

B2

Modelos de Datos y Diseño de Bases de Datos Relacionales

Tema 6. Diseño Lógico

6. Diseño Lógico

1

Objetivos

- ❑ Comprender la conveniencia y ventajas de disponer de un esquema lógico de BD **independiente** del SGBD comercial particular
- ❑ Entender y aplicar las reglas de **transformación** de un esquema conceptual en Modelo Entidad-Relación (MER) en un esquema lógico en el Modelo Relacional (MR)
- ❑ Aprender cómo **evitar** la posible **pérdida de semántica** al traducir elementos del MER a elementos del MR
- ❑ Adquirir **estrategias** para elegir de la opción de diseño lógico más adecuada entre varias alternativas posibles
- ❑ Conocer guías y recomendaciones para traducir un esquema en el MR a un esquema en el modelo de datos específico soportado por el SGBD de implementación

6. Diseño Lógico

2

Contenidos

- 6.1. Objetivos y fases del Diseño Lógico de BD
- 6.2. Diseño Lógico Estándar
- 6.3. Diseño Lógico Específico
 - ▣ Primeras indicaciones: lo practicaremos una vez conozcamos el SQL
- Ejemplos resueltos

- ❖ Boletines de ejercicios (exámenes AV)
 - ❖ Boletín 8. Diseño Lógico 1
 - ❖ Boletín 9. Diseño Lógico 2

6. Diseño Lógico

3

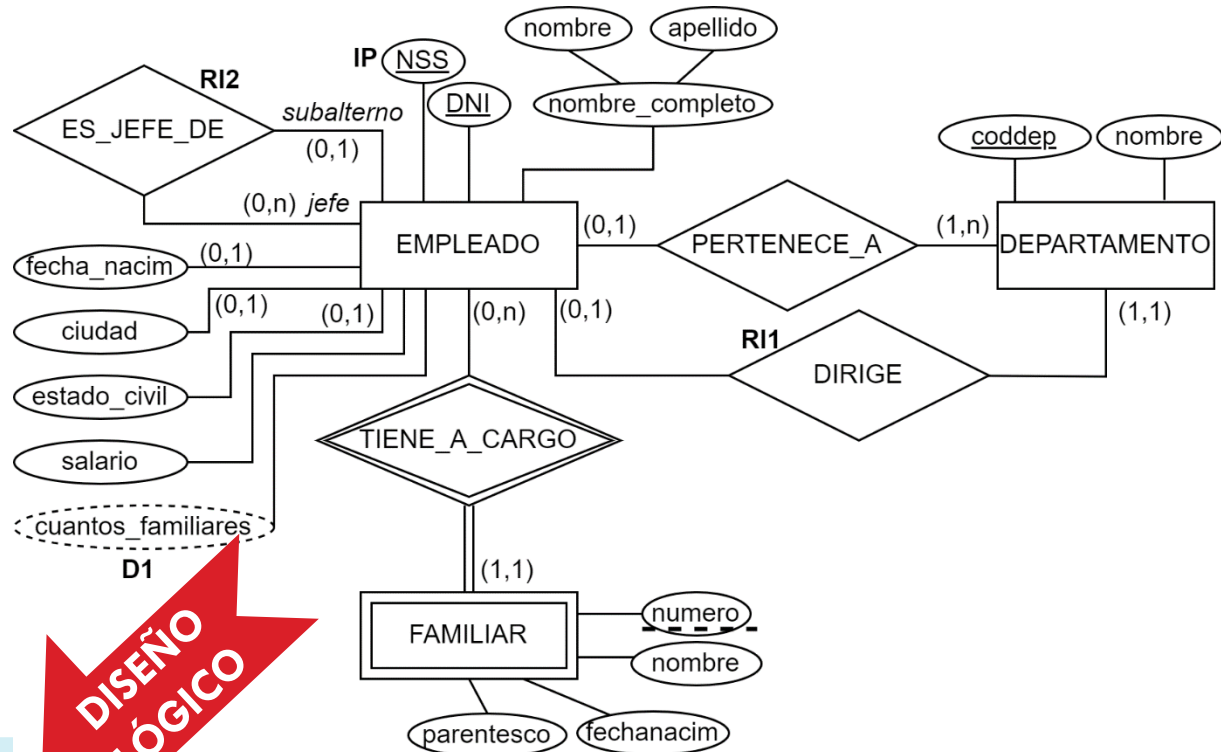
Bibliografía

- [CB 2015] Connolly, T.M.; Begg C.E.: ***Database Systems: A Practical Approach to Design, Implementation, and Management***, 6th Edition. Pearson. (Capítulo 17).
- [EN 2016] Elmasri, R.; Navathe, S.B.: ***Fundamentals of Database Systems***, 7th Edition. Pearson. (Capítulo 9).

6. Diseño Lógico

4

- ¿Cómo convertimos un **esquema conceptual** de datos...
- ... en un **esquema lógico** de base de datos?



EMPLEADO

<u>nss</u>	dni	nombre	apellido	fechanacim	ciudad	est_civil	salario	nssjefe	dep	cuantos_familiares
...										

DEPARTAMENTO

nombre	<u>coddep</u>	nssdire
...		

FAMILIAR

<u>nssemp</u>	<u>numero</u>	nombre	fechanacim	parentesco
...				

6.1 Objetivos y fases del diseño lógico

Objetivos

5

- ❑ El objetivo principal es **transformar el esquema conceptual (EC) de datos en el esquema lógico (EL) de datos**
- ❑ *Otros objetivos del diseño lógico son ...*
 - ▣ Minimizar redundancias en los datos
 - ▣ Conseguir máxima simplicidad
 - ▣ Evitar cargas suplementarias de programación para lograr ...
 - ▣ una estructura lógica correcta, adecuada para los datos
 - ▣ y capaz de soportar las transacciones requeridas

6.1 Objetivos y fases del diseño lógico

Fases

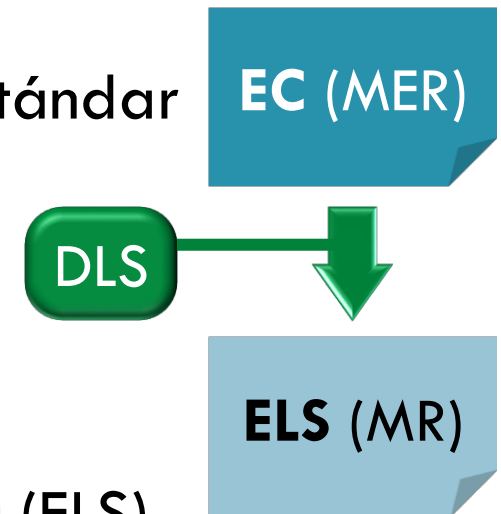
6

❑ **Diseño Lógico eStándar (DLS)**

- ❑ Traducción del EC independiente del SGBD específico
- ❑ Se elige el Modelo de Datos Lógico, pero aún no el SGBD comercial concreto

- ❑ Uso de un Modelo Lógico de Datos estándar

- **Relacional** ⇐
- Red
- Jerárquico
- Orientado a Objetos



- ❑ Obtiene el **Esquema Lógico eStándar (ELS)**

En el Modelo Relacional para describir el ELS se puede usar...

- **Pseudolenguaje** [TABLA(colu1, colu2, ...) y especificaciones]
- Lenguaje de Definición de Datos (LDD) de ANSI SQL

6.1 Objetivos y fases del diseño lógico

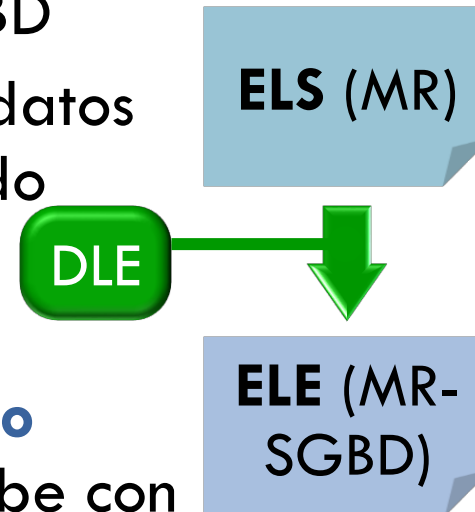
Fases

7

□ Diseño Lógico Específico (DLE)

- ▣ Ya elegido el SGBD específico (comercial)
- ▣ Adaptación del ELS al SGBD
- ▣ Uso del modelo lógico de datos particular del SGBD elegido
 - **Oracle**, MySQL, MariaDB, PostgreSQL, SQLServer, etc.

- ▣ Obtiene el **Esquema Lógico Específico** (ELE) y se describe con el lenguaje propio del SGBD (dialecto SQL)
 - **SQL de Oracle**, SQL de MySQL, SQL de MariaDB, etc.



ORACLE®

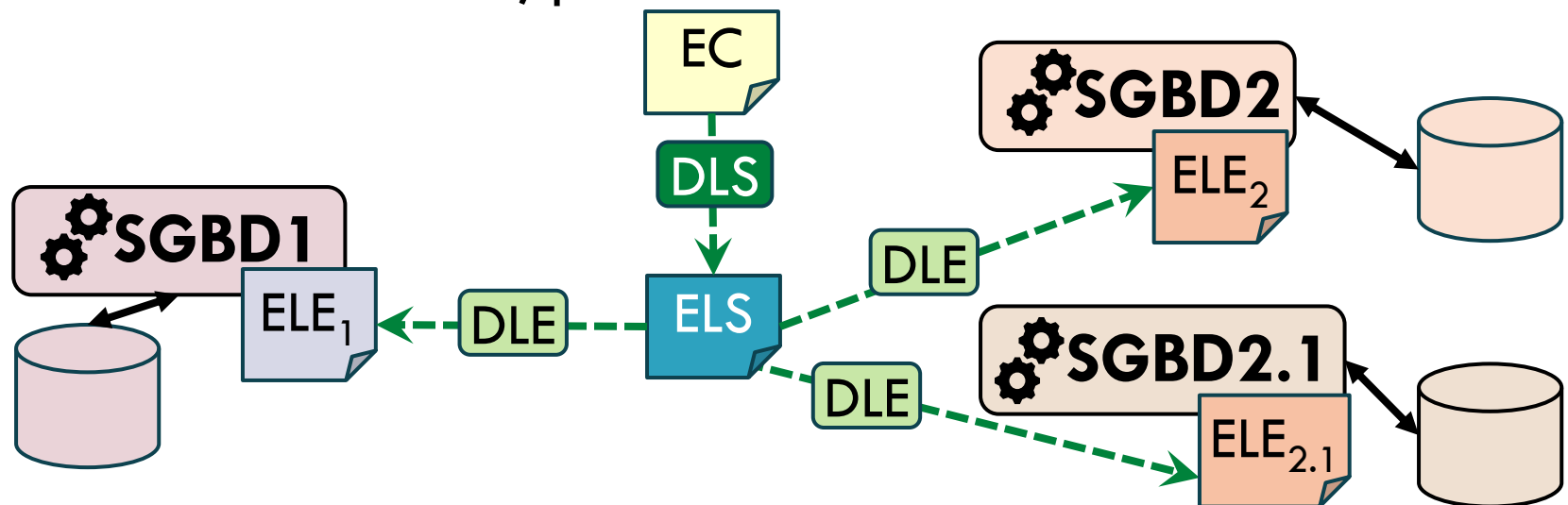


6.1 Objetivos y fases del diseño lógico

Objetivos

8

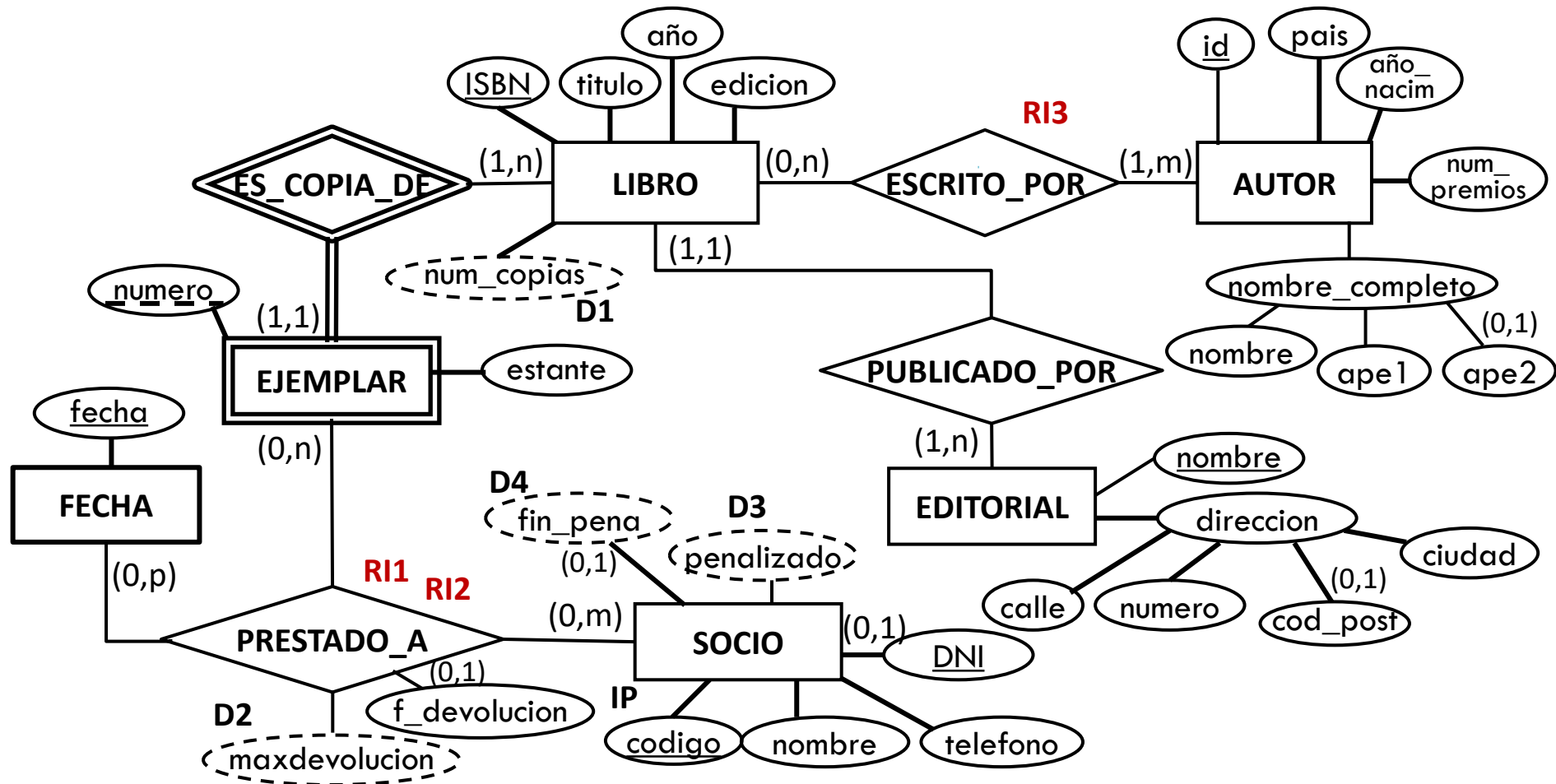
- Dividir el Diseño Lógico en DLS+DLE garantiza que se realiza el diseño con la **máxima portabilidad**
 - ☑ Gracias a la **introducción “tardía” del SGBD** específico (en el DLE) es posible **disminuir el esfuerzo** para...
 - ▣ Implementar el esquema lógico en **distintos SGBD**
 - ▣ **Migrar** a nuevas versiones del mismo SGBD o a otro SGBD diferente, pero del mismo modelo de datos



Esquema Conceptual - ejemplo

9

□ Esquema Conceptual (en el MER) de partida



6.2 Diseño Lógico Estándar

Pasos o actividades

10

1. Obtener relaciones (tablas)
2. Validar las relaciones contra las transacciones de usuario
3. Revisar las restricciones de integridad
4. Validar el esquema lógico con los usuarios

1. Obtener Relaciones (tablas)

11

- Crear **relaciones** para representar *tipos de entidad*, *tipos de relación*, y ciertos *atributos* del esquema conceptual de datos
- Obtendremos relaciones (tablas) a partir de cada...
 1. Tipo de **entidad** [fuerte]
 2. Tipo de **entidad débil**
 3. **Jerarquía**
 4. Tipo de relación binaria **1:N** y **recursiva 1:N**
 5. Tipo de relación binaria **1:1**
 6. Tipo de relación binaria **1:1 recursiva**
 7. Tipo de relación binaria **M:N**
 8. Tipo de relación **n-aria**
 9. **Atributo multivalorado**

1. Obtener Relaciones (tablas)

12

- Cada relación (tabla) obtenida la describiremos mediante una **ficha**, utilizando la siguiente plantilla:

TABLA_EJEMPLO (atributo1, atributo2, atributo3, atributo4, atributo5, atributo6, atributo7, atributo8, atributo9)

Admiten NULL: atributo7, atributo9

Clave Primaria: atributo2

Claves Alternativas (UNIQUE): 1. atributo3 ; 2. (atributo4, atributo5);

Claves Ajenas (FOREIGN KEY):

1. (atributo7) Referencia_a UNATABLA(clave)

2. (atributo8, atributo9) Referencia_a OTRATABLA(clave1,clave2)

Derivados:

1. atributo6 = atributo2*20/100

...

