entidades débiles sin clave



## ▪ Ejemplo

- Los estudiantes se pueden matricular de varias asignaturas y, al menos, de una pero deben hacerlo en un grupo concreto
- Queremos almacenar la calificación que obtienen los alumnos en cada asignatura
- Cada profesor debe pertenecer a un departamento concreto
- Todos los departamentos tienen un director que, además, es un profesor
- Un profesor puede impartir varios grupos de la misma asignatura o de asignaturas diferentes
- Un grupo de una asignatura lo debe impartir, al menos, un profesor
- Las clases de cada asignatura se imparten en días, horas y aulas determinadas
- Para simplificar, consideraremos que los atributos de las entidades son los "normales"