Comprobar:

1. atributo1 IN ('SI', 'NO')

2. atributo6 > 0

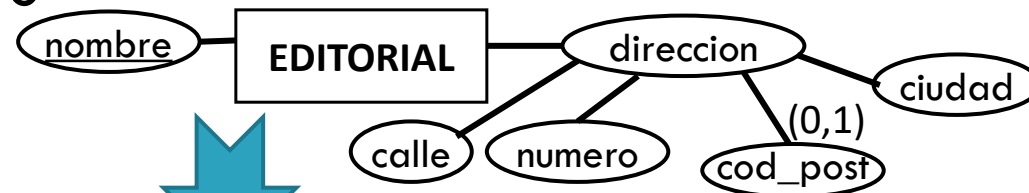
3. atributo7 >= atributo3

...

1.1. Tipo de **entidad fuerte**

13

- Se traduce a una **relación** (tabla)
 - ▣ Recomendado: usar el mismo nombre o uno muy similar
- La relación incluye estos **atributos**
 - ▣ Uno para cada *atributo simple* del tipo de entidad
 - ▣ Uno para cada componente de los *atributos compuestos*
- Y estas **claves**
 - ▣ La clave primaria se deriva del identificador principal
 - ▣ Una clave alternativa (UNIQUE) por cada identificador alternativo

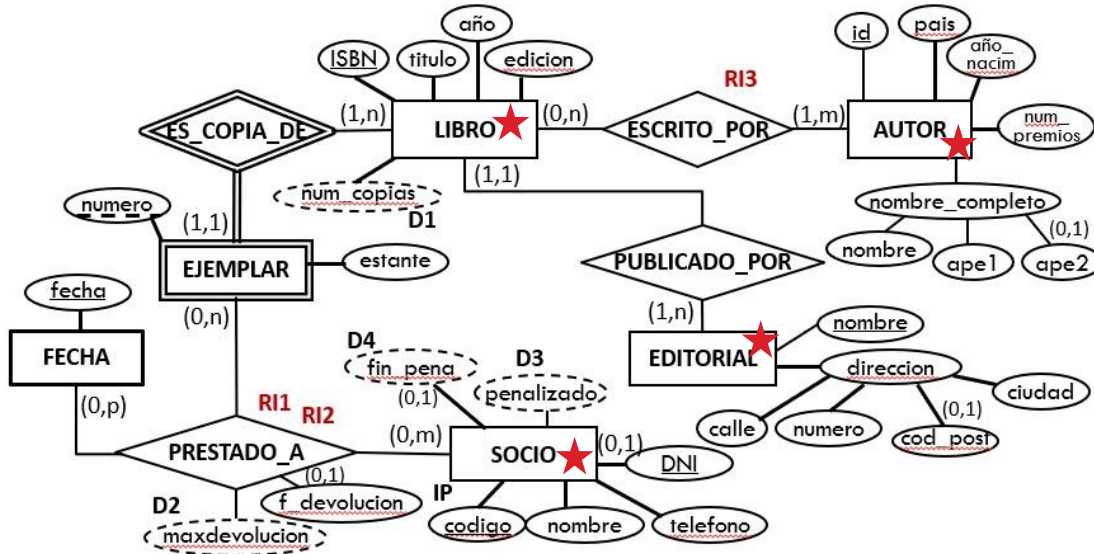


❗ Los atributos deben incluir sus restricciones: si admiten o no el nulo, si son calculados, etc.

EDITORIAL(nombre, calle, numero, cod_post, ciudad)
Admiten NULL: cod_post
Clave primaria: nombre

Ejemplo. Traducción de tipos de entidad fuertes

14



EDITORIAL(nombre,calle, numero,cod_post,ciudad)
 Admiten NULL: **cod_post**
 Clave primaria: **nombre**

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
 Admiten NULL: **DNI,fin_pena**
 Clave primaria: **codigo**
 Clave alternativa: **DNI**

AUTOR(id,nombre,pais,año_nacim,num_premios,nombre,ape1,ape2)
 Admiten NULL: **ape2**
 Clave primaria: **id**

LIBRO(ISBN,título,año,edicion,num_copias)
 Admiten NULL: Ninguno
 Clave primaria: **ISBN**
 Derivado: **num_copias** (nota:falta fórmula de cálculo)

1.2. Tipo de **entidad débil**

15

- Se traduce a una **relación** (tabla) de igual modo que en el *paso 1.1* (tipo de entidad fuerte)
- Puesto que la **clave primaria** del tipo de entidad débil se obtiene (total o parcialmente) de la de su tipo de entidad fuerte, NO puede ser definida hasta traducir el tipo de relación que la conecta a su fuerte



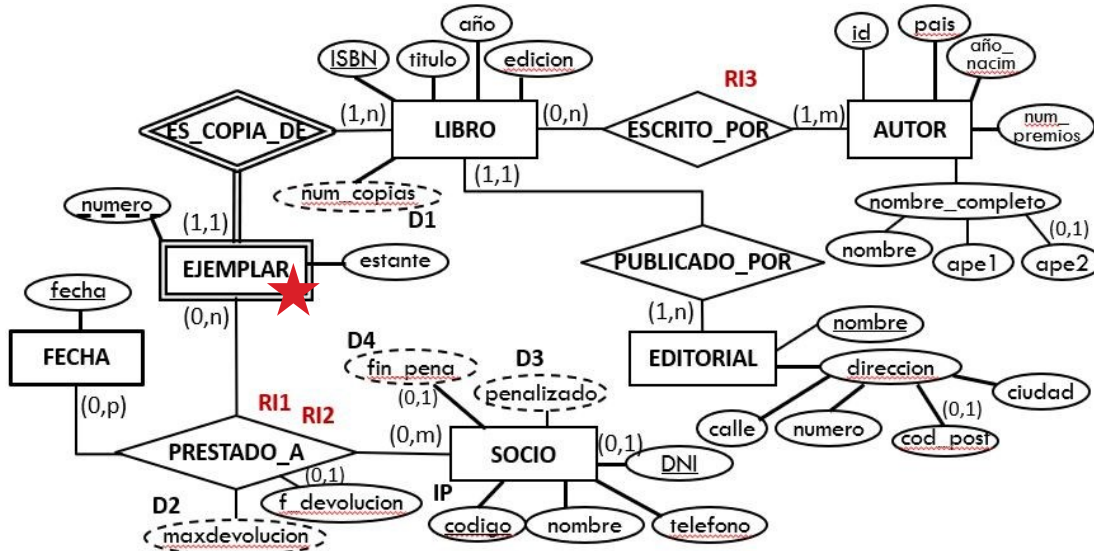
EJEMPLAR(numero, estante)
Admiten NULL: Ninguno
Clave primaria: *<Pendiente>*

① La clave primaria la podremos definir cuando se traduzca el tipo de relación identificador



Ejemplo. Traducción de tipos de entidad débil

16



EDITORIAL(nombre, calle, numero, cod_post, ciudad)
Admiten NULL: **cod_post**
Clave primaria: **nombre**

SOCIO(codigo, DNI, nombre, telefono, penalizado, fin_pena)
Admiten NULL: **DNI**, **fin_pena**
Clave primaria: **codigo**
Clave alternativa: **DNI**

AUTOR(id, nombre, pais, año_nacim, num_premios, nombre, ape1, ape2)
Admiten NULL: **ape2**
Clave primaria: **id**

LIBRO(ISBN, título, año, edicion, num_copias)
Admiten NULL: Ninguno
Clave primaria: **ISBN**
Derivado: **num_copias** (nota: falta fórmula de cálculo)

EJEMPLAR(numero, estante)
Admiten NULL: Ninguno
Clave primaria: **<Pendiente>**

1.3. Traducción de Jerarquías

17

- La traducción depende del tipo de la jerarquía
 - ▣ Disjunta y Total
 - ▣ Disjunta y Parcial
 - ▣ Solapada y Total
 - ▣ Solapada y Parcial

1.3. Traducción de Jerarquías

18

□ Jerarquía disjunta y total

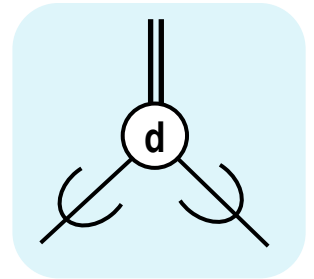
▣ Da lugar a varias relaciones: **una relación por cada combinación supertipo/subtipo**

■ **Tantas relaciones como subtipos**

■ Cada relación contendrá todos los **atributos del correspondiente subtipo y los del supertipo**

▶ El supertipo NO se traduce a relación

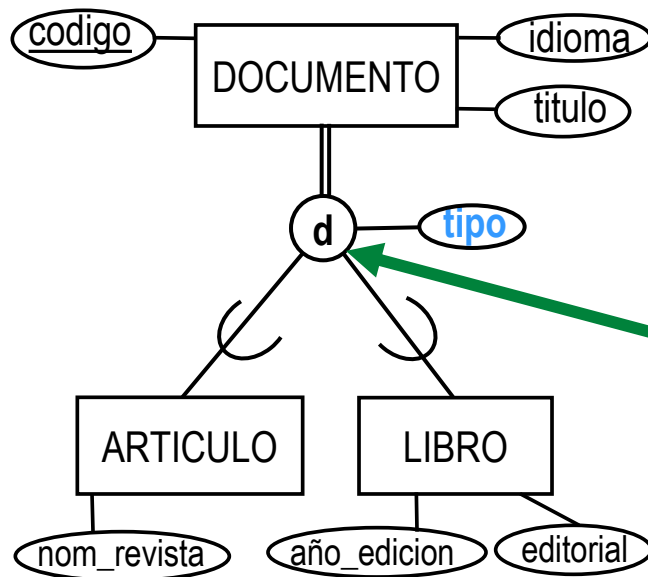
▣ Hay que definir una **restricción de integridad general** (aserto) para garantizar la disyunción



1.3. Traducción de Jerarquías

19

Ejemplo 1 de transformación de una **jerarquía disjunta y total**



ARTICULO (`codigo`, `titulo`, `idioma`, `nom_revista`)
Clave primaria: `codigo`

LIBRO (`codigo`, `titulo`, `idioma`, `año_edicion`, `editorial`)
Clave primaria: `codigo`

Tenemos que añadir algo que garantice la **disyunción**

► Se debe asegurar que **Ninguna instancia de tipo ARTICULO puede ser a la vez instancia de tipo LIBRO**

OOPS!

Esto está redactado a **nivel Conceptual**. Ahora estamos en el **nivel Lógico**, donde sólo hay **relaciones** (tablas) y **claves ajenas** (vínculos)!

❶ El atributo **discriminante** no aparece en ninguna de las relaciones resultado de la traducción

Hay que redactar ASERTOS

Inciso: Redactar ASERTOS

20

- Un aserto es una **restricción de integridad** que **afecta a más de una relación** y expresa una **condición** que sus tuplas deben cumplir

¿Cómo **redactar asertos**?

- Expresar **lo que NO puede ocurrir**
- Usar como vocabulario los **nombres de relación y de atributo**
 - ▣ En el esquema lógico ya no hay tipos de entidad, ni tipos de relación, ni instancias
 - ▣ Tenemos relaciones, atributos, tuplas, valores...

Inciso: Redactar ASERTOS

21

- Aplicamos a nuestro ejemplo:
 - ▶ Se debe comprobar que Ninguna instancia de tipo ARTICULO puede ser a la vez instancia de tipo LIBRO
 - Eso ocurriría si hubiera un artículo y un libro con el mismo código


1) Redactamos el **aserto** usando el **vocabulario** correcto

- ▶ Se debe comprobar que ninguna **tupla** de la **relación** ARTICULO tiene el mismo valor en el **atributo** “codigo” que alguna **tupla** de la **relación** LIBRO

2) Mejoramos la **redacción** del aserto para que sea algo más **formal**

- ▶ **Comprobar** que no existe una tupla de ARTICULO tal que el valor de “codigo” **esté entre** los valores de “codigo” de las **tuplas** de LIBRO

3) Le damos un **nombre** al aserto y formateamos su redacción con estilo de **pseudocódigo**



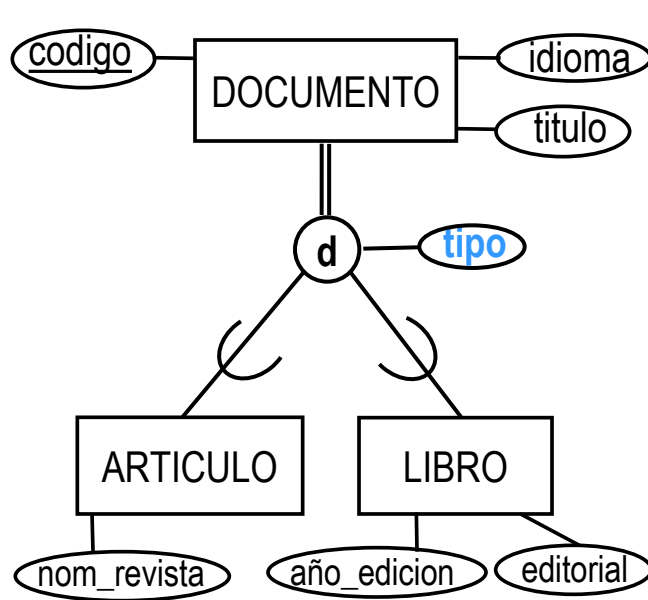
```
ASERTO RI_articulo_xor_libro
COMPROBAR_QUE (
    NO_EXISTE (una tupla de ARTICULO
                tal que el valor de “codigo”
                ESTÉ_ENTRE (valores de “codigo”
                           de LIBRO)));
```

1.3. Traducción de Jerarquías

¡Retomamos!

22

Ejemplo 1 de transformación de una jerarquía disjunta y total



ARTICULO (código, título, idioma, nom_revista)

Clave primaria: código

LIBRO (código, título, idioma, año_edicion, editorial)

Clave primaria: código

ASERTO RI_articulo_xor_libro ←

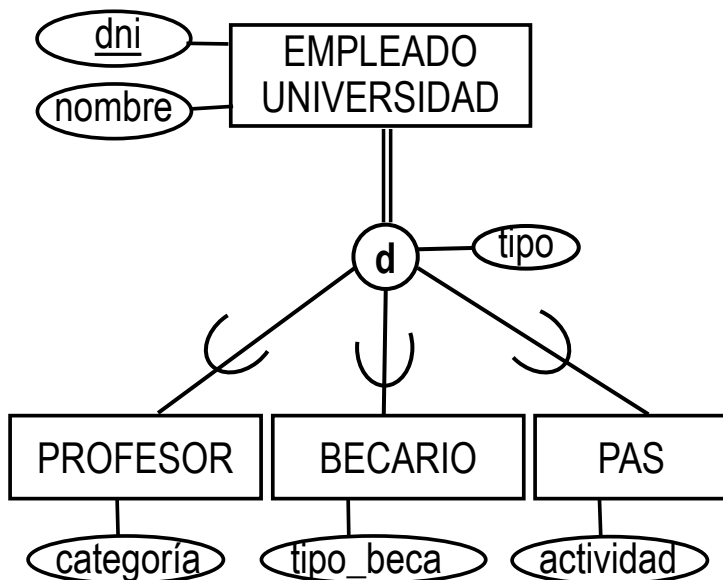
COMPROBAR_QUE (

NO_EXISTE (una tupla de ARTICULO
tal que el valor de "código"
ESTÉ_ENTRE (valores de "código"
en LIBRO)));

1.3. Traducción de Jerarquías

23

Ejemplo 2 de transformación de una **jerarquía disjunta y total**



PROFESOR(dni, nombre, categoría)

Clave primaria: dni

BECARIO(dni, nombre, tipo_beca)

Clave primaria: dni

PAS(dni, nombre, actividad)

Clave primaria: dni

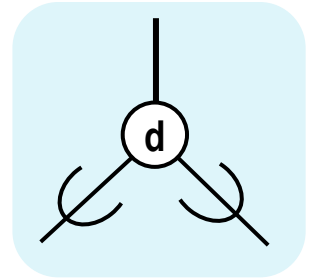
ASERTO RI_profesor_xor_becario_xor_pas
COMPROBAR_QUE (
 NO_EXISTE ((valores de “dni”
 de PROFESOR)
 INTERSECCION
 (valores “dni” de BECARIO)
 INTERSECCION
 (valores “dni” de PAS)));

1.3. Traducción de Jerarquías

24

□ Jerarquía disjunta y parcial

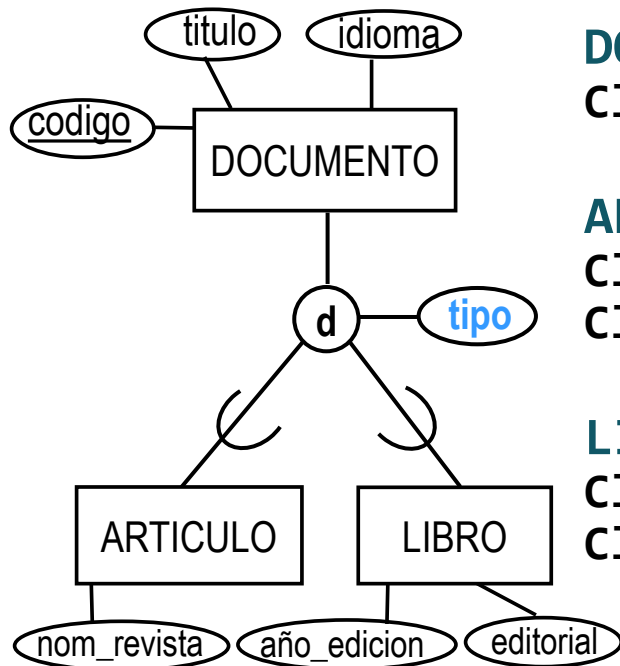
- Da lugar a varias relaciones: **una relación para el supertipo y una relación para cada subtipo**
- Las relaciones correspondientes a los subtipos...
 - Contienen una **clave ajena** que referencia a la clave primaria de la relación correspondiente al supertipo
 - La **clave primaria** de cada relación (de subtipo) es dicha clave ajena
- Hay que definir una **restricción de integridad general** (aserto) para garantizar la disyunción



1.3. Traducción de Jerarquías

25

Ejemplo de traducción de una jerarquía disjunta y parcial



DOCUMENTO (*codigo*, titulo, idioma)
Clave primaria: codigo

ARTICULO (*codigo*, nom_revista)
Clave primaria: codigo
Clave ajena: codigo Referencia_a DOCUMENTO(codigo)

LIBRO (*codigo*, año_edicion, editorial)
Clave primaria: codigo
Clave ajena: codigo Referencia_a DOCUMENTO(codigo)

ASERTO RI_articulo_xor_libro

COMPROBAR_QUE (
 NO_EXISTE (una tupla de ARTICULO
 tal que el valor de “codigo”
 ESTÉ_ENTRE (valores de “codigo”
 en LIBRO)));

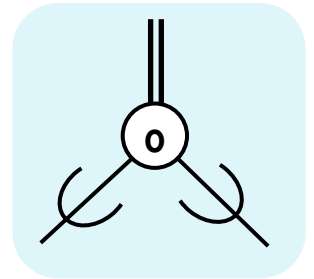
❗ El atributo **discriminante** **no** aparece en ninguna de las relaciones resultado de la traducción

1.3. Traducción de Jerarquías

26

□ Jerarquía solapada y total

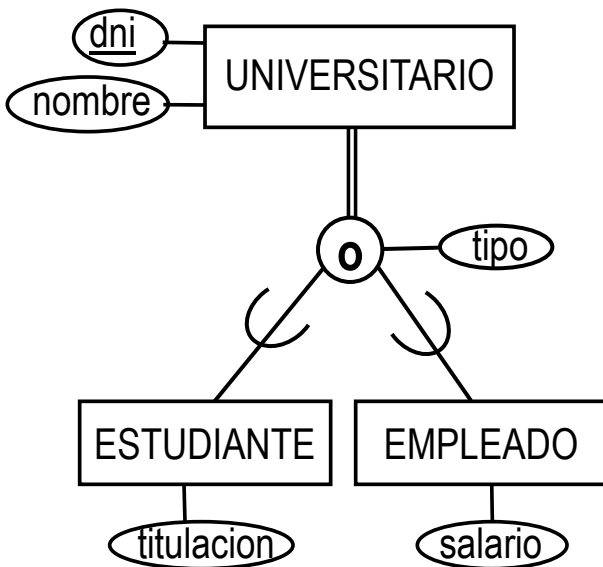
- ▣ Da lugar a **una única relación**, que corresponde al **supertipo**
 - Debe incluir uno o más atributos discriminantes para distinguir el tipo concreto de cada tupla
 - Los atributos de los subtipos han de admitir el nulo
- ▣ Hay que definir **restricciones de comprobación** para asegurar la corrección de los datos para cada tupla según su tipo



1.3. Traducción de Jerarquías

27

Ejemplo de traducción de **jerarquía solapada total** OPCIÓN 1



UNIVERSITARIO(dni, nombre, **tipo**,
titulacion, salario)

Admiten NULL: titulacion, salario

Clave primaria: dni

Comprobar:

- tipo IN ('Estudiante', 'Empleado', 'Est+Emp')
- ((tipo = 'Estudiante'
AND titulacion IS NOT NULL)
OR (tipo = 'Empleado'
AND salario IS NOT NULL)
OR (tipo = 'Est+Emp'
AND titulacion IS NOT NULL
AND salario IS NOT NULL))

❶ Uso de **un atributo discriminante**, con un valor para cada subtipo ('Estudiante' o 'Empleado') y un valor extra para el solapamiento ('Est+Emp')

1.3. Traducción de Jerarquías

28

Ejemplo de traducción de **jerarquía solapada total** OPCIÓN 2

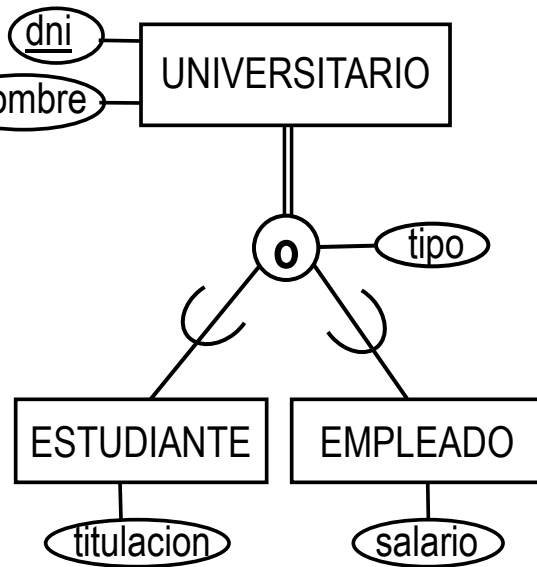
UNIVERSITARIO(dni, nombre, *estudia*, *trabaja*,
titulacion, salario)

Admiten NULL: titulacion, salario

Clave primaria: dni

Comprobar:

- estudia IN ('S', 'N')
- trabaja IN ('S', 'N')
- (estudia='S' AND titulacion IS NOT NULL)
OR (estudia='N' AND titulacion IS NULL)
- (trabaja='S' AND salario IS NOT NULL)
OR (trabaja='N' AND salario IS NULL)



❶ Uso de **dos atributos discriminantes** (uno por cada subtipo), con un valor SÍ/NO

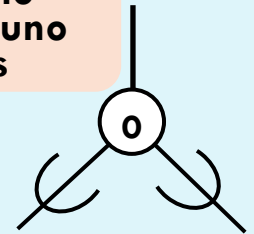
1.3. Traducción de Jerarquías

29

□ Jerarquía solapada y parcial

- Da lugar a 2 relaciones: **una relación para el supertipo y una relación para todos los subtipos**

Contiene todas las tuplas, incluidas las que **no pertenecen a ninguno** de los subtipos



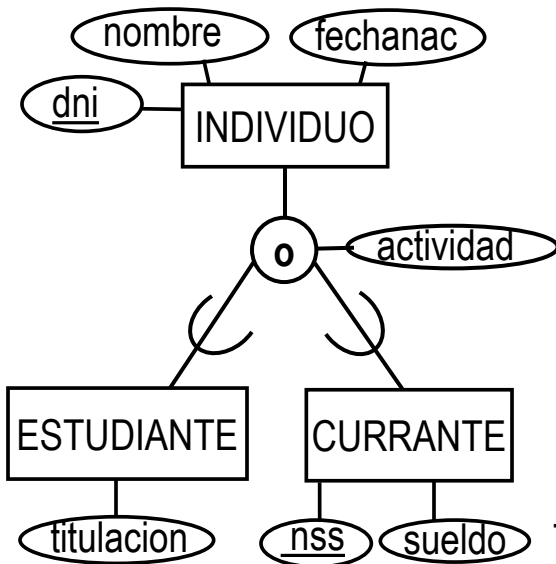
Contiene las tuplas que **pertenecen a uno o varios subtipos**

- La relación que agrupa a los subtipos...
 - Incluye una **clave ajena** que referencia a la relación correspondiente al supertipo
 - La **clave primaria** es dicha clave ajena
 - Incluye **discriminante(s)** para distinguir el tipo de cada tupla
 - Debe contener **restricciones de comprobación** que garanticen la corrección de los datos
 - Los atributos de los subtipos han de admitir el nulo

1.3. Traducción de Jerarquías

30

Ejemplo de traducción de jerarquía solapada parcial OPCIÓN 1



❶ Uso de **un atributo discriminante**, con un valor para cada tipo y un valor extra para el solapamiento

INDIVIDUO(dni, nombre, fechanac)

Clave primaria: dni

INDIVIDUO_ACTIVO(dni, **actividad**,
titulacion, nss, sueldo)

Admiten NULL: titulacion, nss, sueldo

Clave primaria: **dni**

Clave alternativa: **nss**

Clave ajena: dni Referencia_a INDIVIDUO(dni)

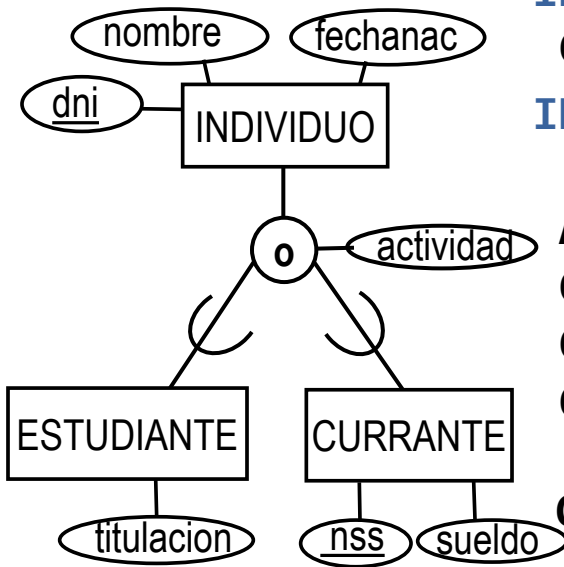
Comprobar:

- actividad IN ('Estudia', 'Trabaja', 'Est+Trab')
- ((actividad='Estudia' AND titulacion IS NOT NULL)
OR (actividad='Trabaja' AND nss IS NOT NULL
AND sueldo IS NOT NULL)
OR (actividad='Est+Trab'
AND titulacion IS NOT NULL
AND nss IS NOT NULL AND sueldo IS NOT NULL))

1.3. Traducción de Jerarquías

31

Ejemplo de traducción de jerarquía solapada parcial OPCIÓN 2



INDIVIDUO(dni, nombre, fechanac)

Clave primaria: dni

INDIVIDUO_ACTIVO(dni, *estudia*, *curra*,
titulacion, nss, sueldo)

Admiten NULL: titulacion, nss, sueldo

Clave primaria: **dni**

Clave alternativa: **nss**

Clave ajena: dni Referencia_a INDIVIDUO(dni)

Comprobar:

- estudia IN ('S', 'N') AND curra IN ('S', 'N')
- ((estudia='S' AND titulacion IS NOT NULL)
OR (estudia='N' AND titulacion IS NULL))
- ((curra='S' AND nss IS NOT NULL
AND sueldo IS NOT NULL)
OR (curra='N' AND nss IS NULL AND sueldo IS NULL))

❶ Uso de **varios**
atributos
discriminantes:
uno por subtipo,
con un valor SI/NO

1.4. Tipo de **relación 1:N**


32

- Se traduce a una **clave ajena** que referencia desde una relación (tabla) a la otra

¿En qué relación se introduce la clave ajena?


- Hay que identificar los tipos de entidad 'padre' e 'hijo'
 - ▣ Tipo de entidad '**padre**': el que **participa muchas veces** en el tipo de relación
 - ▣ Tipo de entidad '**hijo**': el que participa **sólo una vez**



- En PUBLICADO_POR, cada LIBRO participa sólo una vez, pues es publicado por una y sólo una editorial, mientras que EDITORIAL participa muchas veces, pues puede haber editado muchos libros
- **Padre: EDITORIAL; Hijo: LIBRO** 

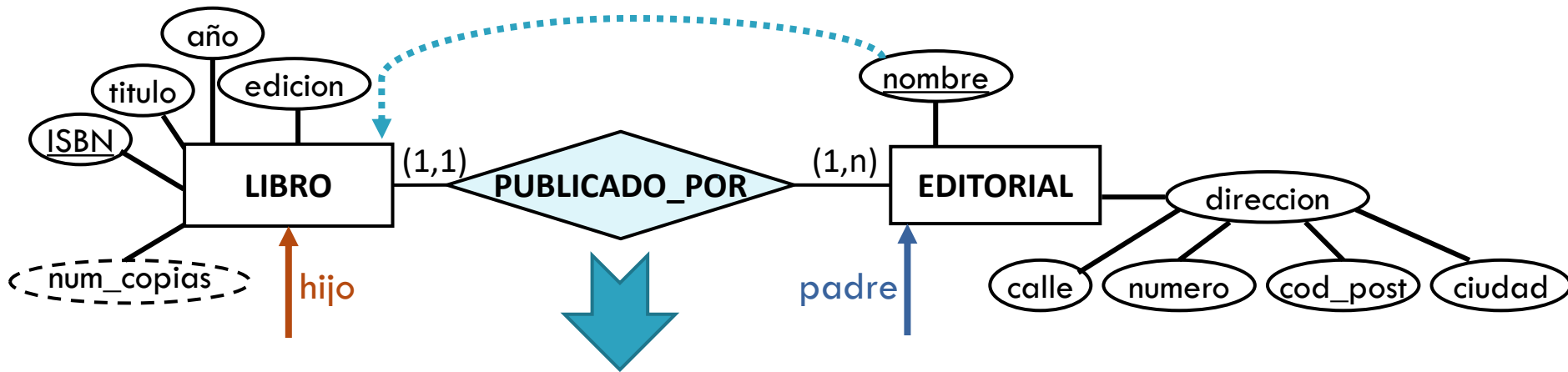
1.4. Tipo de relación 1:N

33

- La clave ajena servirá para enlazar cada tupla (fila) ‘hijo’ con su correspondiente tupla ‘padre’
 - ▣ Cada LIBRO indicará cuál es su EDITORIAL
- ▶ Hay que **añadir al tipo relación ‘hijo’ una copia de la clave primaria del tipo relación ‘padre’** 
- ▣ Es la “propagación de la clave”: el *padre* transmite su clave al *hijo*, donde es una clave ajena que referencia al padre
- ▣ La clave ajena tendrá **uno o varios atributos** según tenga la clave primaria a la que referencia
 - Si sólo es un atributo puede llamarse como la relación a la que referencia (recomendado)
 - Pero cada atributo componente de la clave ajena puede tener el mismo nombre que el atributo del que es copia, o puede llamarse de otra forma

1.4. Tipo de relación 1:N

34



EDITORIAL(nombre,calle,numero,cod_post,ciudad)

Admiten NULL: Ninguno

Clave primaria: nombre

LIBRO(ISBN,titulo,año,edicion,num_copias,**editorial**)

Admiten NULL: Ninguno

Clave primaria: ISBN

Clave ajena: editorial Referencia_a EDITORIAL(nombre)

Derivado: num_copias (nota:falta fórmula de cálculo)

1.4. Tipo de relación 1:N

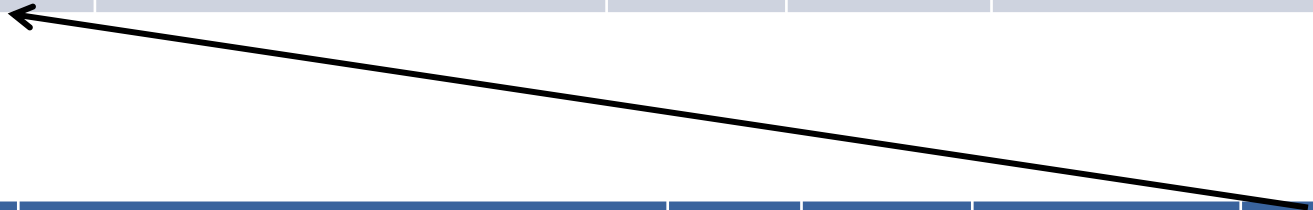
35

EDITORIAL

<u>nombre</u>	calle	numero	cod_post	ciudad
Espasa	Josefa Valcárcel	42	28027	Madrid
Santillana	Avenida de los Artesanos	6	28760	Tres Cantos, Madrid
Edebé	Paseo San Juan Bosco	62	08017	Barcelona

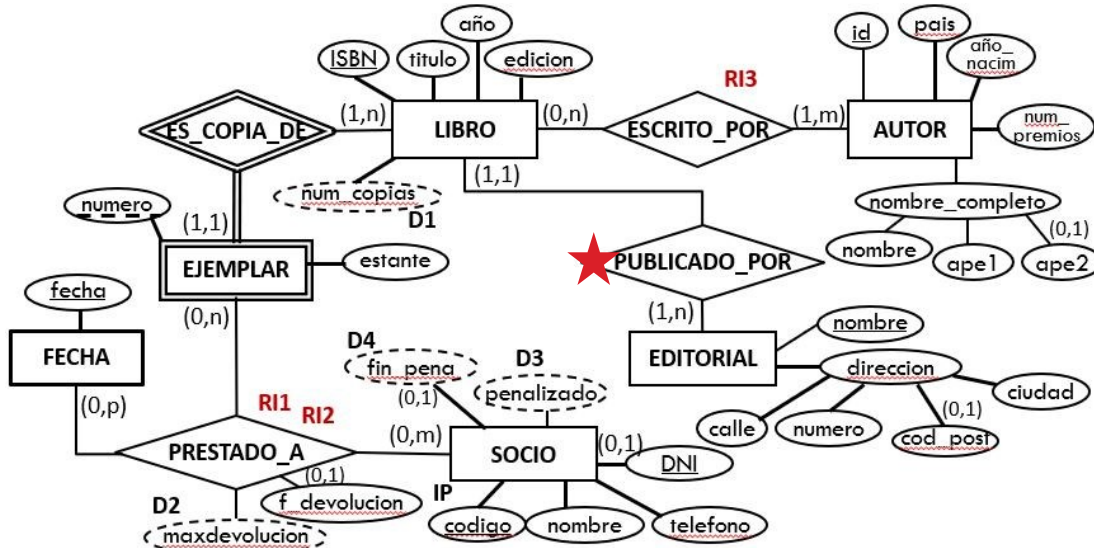
LIBRO

<u>ISBN</u>	título	año	edicion	num_copias	editorial
9788408217251	Un científico en el supermercado	2019	1	14	Espasa
9788491223542	Malamandra	2019	5	5	Santillana
9788468315775	Mentira	2015	4	12	Edebé
9788467009101	Divina comedia	2010	50	25	Espasa
9788468319612	La nueva vida del señor Rutin	2014	23	16	Edebé



Ejemplo. Traducción de tipos de relación 1:N

36



EDITORIAL(nombre,calle, numero,cod_post,ciudad)
 Admiten NULL: **cod_post**
 Clave primaria: **nombre**

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
 Admiten NULL: **DNI,fin_pena**
 Clave primaria: **codigo**
 Clave alternativa: **DNI**

AUTOR(id,nombre,pais,año_nacim,num_premios,nombre,ape1,ape2)
 Admiten NULL: **ape2**
 Clave primaria: **id**

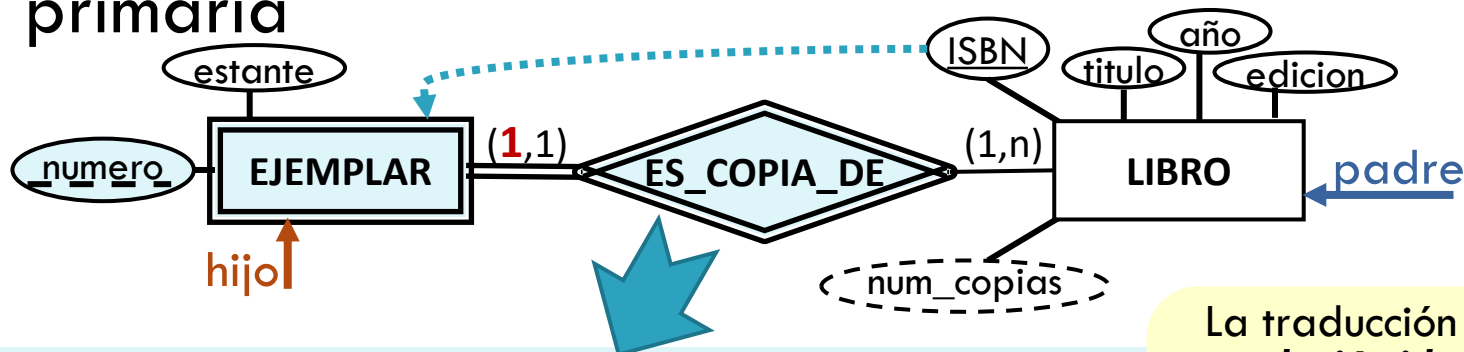
LIBRO(ISBN,título,año,edicion,num_copias, **editorial**)
 Admiten NULL: Ninguno
 Clave primaria: **ISBN**
Clave ajena: editorial REFERENCIA_A EDITORIAL(nombre)
 Derivado: num_copias (nota:falta fórmula de cálculo)

EJEMPLAR(numero, estante)
 Admiten NULL: Ninguno
 Clave primaria: <Pendiente>

1.4. Tipo de relación 1:N identificador

37

- Si el tipo de entidad 'hijo' es **débil**, se traduce igual y además, la clave ajena forma parte de la clave primaria



La traducción de **tipo de relación identificador** permite **ipor fin! definir la clave primaria de la relación que corresponde al tipo de entidad débil**

EJEMLAR (numero, estante, **ISBN**)

Admiten NULL: Ninguno

Clave primaria: (**ISBN**, numero)

Clave ajena: **ISBN Referencia_a LIBRO(ISBN)**

LIBRO(ISBN, titulo, año, edicion, num_copias, editorial)

Admiten NULL: Ninguno

Clave primaria: ISBN

Clave ajena: editorial Referencia_a EDITORIAL(nombre)

Derivado: num_copias [para cada tupla l contar tuplas e de EJEMLAR

cuyo valor de e.ISBN es igual a l.ISBN]

Podemos redactar la fórmula de cálculo

1.4. Tipo de relación 1:N identificador

ISBN	titulo	año	edicion	num_copias	editorial
9788408217251	Un científico en el supermercado	2019	1	14	Espasa
9788491223542	Malamandra	2019	5	5	Santillana
9788468315775	Mentira	2015	4	12	Edebé
9788467009101	Divina comedia	2010	50	25	Espasa
9788468319612	La nueva vida del señor Rutilio				Edebé

LIBRO

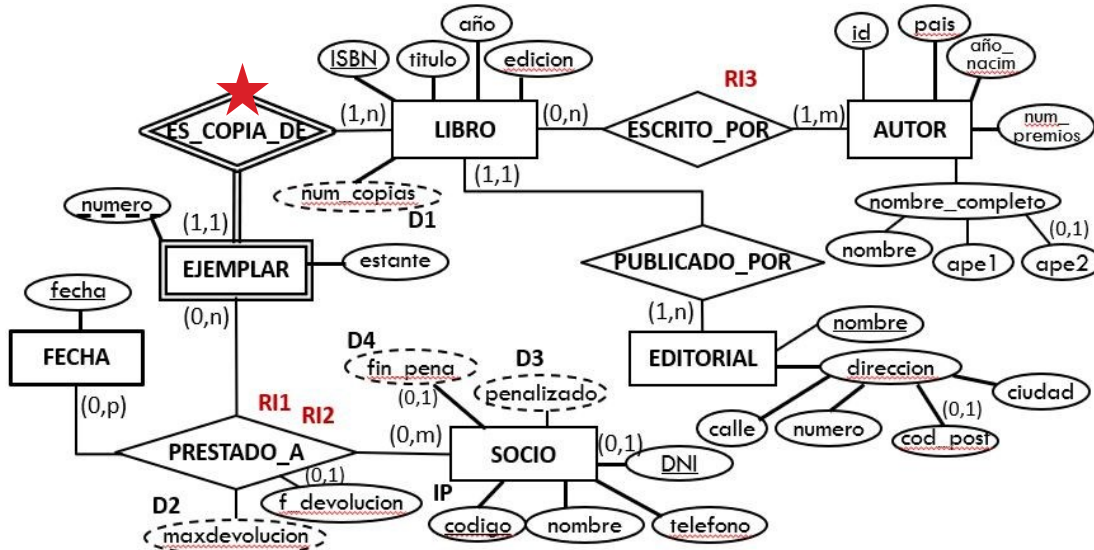
EJEMPLAR

Un ejemplar solo se distingue de otro ejemplar conociendo tanto el ISBN de su libro como su número de ejemplar

ISBN	numero	estante
9788408217251	1	H4
9788408217251	2	H4
9788408217251	3	H5
9788467009101	1	M2
9788467009101	2	M2
9788468319612	1	S1
9788491223542	1	S2
9788491223542	2	A3
9788468315775	1	S7

Ejemplo. Traducción de tipos de relación 1:N identificador

39



EDITORIAL(nombre,calle, numero,cod_post,ciudad)
Admiten NULL: **cod_post**
Clave primaria: **nombre**

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
Admiten NULL: **DNI,fin_pena**
Clave primaria: **codigo**
Clave alternativa: **DNI**

AUTOR(id, nombre, pais, año_nacim,num_premios,nombre,ape1,ape2)
Admiten NULL: **ape2**
Clave primaria: **id**

LIBRO(ISBN, título, año, edición, num_copias, editorial)
Clave primaria: **ISBN**
Clave ajena: **editorial** Referencia_a **EDITORIAL**(nombre)

Derivado: **num_copias** [para cada tupla 1 contar tuplas e de EJEMPLAR cuyo...]

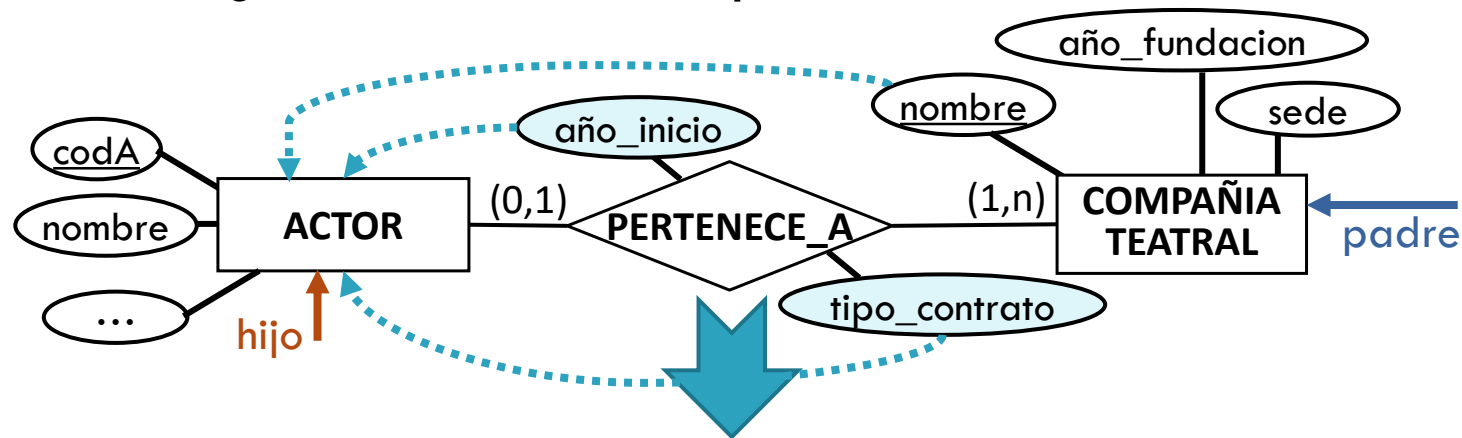
EJEMPLAR(numero, estante, ISBN)
Clave primaria: **(ISBN, numero)**

Clave ajena: **ISBN** Referencia_a **LIBRO**(ISBN)

1.4. Tipo de relación 1:N con atributos

40

- Si el tipo de relación contiene atributos, se añaden como atributos en el tipo de relación 'hijo'
 - ▣ “Persiguen” a la clave ajena



COMPAÑIA(nombre, año_fundacion, sede)

Clave primaria: nombre

ACTOR(codA, nombre, ..., *compañia*, **año_inicio_compañia**, **tipo_contrato**)

Clave primaria: codA

Clave ajena: *compañia* Referencia_a COMPAÑIA(nombre)

1.4. Tipo de relación 1:N con atributos

41

COMPAÑIA

<u>nombre</u>	año_fundacion	sede
Galeras	1972	Almería
Café de las Artes	1965	Madrid
Quimera	2012	Valencia

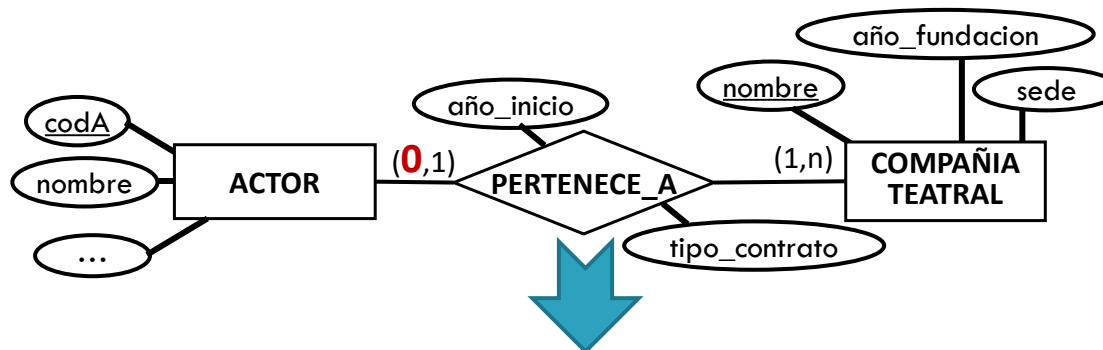
ACTOR

<u>codA</u>	nombre	... compañía	año_inicio_compañia	tipo_contrato
a01	Susana Sosa	Quimera	2013	temporada
a18	Vilma Valles	Café de las Artes	2000	fijo
a07	Manuel Mero	Quimera	2012	fijo
a43	Borja Bocas	Galeras	1990	temporada

1.4. Tipo de relación **1:N**, cardinalidad

42

- La **cardinalidad mínima 0** del tipo de **entidad 'hijo'** indica que la **clave ajena SÍ admite NULO**
 - ▣ Y con ella, el resto de **atributos del tipo de relación**



COMPAÑIA(nombre, año_fundacion, sede)

Admiten NULL: Ninguno

Clave primaria: nombre

ACTOR(codA, nombre, ..., compañía, año_inicio_compañía, tipo_contrato)

Admiten NULL: **compañía, año_inicio_compañía, tipo_contrato**

Clave primaria: codA

Clave ajena: compañía Referencia_a COMPAÑIA(nombre)

1.4. Tipo de relación 1:N, cardinalidad

43

COMPañIA

nombre	año_fundacion	sede
Galeras	1972	Almería
Café de las Artes	1965	Madrid
Quimera	2012	Valencia



Hay que poder representar a actores/actrices que NO pertenecen a ninguna compañía teatral

ACTOR

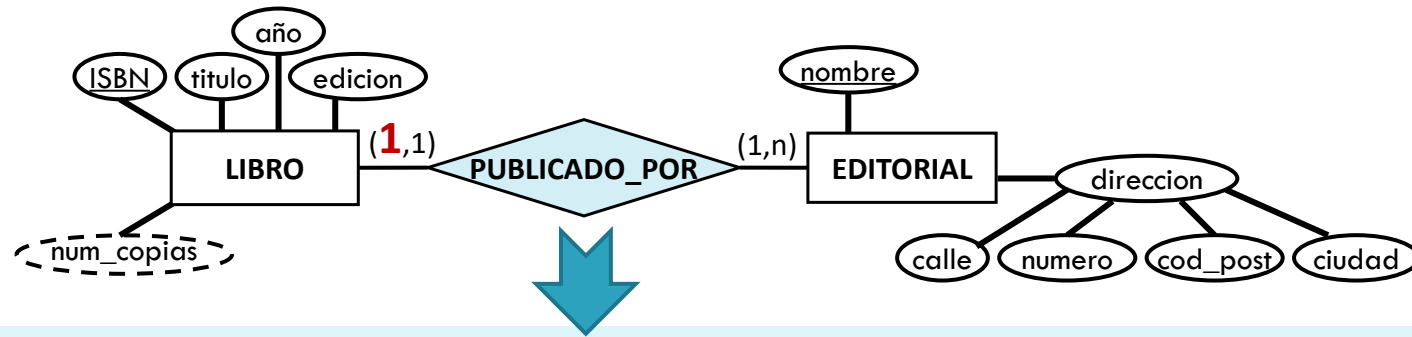
codA	nombre	...	compañía	año_inicio_compañía	tipo_contrato
a01	Susana Sosa		Quimera	2013	temporada
a18	Vilma Valles		Café de las Artes	2000	fijo
a07	Manuel Mero		Quimera	2012	fijo
a43	Borja Bocas		Galeras	1990	temporada
a02	Clotilde Calle		NULL	NULL	NULL



1.4. Tipo de relación 1:N, cardinalidad

44

- La **cardinalidad mínima 1** del tipo de entidad ‘hijo’ indica que la **clave ajena NO admite NULO**



EDITORIAL(nombre,calle,numero,cod_post,ciudad)

Admiten NULL: Ninguno

Clave primaria: nombre

LIBRO(ISBN,titulo,año,edicion,num_copias,editorial)

Admiten NULL: **Ninguno**

Clave primaria: ISBN

Clave ajena: editorial Referencia_a EDITORIAL(nombre)

Derivado: num_copias [para cada ISBN, contar instancias de EJEMPLAR cuyo valor de “ISBN” es igual a dicho ISBN]

1.4. Tipo de relación 1:N, cardinalidad

45

EDITORIAL

<u>nombre</u>	calle	numero	cod_post	ciudad
Espasa	Josefa Valcárcel	42	28027	Madrid
Santillana	Avenida de los Artesanos	6	28760	Tres Cantos, M
Edebé	Paseo San Juan Bosco	62	08017	Barcelona

Si un libro contuviera NULL en la clave ajena (atributo 'editorial') significaría que no tiene editorial, así que se incumpliría la cardinalidad mínima 1: TODO libro debe tener una

LIBRO

<u>ISBN</u>	título	año	edicion	num_copias	editorial
9788408217251	Un científico en el supermercado	2019	1	14	Espasa
9788491223542	Malamandra	2019	5	5	Santillana
9788468315775	Mentira	2015	4	12	NULL
9788467009101	Divina comedia	2010	50	25	Espasa
9788468319612	La nueva vida del señor Rutin	2014	23	16	Edebé

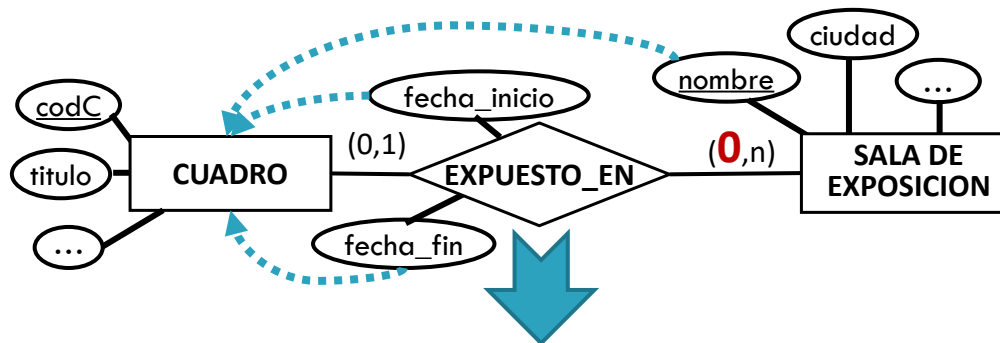


KO!

1.4. Tipo de relación 1:N, cardinalidad

46

- La **cardinalidad mínima 0** del tipo de **entidad 'padre'** indica que puede haber tuplas (filas) en la relación 'padre' no referenciadas mediante la clave ajena



SALA_EXPOSICION(nombre,ciudad,...)

Admiten NULL: Ninguno

Clave primaria: nombre

Nada nuevo en la definición de las relaciones tras la traducción

CUADRO(codC,titulo,...,sala,fecha_inicio_expo,fecha_fin_expo)

Admiten NULL: sala, fecha_inicio_expo, fecha_fin_expo

Clave primaria: codC

Clave ajena: sala Referencia_a SALA_EXPOSICION(nombre)

1.4. Tipo de relación 1:N, cardinalidad

47

SALA_EXPOSICION

<u>nombre</u>	ciudad	...
Babel	Murcia	
Estampa	Madrid	
La Aurora	Murcia	



Hay salas de exposición no referenciadas por ningún cuadro ('Estampa' por ejemplo)
OK **cardinalidad mínima 0** de SALA_EXPOSICION

❗ Recordamos: CUADRO tiene cardinalidad mínima 0 en 'EXPUESTO EN', por lo que la clave ajena y atributos 'fecha_inicio_expo' y 'fecha_fin_expo' **admiten nulos**

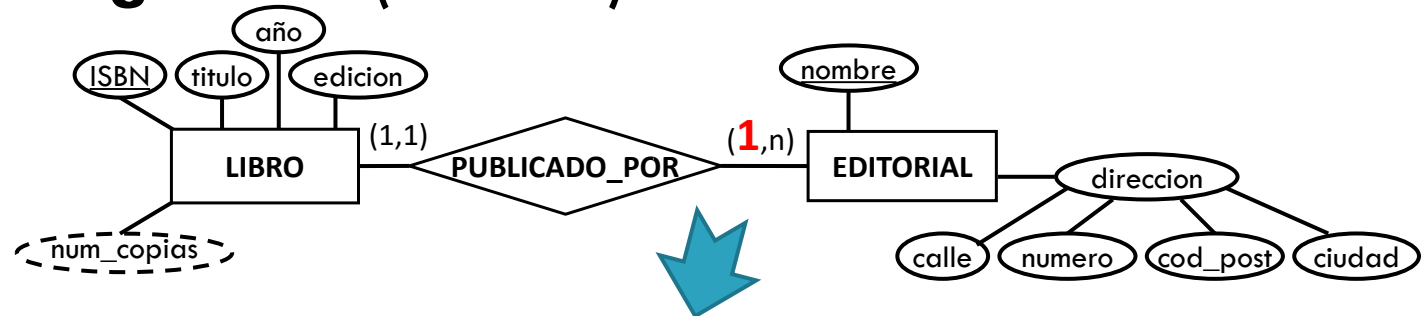
CUADRO

<u>codC</u>	titulo	... sala	fecha_inicio_expo	fecha_fin_expo
c01	Mao Zoup	Babel	20/10/2019	30/10/2019
c30	Atardecer otoñal	NULL	NULL	NULL
c17	Canned Love	Babel	10/11/2019	20/11/2019
c23	La alegría de vivir	La Aurora	05/11/2019	25/11/2019
c04	Paisaje veraniego	NULL	NULL	NULL
c08	Nocturno Festivo	La Aurora	15/10/2019	31/10/2019

1.4. Tipo de relación 1:N, cardinalidad

48

- La **cardinalidad mínima 1** del tipo de **entidad ‘padre’** debe representarse mediante una **restricción de integridad general (Aserio)**



Esa cardinalidad mínima representa que

“Toda EDITORIAL ha publicado al menos un LIBRO”.

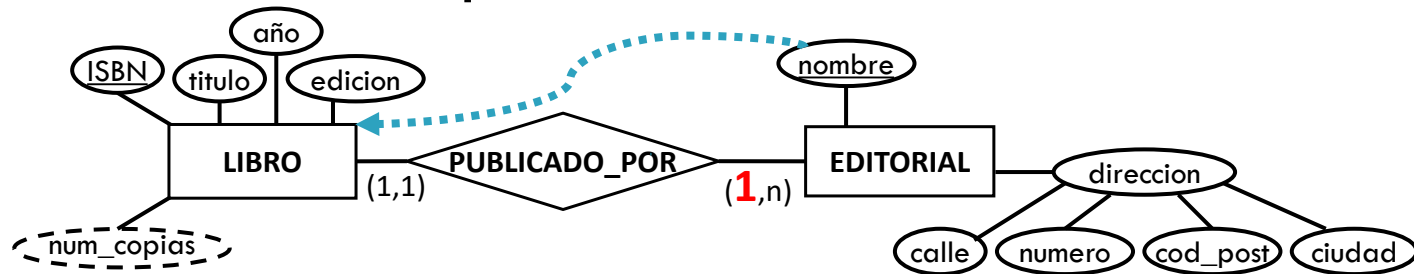
► Hay que asegurar que no existe una EDITORIAL que no esté vinculada con al menos un LIBRO vía PUBLICADO_POR.

Pero este texto no sirve para el Esquema Lógico, porque PUBLICADO_POR “ya no existe”. Sólo tenemos **relaciones** (tablas) y **claves ajenas** (vínculos)...

1.4. Tipo de relación 1:N, cardinalidad

49

- (cont.) Traducción de la cardinalidad mínima 1 del tipo de entidad 'padre' a un aserto



EDITORIAL(nombre, calle, numero, cod_post, ciudad)

LIBRO(ISBN, título, año, edición, num_copias, *editorial*)

Clave ajena: editorial Referencia_a EDITORIAL(nombre)

En el Esquema Lógico tenemos las relaciones EDITORIAL y LIBRO:

- No debe existir una tupla de EDITORIAL que no esté referenciada desde LIBRO.
- Es decir, hay que comprobar que no existe una tupla en EDITORIAL donde el valor de “nombre” (PK) no sea uno de los valores de “editorial” (FK) de las tuplas de LIBRO.

1.4. Tipo de relación 1:N, cardinalidad

50

EDITORIAL

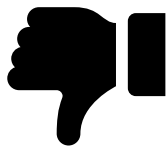
<u>nombre</u>	calle	numero	cod_post	ciudad
Espasa	Josefa Valcárcel	42	28027	Madrid
Santillana	Avenida de los Artesanos	6	28760	Tres Cantos, Madrid
Edebé	Paseo San Juan Bosco	62	08017	Barcelona
Fandom Books	C/ Valentín Beato	21	28037	Madrid

Esto es lo que NO debe ocurrir: no hay ningún libro cuya editorial sea "Fandom Books"


KO!

LIBRO

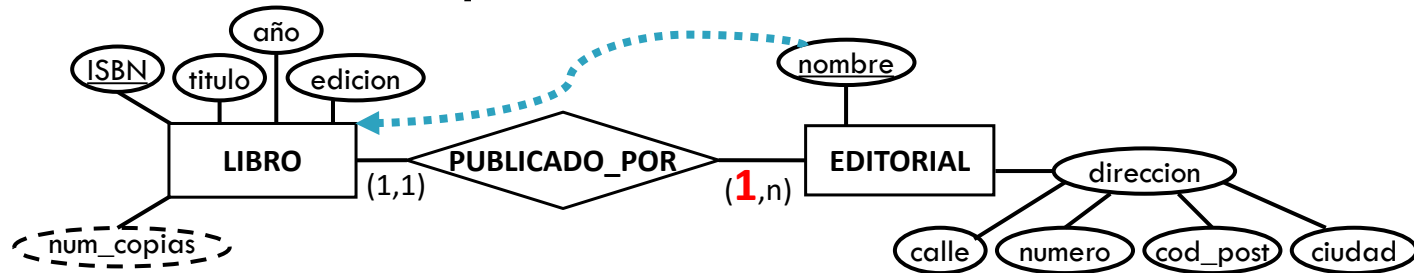
<u>ISBN</u>	título	año	edición	num_copias	editorial
9788408217251	Un científico en el supermercado	2019	1	14	Espasa
9788491223542	Malamandra	2019	5	5	Santillana
9788468315775	Mentira	2015	4	12	Edebé
9788467009101	Divina comedia	2010	50	25	Espasa
9788468319612	La nueva vida del señor Rutin	2014	23	16	Edebé



1.4. Tipo de relación 1:N, cardinalidad

51

- (cont.) Traducción de la cardinalidad mínima 1 del tipo de entidad 'padre' a un aserto



EDITORIAL(nombre, calle, numero, cod_post, ciudad)

LIBRO(ISBN, titulo, año, edición, num_copias, *editorial*)

Clave ajena: editorial Referencia_a EDITORIAL(nombre)

La redacción más adecuada es esta:

► **ASERTO RI_editorial_publica_libros**

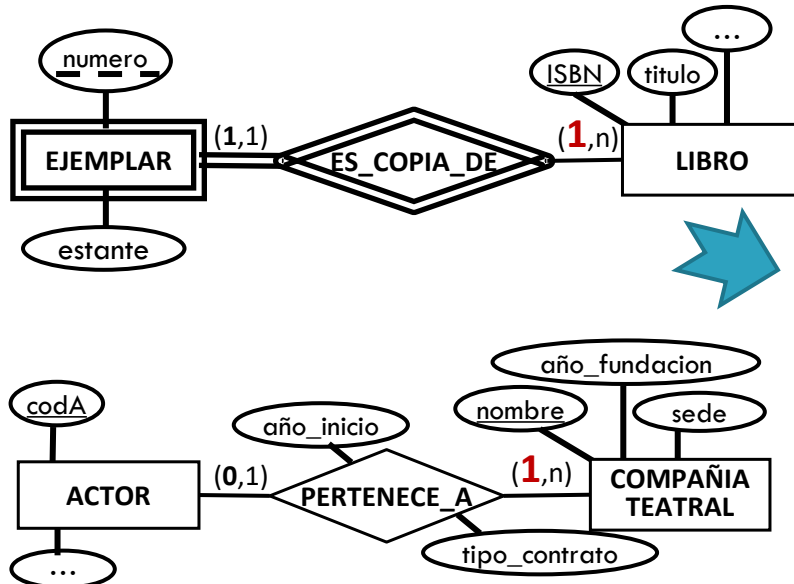
COMPROBAR_QUE (NO_EXISTE (una tupla en EDITORIAL

donde el valor de "nombre" NO_ESTÉ_ENTRE
(valores de "editorial" en LIBRO)));



1.4. Tipo de relación 1:N, cardinalidad

52



EJEMPLAR (numero, estante, **ISBN**)
 Admiten NULL: Ninguno
 Clave primaria: (**ISBN**, numero)
 Clave ajena: ISBN Referencia_a LIBRO(**ISBN**)

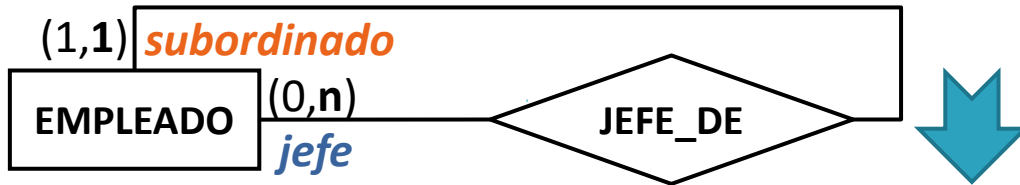
ASERTO RI_libro_tiene_copias
COMPROBAR_QUE (
 NO_EXISTE (una tupla en LIBRO
 donde el valor de “ISBN”
 NO_ESTÉ_ENTRE (valores de
 “ISBN” en EJEMPLAR)));

ACTOR(codA,nombre,..., **compañía**, año_inicio_compañía, tipo_contrato)
 Admiten NULL: compañía, año_inicio_compañía, tipo_contrato
 Clave primaria: codA
 Clave ajena: compañía Referencia_a COMPAÑIA(nombre)

ASERTO RI_compañía_tiene_actores
COMPROBAR_QUE (**NO_EXISTE** (una tupla en COMPAÑIA
 donde el valor de “nombre” **NO_ESTÉ_ENTRE**
 (valores de “compañía” en ACTOR)));

1.4. Tipo de relación 1:N recursiva

53



EMPLEADO(codE, nombre, dni, direccion,..., **jefe**)

Admiten NULL: Ninguno

Clave primaria: codE

Clave alternativa: dni

Clave ajena: jefe Referencia_a EMPLEADO(codE)

Comprobar: **codE<>jefe**

1 Deducir los roles “**padre**” e “**hijo**” según cardinalidades máximas (como siempre):
Padre: jefe (lado “muchos”)
Hijo: subordinado (lado “1”)

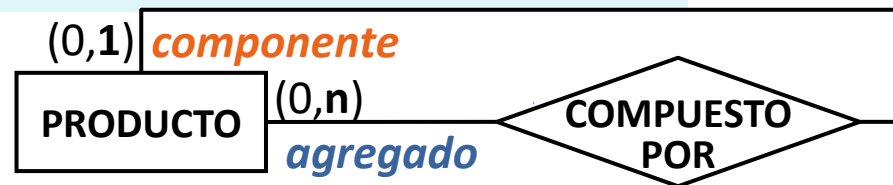
2 La **relación** (tabla) **ya definida** para el tipo de entidad **SIEMPRE** debe corresponder al “**hijo**”

EMPLEADO = subordinado

PRODUCTO = componente

3

... y contiene una **clave ajena** que es una **referencia al “padre”**, de ahí que la FK tome el nombre del **rol** con cardinalidad “muchos”



PRODUCTO(codP, nombre, descripcion,..., **agregado**)

Admiten NULL: agregado

Clave primaria: codP

Clave ajena: agregado Referencia_a PRODUCTO(codP)

Comprobar: **codP<>agregado**

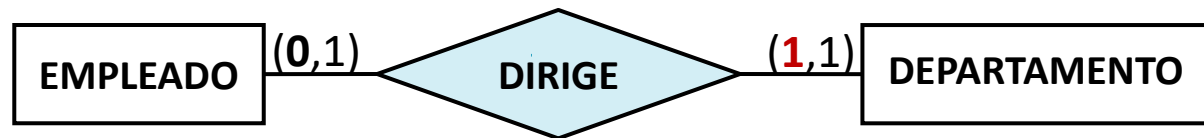
46-56 revisar

1.5. Tipo de **relación 1:1**

54

- Se traduce a una **clave ajena**
- ¿En qué relación se incluye la clave ajena? Se decide con base en la **participación** de cada tipo de entidad
 - ▣ Cardinalidad mínima 0 = participación **opcional** o parcial
 - ▣ Cardinalidad mínima 1 = participación **obligatoria** o total
- Casos que nos podemos encontrar:

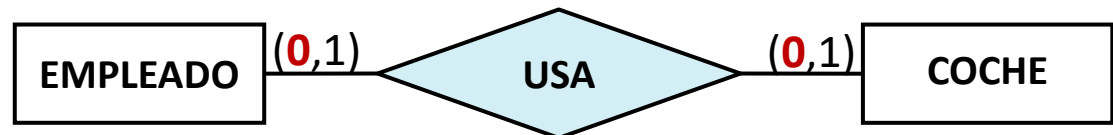
a. Obligatoria en
un lado



b. Obligatoria en
ambos lados



c. Opcional en
ambos lados

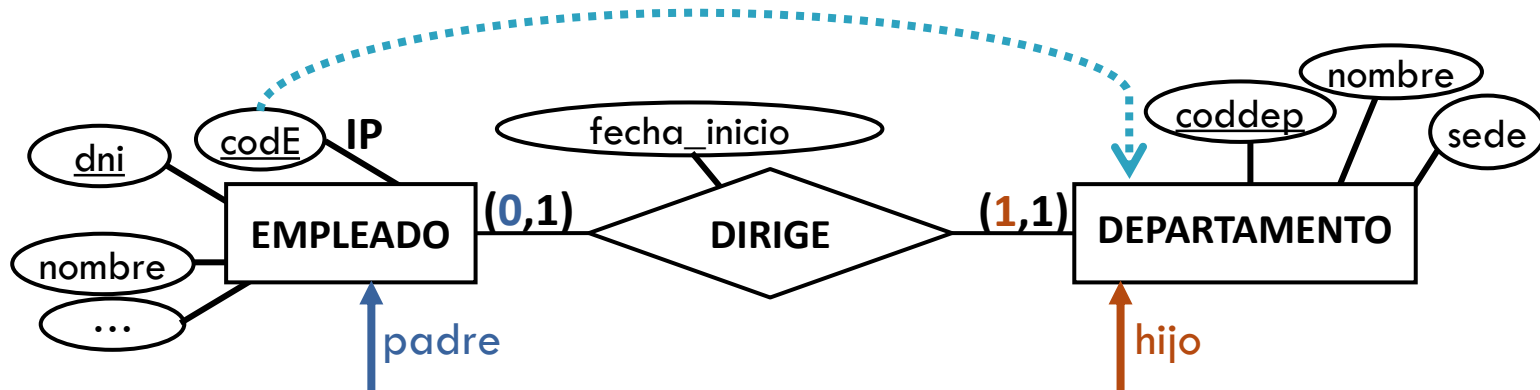


1.5. Tipo de relación 1:1

55

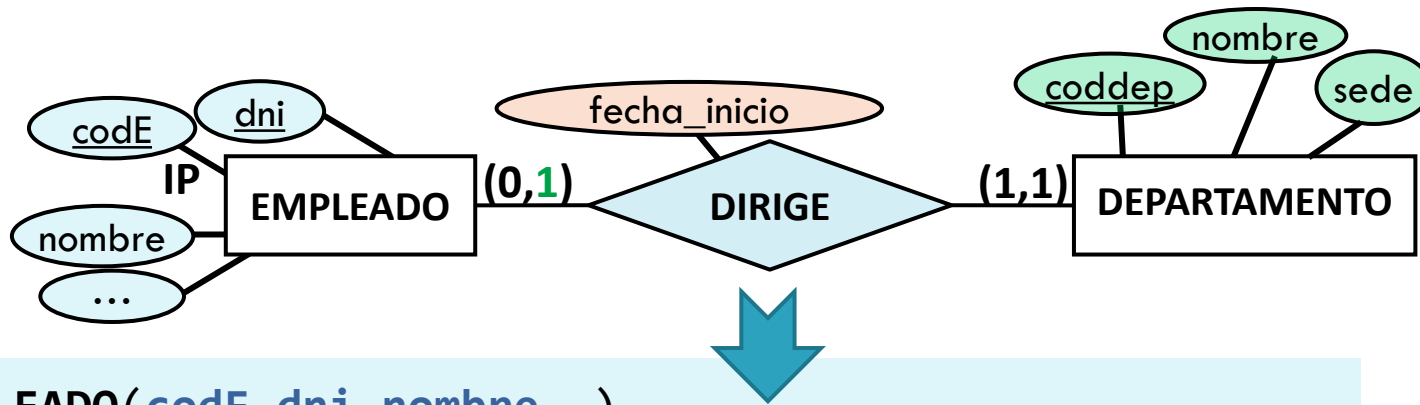
a. Participación **obligatoria** en un lado

- ▣ Tipo de entidad '**padre**': la de **participación opcional**
 - Cardinalidad mínima 0
- ▣ Tipo de entidad '**hijo**': la de **participación obligatoria** o total
 - Cardinalidad mínima 1
- ▣ Traducir a una **clave ajena** en la relación correspondiente al tipo de entidad **hijo**
 - Igual que las relaciones 1:N



1.5. Tipo de relación 1:1

56



EMPLEADO(**codE**, dni, nombre, ...)

Admiten NULL: Ninguno

Clave primaria: codE

Clave alternativa: dni

DEPARTAMENTO(**coddep**, nombre, ..., *director*, fecha_inicio)

Admiten NULL: Ninguno

Clave primaria: coddep

Clave alternativa: **director** ★

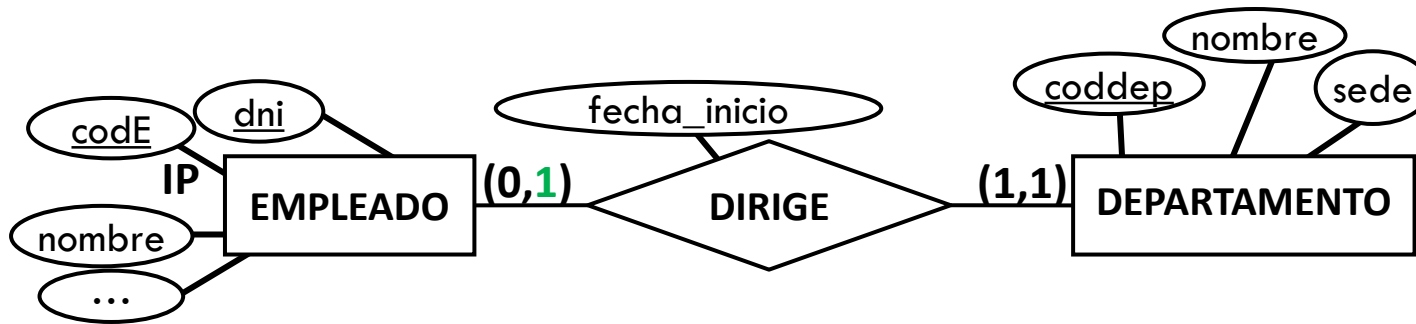
Clave ajena: director Referencia_a EMPLEADO(codE)

Diferencia con la traducción de las 1:N

❗ La **cardinalidad máxima 1** de la relación “padre” hace que la clave ajena en la relación “hijo” también sea Clave Alternativa... Veámoslo...

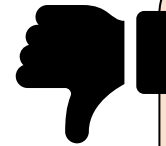
1.5. Tipo de relación 1:1

57



EMPLEADO

<u>codE</u>	nombre	...
e1	Laura López	
e2	María Martínez	
e3	Abel Ayala	
e4	Pedro Palao	
e5	Simona Sánchez	



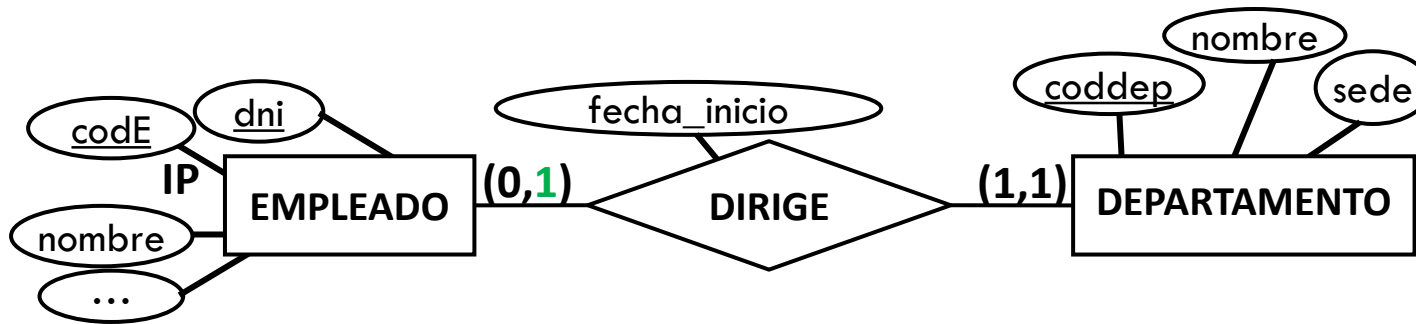
Si el atributo “director”
Sí repitiera valores,
significaría que el mismo
empleado puede dirigir
varios departamentos

DEPARTAMENTO

<u>coddep</u>	nombre	director	fecha_inicio	...
d1	Informática	e2	23/02/2015	
d2	Sistemas	e2	15/04/2016	
d3	Administración	e1	01/02/2013	

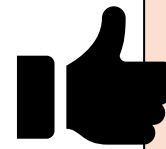
1.5. Tipo de relación 1:1

58



EMPLEADO

<u>codE</u>	nombre	...
e1	Laura López	
e2	María Martínez	
e3	Abel Ayala	
e4	Pedro Palao	
e5	Simona Sánchez	



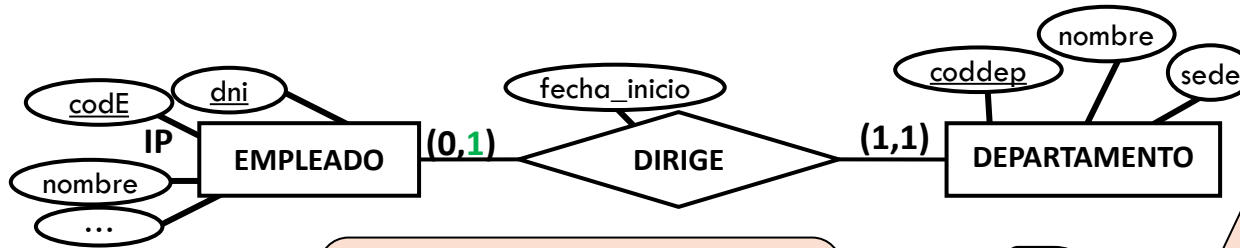
El atributo 'director' **debe ser clave**. Así se asegura que no puede repetir valores (un mismo empleado no puede dirigir varios departamentos):

DEPARTAMENTO

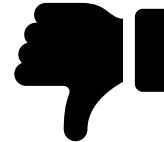
<u>coddep</u>	nombre	director	fecha_inicio	...
d1	Informática	e2	23/02/2015	
d2	Sistemas	e3	15/04/2016	
d3	Administración	e1	01/02/2013	

1.5. Tipo de relación 1:1

59



La clave ajena no puede repetir valores (cardinalidad máxima 1)



¿Sería correcto traducir **AL REVÉS**?
(añadir la clave ajena en la de participación opcional -- padre)

KO: Demasiados NULL en “departamento_dirigido” y “fecha_inicio_dirección” (para todos aquellos empleados que NO son directores, que serán la MAYORÍA)

EMPLEADO

<u>codE</u>	nombre	departamento_dirigido	fecha_inicio_direccion
e1	Laura López	d3	01/02/2013
e2	María Martínez	d1	23/02/2015
e3	Abel Ayala	d2	15/04/2016
e4	Pedro Palao	NULL	NULL
e5	Simona Sánchez	NULL	NULL

DEPARTAMENTO

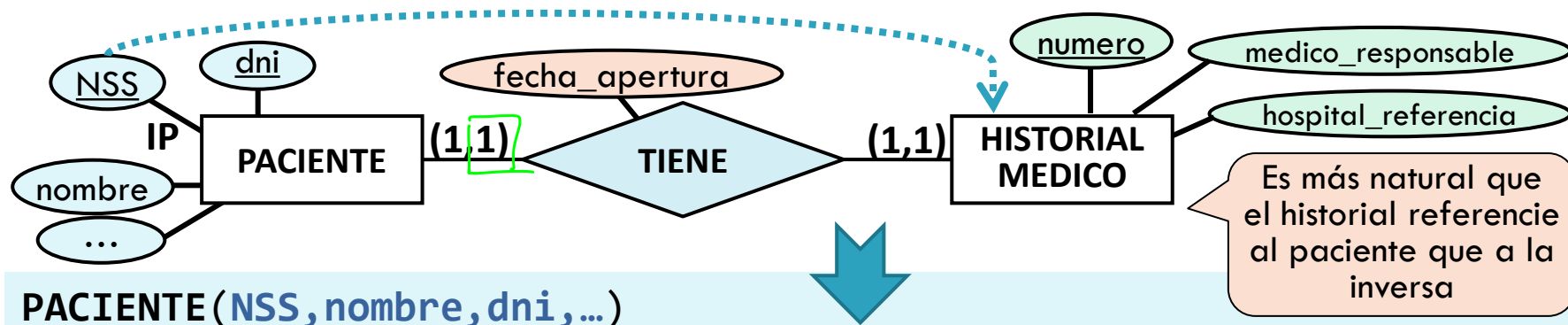
<u>coddep</u>	nombre
d1	Informática
d2	Sistemas
d3	Administración

1.5. Tipo de relación 1:1

60

b. Participación **obligatoria** en ambos lados.

- ▣ Traducir a una **clave ajena**, que se puede incluir en cualquiera de las dos relaciones
- ▣ Aunque lo habitual es que tenga más sentido que sea una concreta la que reference a la otra (usar el *sentido común*)



PACIENTE(NSS,nombre,dni,...)

...

HISTORIAL(numero,medico,hospital,paciente,fecha_apertura)

Admiten NULL: Ninguno

Clave primaria: numero. Clave alternativa: **paciente**★

Clave ajena: paciente Referencia_a PACIENTE(NSS)

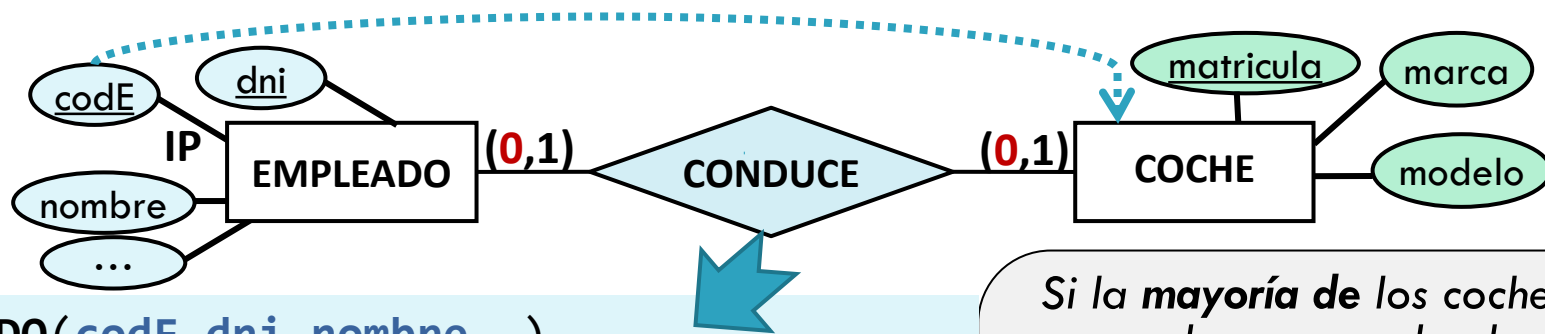
Hay que añadir un ASERTO para asegurar que **todo paciente tiene un historial**

1.5. Tipo de relación 1:1

61

c. Participación **opcional** en ambos lados

- Elegir “al azar” pero “con sentido común” los tipos de entidad ‘padre’ e ‘hijo’ y traducir a **clave ajena**



EMPLEADO(**codE**, dni, nombre, ...)

Admiten NULL: Ninguno

Clave primaria: codE; Clave alternativa: dni

COCHE(**matricula**, marca, modelo, **conductor**)

Admiten NULL: **conductor**

Clave primaria: matricula

Clave alternativa: **conductor**★

Clave ajena: conductor

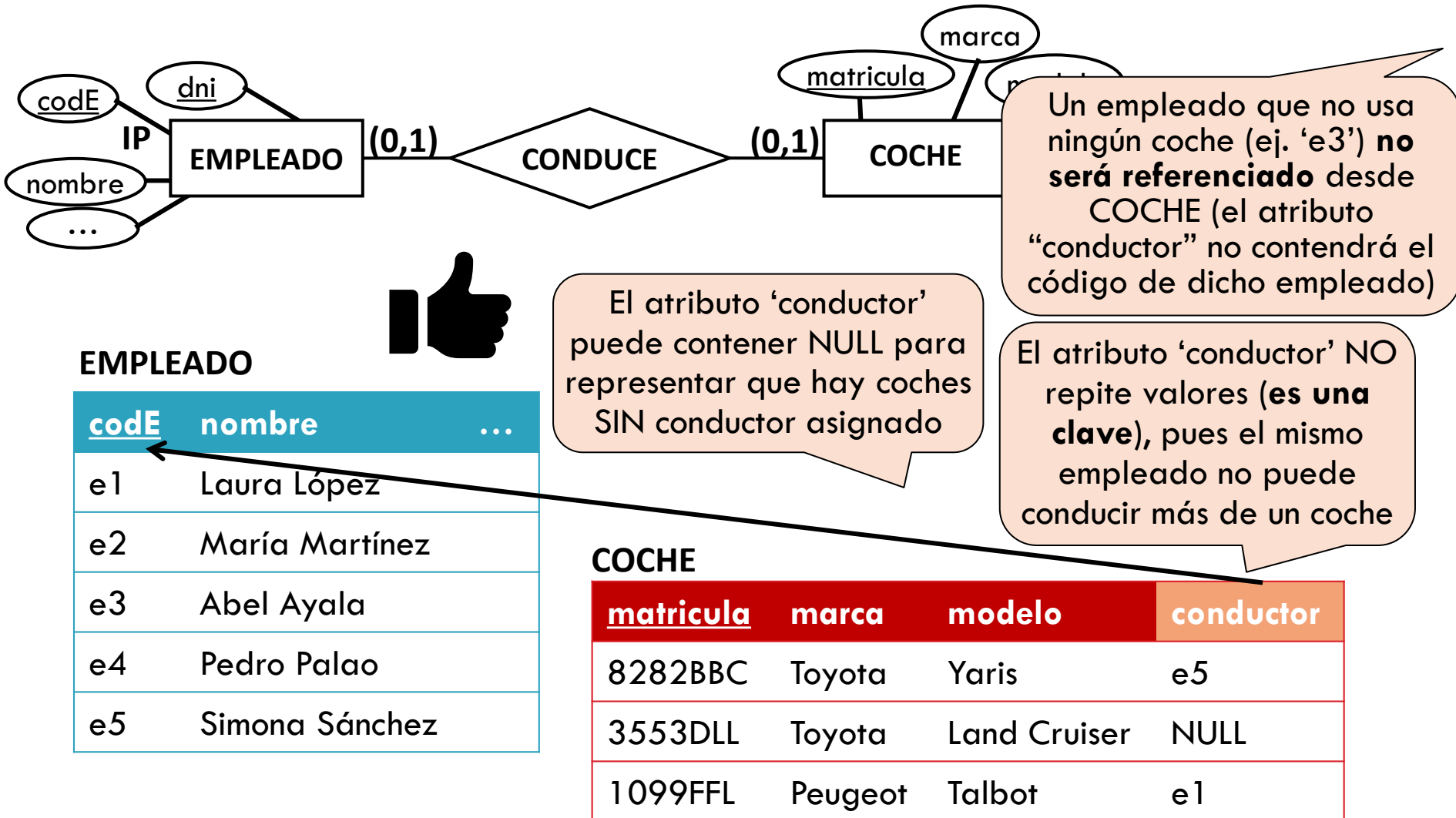
Referencia_a EMPLEADO(codE)

Si la **mayoría** de los coches son usados por empleados, y sólo una **minoría de** empleados usa coche, COCHE está “cerca” de la participación obligatoria:
 ► COCHE es el tipo de entidad hijo

❗ Recordamos: la clave ajena en la relación ‘hijo’ también es Clave Alternativa

1.5. Tipo de relación 1:1

62

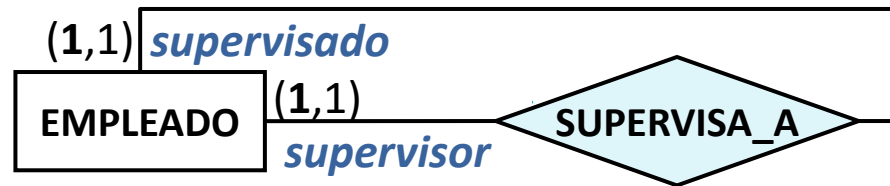


1.6. Tipo de relación **1:1 recursiva**

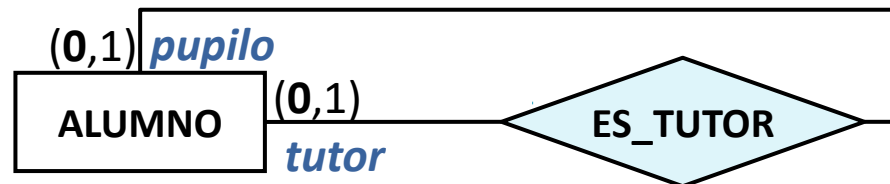
63

- ¿Incluir **una clave ajena** en la relación **o** crear una **nueva relación** (tabla) para el tipo de relación?
- Decidir con base en la **participación** de cada tipo de entidad (cardinalidad mínima)

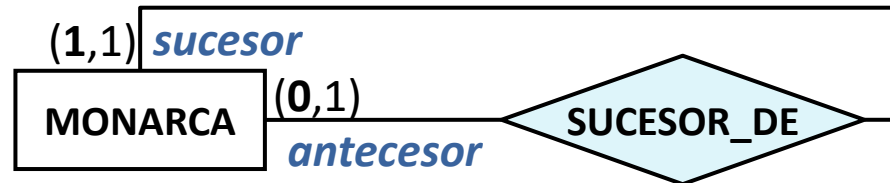
a. Obligatoria en
ambos lados



b. Opcional en
ambos lados



c. Obligatoria en
un lado



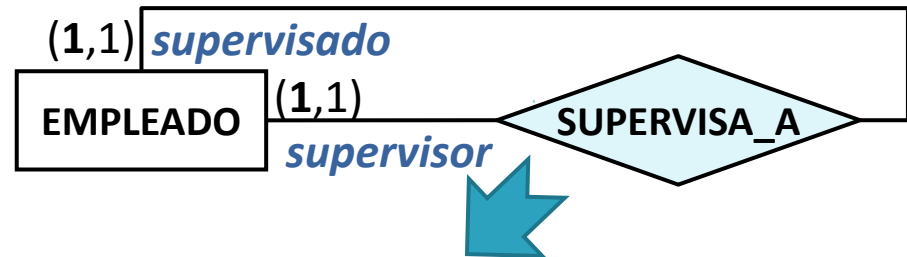
1.6. Tipo de relación 1:1 recursiva

64

a. Participación **obligatoria en ambos lados**

- Añadir a la relación correspondiente al tipo de entidad una **copia de su propia clave primaria**
- Dicha copia es una **clave ajena** que referencia a la misma relación, y debe renombrarse adecuadamente (usar roles)

Por la cardinalidad mínima 1 de "supervisor" hay que añadir un ASERTO para asegurar que **todo empleado supervisa a un empleado**



EMPLEADO(**code**, nombre, dni, direccion,..., **supervisor**)

Admiten NULL: Ninguno

Clave primaria: code

Claves alternativas: (dni), (**supervisor**)

Clave ajena: supervisor Referencia_a EMPLEADO(**code**)

Comprobar: **code**<>**supervisor**

❗ Recordamos: la clave ajena también es Clave Alternativa (por traducir una 1:1)

1.6. Tipo de relación 1:1 recursiva

ej Administrador practicas

65

b. Participación **opcional en ambos lados**

- ▣ Crear una **nueva relación** para el tipo de relación
 - Con 2 **copias de la clave primaria de la relación** en la que se ha traducido el tipo de entidad
 - Las dos son **claves ajenas** a la misma relación
 - Renombrar adecuadamente dichas claves ajenas (usar **roles**)

ALUMNO(dni,num_expediente,nombre,...)

Admiten NULL: Ninguno

Clave primaria: dni

Clave alternativa: num_expediente

ES_TUTOR(**tutor**, **pupilo**, fecha_inicio)

Admiten NULL: Ninguno

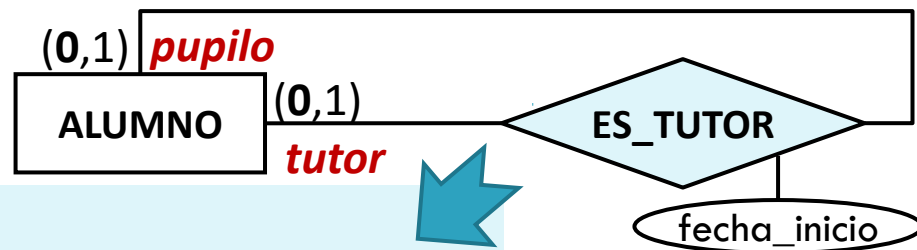
Clave primaria: **tutor**; Clave alternativa: **pupilo**

Clave ajena: tutor Referencia_a ALUMNO(dni)

Clave ajena: pupilo Referencia_a ALUMNO(dni)

Comprobar: **tutor<>pupilo**

< > = !=



① En la nueva relación, una de las claves ajenas será la **clave primaria**, y la otra será **clave alternativa**

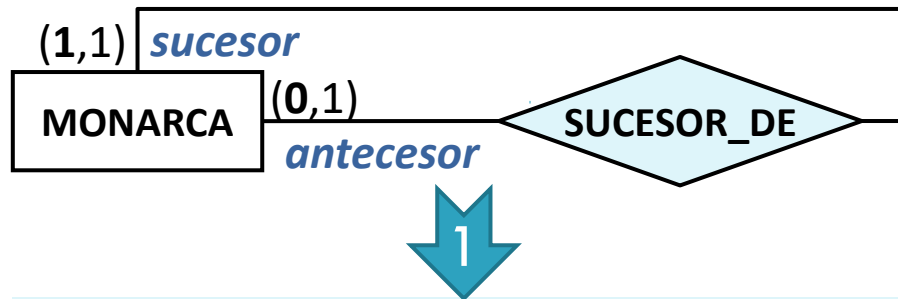
1.6. Tipo de relación 1:1 recursiva

66

c. Participación **obligatoria en un lado.**

▣ Traducir a **clave ajena**, como en el caso **a**.

- Tipo de entidad '**padre**': la de participación **opcional** - (0,1)
- Tipo de entidad '**hijo**': la de participación **obligatoria** - (1,1)



La **relación** ya definida para el tipo de entidad, siempre **corresponde al "hijo"**:
MONARCA = sucesor

MONARCA(nombre, casa, f_coronacion, fin_reinado, consorte, ..., **antecesor**)

Admiten NULL: fin_reinado, consorte

Clave primaria: nombre

Clave alternativa: antecesor

Clave ajena: antecesor Referencia_a MONARCA(nombre)

Comprobar: nombre<>antecesor

1.7. Tipo de **relación M:N**

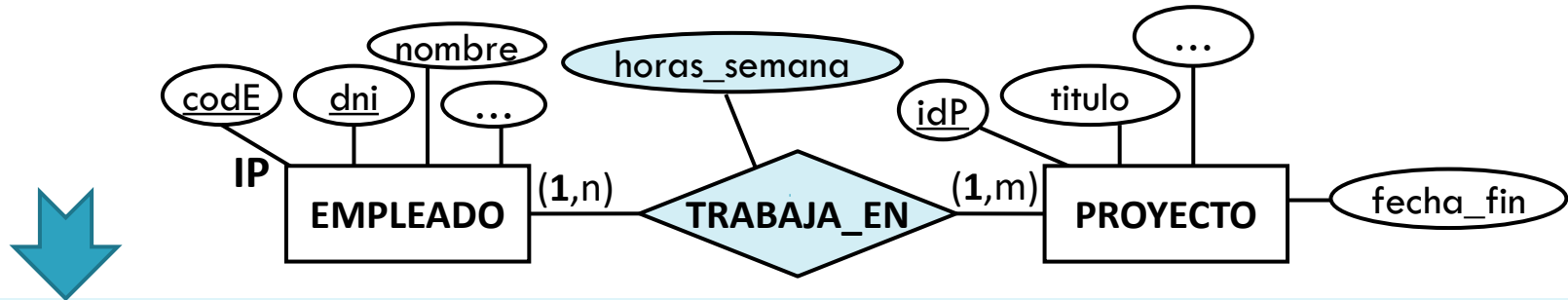
67

- ❑ Crear una **nueva relación (tabla)**
 - ❑ Añadir una copia de las claves primarias de los tipos de entidad conectados: serán **claves ajenas** a cada una de las relaciones correspondientes a dichos tipos de entidad
 - ❑ Incluir **atributos** (columnas) para los **atributos** del tipo de **relación**
 - ❑ La **clave primaria** de la nueva relación será **la concatenación de ambas claves ajenas**
 - ❑ Añadir un **aserto** por cada cardinalidad mínima 1



1.7. Tipo de relación M:N

68



TRABAJA_EN(*empleado*, *proyecto*, horas_semana)

Admiten NULL: Ninguno

Clave primaria: (*empleado*, *proyecto*)

Clave ajena: empleado Referencia_a EMPLEADO(*codE*)

Clave ajena: proyecto Referencia_a PROYECTO(*idP*)

ASERTO **RI_empleado_trabaja_en_proyectos** COMPROBAR_QUE
 (NO_EXISTE (una tupla en EMPLEADO
 donde el valor de "codE" NO_ESTÉ_ENTRE
 (valores de "empleado" en TRABAJA_EN))));

ASERTO **RI_proyecto_tiene_empleados** COMPROBAR_QUE
 (NO_EXISTE (una tupla en PROYECTO
 donde el valor de "idP" NO_ESTÉ_ENTRE
 (valores de "proyecto" en TRABAJA_EN))));

Las **cardinalidades mínimas 1** de los tipos de entidad conectados por la M:N hacen necesaria la definición de 2 **asertos**

1.7. Tipo de relación M:N

69

EMPLEADO

<u>codE</u>	nombre	...
e1	Laura López	
e2	María Martínez	
e3	Abel Ayala	
e4	Pedro Palao	
e5	Simona Sánchez	

PROYECTO

<u>idP</u>	título	...
p1	Proyecto 1	
p2	Proyecto 2	
p3	Proyecto 3	
p4	Proyecto 4	

TRABAJA_EN

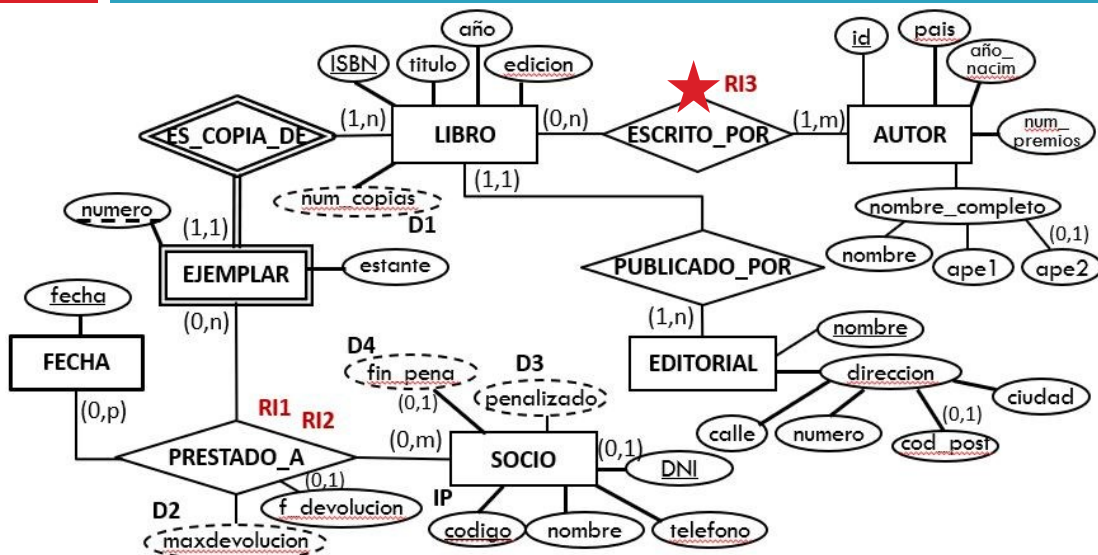
<u>empleado</u>	<u>proyecto</u>	horas_semana
e2	p2	10
e2	p4	25
e3	p3	35
e1	p1	30
e4	p1	20
e5	p2	35
e4	p4	15



Cada fila de TRABAJA_EN se distingue de las demás mediante los valores de los atributos (empleado, proyecto): esa es su clave

Ejemplo. Traducción de tipos de relación M:N

70



EDITORIAL(nombre,calle, numero,cod_post,ciudad)
Admiten NULL: **cod_post**
Clave primaria: **nombre**

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
Admiten NULL: **DNI,fin_pena**
Clave primaria: **codigo**
Clave alternativa: **DNI**

AUTOR(id,nombre,pais,año_nacim, num_premios,nombre,ape1,ape2)
Admiten NULL: **ape2**
Clave primaria: **id**

LIBRO(ISBN,título,año,edicion, num_copias, **editorial**)
Admiten NULL: Ninguno
Clave primaria: **ISBN**
Clave ajena: **editorial**
Referencia_a **EDITORIAL(nombre)**
Derivado: **num_copias** [para cada...]

EJEMPLAR(numero, estante, **ISBN**)
Admiten NULL: Ninguno
Clave primaria: (**ISBN, numero**)
Clave ajena: **ISBN**
Referencia_a **LIBRO(ISBN)**

ESCRITO_POR(libro,autor)

Admiten NULL: Ninguno

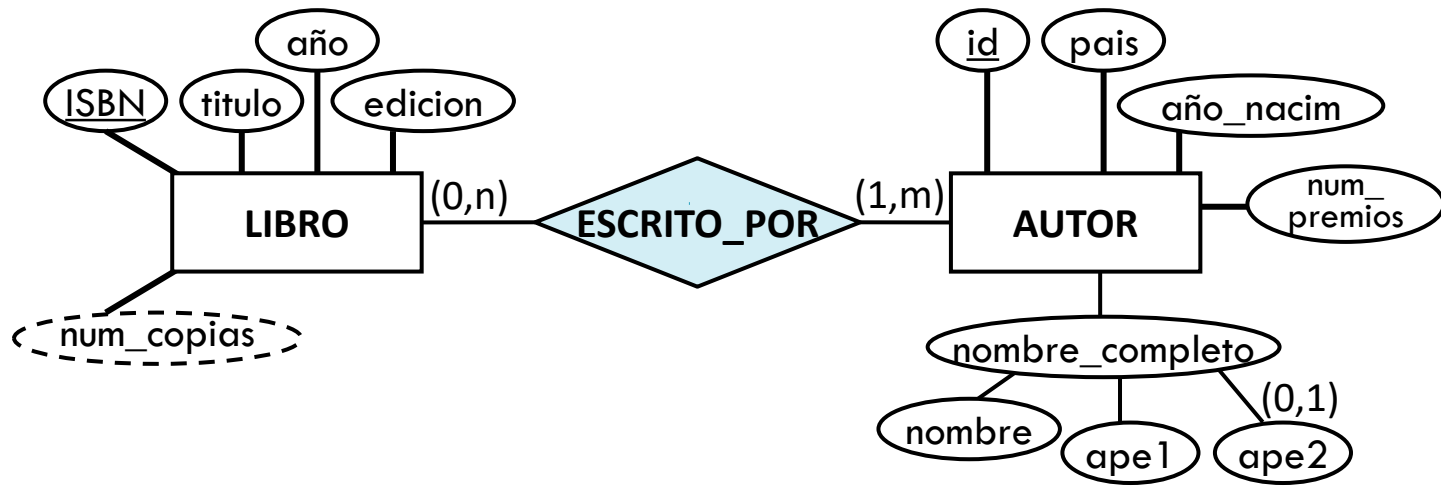
Clave primaria: (**libro, autor**)

Clave ajena: **libro** Referencia_a **LIBRO(ISBN)**

Clave ajena: **autor** Referencia_a **AUTOR(id)**

1.7. Tipo de relación M:N

71



ESCRITO_POR(*libro*, *autor*)

Admiten NULL: Ninguno

Clave primaria: (*libro*, *autor*)

Clave ajena: libro Referencia_a LIBRO(ISBN)

Clave ajena: autor Referencia_a AUTOR(id)

ASERTO **RI_autor_escribe_libros** COMPROBAR_QUE

(NO_EXISTE (una tupla de AUTOR
donde el valor de "id" NO_ESTÉ_ENTRE
(valores de "autor" en ESCRITO_POR)));

La **cardinalidad mínima 1** del tipo de entidad AUTOR hace necesaria la definición de un **aserto**

1.7. Tipo de relación M:N

72

LIBRO

<u>ISBN</u>	titulo	...
9788497404518	La Celestina	
9788467006971	Para Ana (de tu muerto)	
9788497939072	Crimen y Castigo	
9788498003116	Fundación	
9788493806125	Don Quijote de la Mancha	
9788435018364	Yo, robot	

AUTOR

<u>id</u>	...	apellido1	...
a1		Cervantes	
a2		Del Val	
a3		Dostoievski	
a4		Asimov	
a5		De Rojas	
a6		Roca	

ESCRITO_POR

<u>libro</u>	<u>autor</u>
9788497404518	a5
9788467006971	a2
9788467006971	a6
9788497939072	a3
9788498003116	a4
9788493806125	a1
9788435018364	a4

El atributo "libro" puede tener valores repetidos: hay libros con varios autores

El atributo "autor" puede tener valores repetidos: hay autores con varios libros

Cada fila de ESCRITO_POR se distingue de las demás mediante los valores de los atributos (libro, autor): esa es su clave, compuesta



1.7. Tipo de relación M:N

73

LIBRO

ISBN	titulo	...
9788497404518	La Celestina	
9788467006971	Para Ana (de tu muerto)	
9788497939072	Crimen y Castigo	
9788498003116	Fundación	
9788493806125	Don Quijote de la Mancha	
9788435018364	Yo, robot	
9788408055785	Las mil y una noches	

AUTOR

id	...	apellido1	...
a1		Cervantes	
a2		Del Val	
a3		Dostoievski	
a4		Asimov	
a5		De Rojas	
a6		Roca	

ESCRITO_POR

libro	autor
9788497404518	a5
9788467006971	a2
9788467006971	a6
9788497939072	a3
9788498003116	a4
9788493806125	a1
9788435018364	a4

*a) ¿Cómo se representa la cardinalidad mínima 0 de **LIBRO** en **ESCRITO_POR**?*
*b) ¿Cómo se representa un **LIBRO** cuyo autor se desconoce (libro **anónimo**, sin autor registrado)? Veámoslo en la siguiente diapositiva*

1.7. Tipo de relación M:N

74

LIBRO

ISBN	titulo	...
9788497404518	La Celestina	
9788467006971	Para Ana (de tu muerto)	
9788497939072	Crimen y Castigo	
9788498003116	Fundación	
9788493806125	Don Quijote de la Mancha	
9788435018364	Yo, robot	
9788408055785	Las mil y una noches	

AUTOR

id	...	apellido1	...
a1		Cervantes	
a2		Del Val	
a3		Dostoievski	
a4		Asimov	
a5		De Rojas	
a6		Roca	

↑
ESCRITO_POR

libro	autor
9788497404518	a5
9788467006971	a2
9788467006971	a6
9788497939072	a3
9788498003116	a4
9788493806125	a1
9788435018364	A4
9788408055785	NULL

ERROR

— KO!

¿¿Mediante una fila en ESCRITO_POR con un NULL en el "autor"??

La **clave primaria** de ESCRITO_POR es (libro, autor), por lo que NINGUNO de esos atributos puede contener un NULL

1.7. Tipo de relación M:N

75

LIBRO

ISBN	titulo	...
9788497404518	La Celestina	
9788467006971	Para Ana (de tu muerto)	
9788497939072	Crimen y Castigo	
9788498003116	Fundación	
9788493806125	Don Quijote de la Mancha	
9788435018364	Yo, robot	
9788408055785	Las mil y una noches	

AUTOR

id	...	apellido1	...
a1		Cervantes	
a2		Del Val	
a3		Dostoievski	
a4		Asimov	
a5		De Rojas	
a6		Roca	

↑ ESCRITO_POR

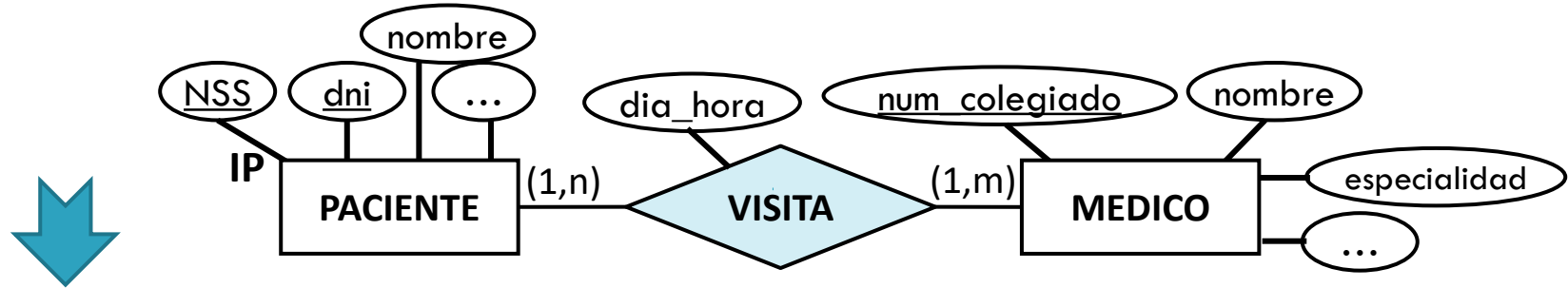
libro	autor
9788497404518	a5
9788467006971	a2
9788467006971	a6
9788497939072	a3
9788498003116	a4
9788493806125	a1
9788435018364	a4

El libro anónimo “Las mil y una noches” **simplemente no aparece en la relación ESCRITO_POR** : ninguna fila le hace referencia (lógico, porque no tiene autores)



1.7. Tipo de relación M:N

76



VISITA(*paciente*, *doctor*, dia_hora)

Admiten NULL: Ninguno

Clave primaria: (*paciente*, *doctor*)

Se repite

Pero... ¿Esto ES clave?

Clave ajena: paciente Referencia_a PACIENTE(NSS)

Clave ajena: doctor Referencia_a MEDICO(num_colegiado)

ASERTO **RI_paciente_visita** COMPROBAR_QUE

(NO_EXISTE (una tupla en PACIENTE
donde el valor de "NSS" NO_ESTÉ_ENTRE
(valores de "paciente" en VISITA)));

ASERTO **RI_medico_visita** COMPROBAR_QUE

(NO_EXISTE (un MEDICO
donde el valor de "num_colegiado" NO_ESTÉ_ENTRE
(valores de "doctor" en VISITA)));

1.7. Tipo de relación M:N

77

PACIENTE

<u>NSS</u>	nombre	...
123456789012	Higinia Hernandez	
456789012345	Torcuato Torres	
678901234567	Bonifacia Berza	
234567890123	Lucrecio Lorente	
567890123456	Zacarías Zornoza	
345678901234	Damiana Díaz	

MEDICO

<u>num</u>	...
<u>colegiado</u>	nombre
12345	Sonsoles Sáez
11223	Vilma Val
44332	Jonás Jaén

VISITA

<u>paciente</u>	<u>doctor</u>	día_hora
123456789012	12345	04/11/2019 10:05
456789012345	12345	04/11/2019 11:35
456789012345	12345	08/11/2019 12:20
678901234567	11223	04/11/2019 10:05
234567890123	44332	04/11/2019 10:05
567890123456	12345	09/11/2019 09:15
345678901234	44332	05/11/2019 13:00
567890123456	11223	06/11/2019 11:45

El atributo “paciente” puede tener valores repetidos: hay pacientes que acuden a varias visitas

El atributo “doctor” puede tener valores repetidos: los médicos atienden varias visitas

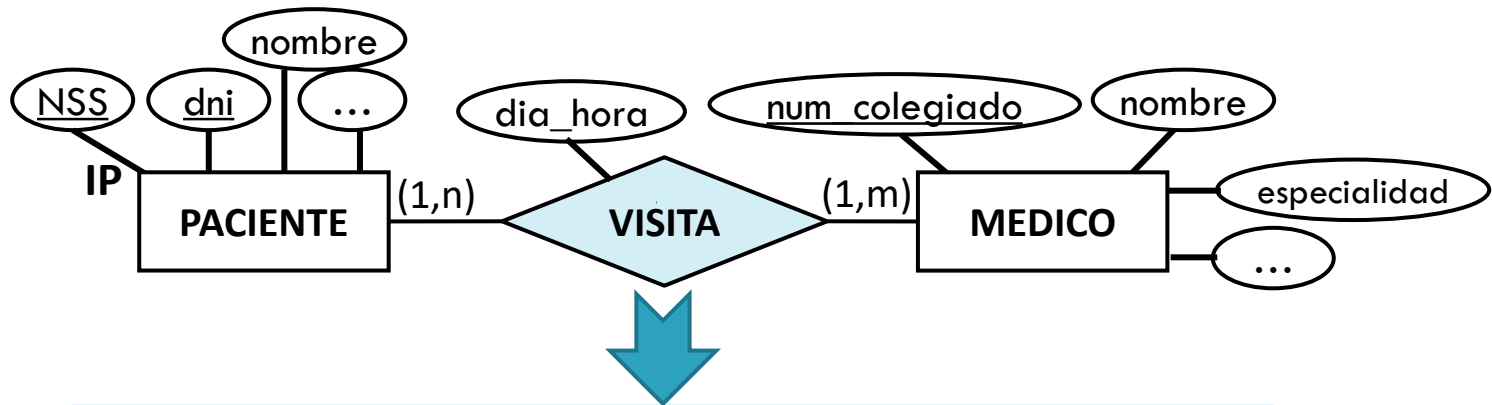
Pero un mismo paciente puede visitar varias veces al mismo médico: cada VISITA concreta **NO se distingue de las demás sólo con los valores (paciente, doctor)**

🧠 (paciente, doctor) **NO** es CLAVE

1.7. Tipo de relación M:N

78

- Si la **concatenación de ambas claves ajenas** **no** forma una **clave**, significa que en el esquema conceptual se **omitió un tipo de entidad** conectado al tipo de relación
 - Es un error de diseño que se puede **rectificar** ahora
 - **Añadir alguno de los atributos** de la relación para conseguir una clave



VISITA(paciente, doctor, dia_hora)

Clave primaria: (paciente, doctor, dia_hora)

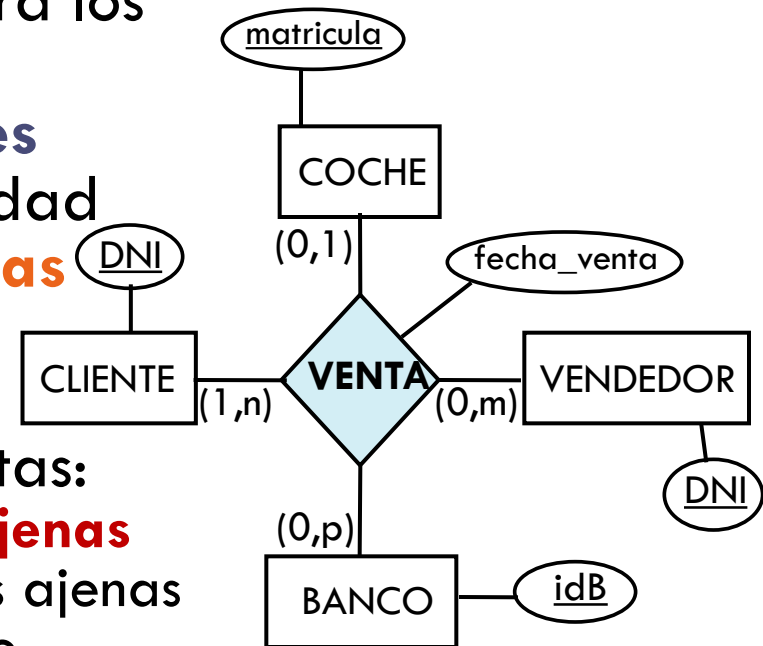
Ahora SÍ

...

1.8. Tipo de **relación n-aria**

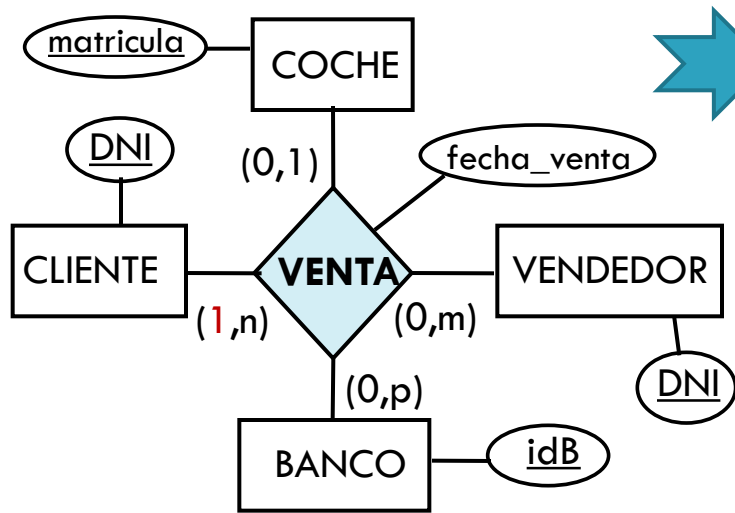
79

- Crear una **nueva relación (tabla)**
 - ▣ Incluir **atributos** (columnas) para los atributos del tipo **de relación**
 - ▣ Añadir una **copia de las claves primarias** de los tipos de entidad conectados: serán **claves ajenas** a cada uno de ellos
 - ▣ La **clave primaria** de la nueva relación será una de estas:
 - **Concatenación de las claves ajenas**
 - Combinación de **algunas** claves ajenas
 - (Subsanación de error de diseño conceptual) La concatenación de **varias claves ajenas y algún atributo**



1.8. Tipo de relación n-aria

80



Incluye una clave ajena que referencia a la clave primaria de cada relación correspondiente a cada tipo de entidad que conecta

claves ajenas nunca tienen NULL

VENTA(*matricula*, *vendedor*, *cliente*, *banco*, fecha_venta)

Admiten NULL: ninguno

Clave primaria:

(*matricula*, *vendedor*, *cliente*, *banco*)

Clave ajena: matricula

Referencia_a COCHE(matricula)

Clave ajena: vendedor

Referencia_a VENDEDOR(DNI)

Clave ajena: cliente

Referencia_a CLIENTE(DNI)

Clave ajena: banco

Referencia_a BANCO(idB)

ASERTO **RI_cliente_venta** COMPROBAR_QUE

(NO_EXISTE (una tupla de CLIENTE
donde el valor de "DNI"
NO_ESTÉ_ENTRE (valores de
"cliente" en VENTA)));

1.8. Tipo de relación n-aria

81

COCHE

<u>matricula</u>	marca	...
0102BCD	Toyota	
2233VVB	Seat	
9988LDF	Seat	
4455PKQ	Toyota	
3366QWR	Mercedes	
1234MNP	Peugeot	

VENTA

<u>matricula</u>	vendedor	cliente	banco	fecha_venta
0102BCD	56789012S	22334455D	B02	01/11/2019
2233VVB	67890123V	11223344B	B04	10/12/2018
9988LDF	56789012S	44556677K	B01	06/09/2019
4455PKQ	67890123V	11223344B	B04	22/02/2019
1234MNP	56789012S	33445566G	B04	30/10/2019

VENDEDOR

<u>DNI</u>	nombre	...
56789012S	Salvadora Sueño	
67890123V	Viriato Valles	
78901234M	Melania Mirón	

CLIENTE

<u>DNI</u>	nombre	...
11223344B	Bonifacio Baños	
22334455D	Dionisia Dimas	
33445566G	Gregoria Güell	
44556677K	Karino Kent	

BANCO

<u>id</u>	nombre	...
B02	BBVA	
B04	Banco Santander	
B01	Bankia	
B03	Evo Banco	
B05	Banco Sabadell	
B06	Caixa Bank	

*¿Qué atributos pueden tener **valores repetidos**?
¿Cuáles no?*

1.8. Tipo de relación n-aria

82

□ A veces, la **clave primaria puede reducirse**:



- ▣ La cardinalidad máxima de COCHE en VENTA es 1, lo que significa que un coche participa, como mucho, en una relación de tipo VENTA (es adquirido una única vez).
- ▣ Esto indica que un coche identifica unívocamente la venta en la que participa, de forma que “matricula” puede ser la clave primaria de VENTA

`VENTA(matricula, vendedor, cliente, banco, fecha_venta)`

Admiten NULL: **ninguno**

Clave primaria: **matricula**

Ningún atributo admite nulos para evitar ventas sin cliente, sin vendedor o sin banco

Clave ajena: matricula Referencia_a COCHE(matricula)

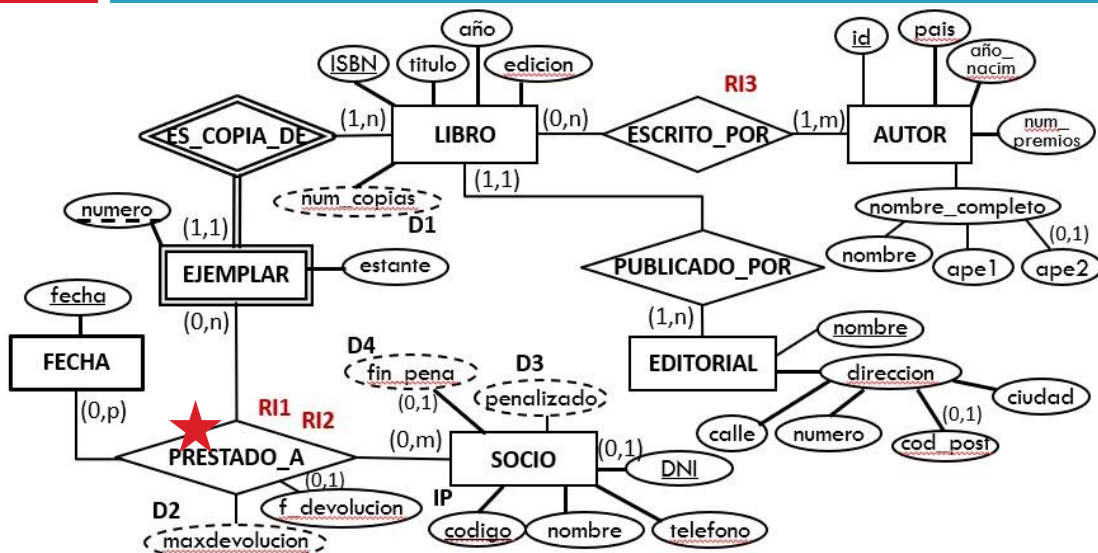
Clave ajena: vendedor Referencia_a VENDEDOR(DNI)

Clave ajena: cliente Referencia_a CLIENTE(DNI)

Clave ajena: banco Referencia_a BANCO(idB)

Ejemplo. Traducción de tipos de relación n-arias

83



EDITORIAL(nombre,calle, numero,cod_post,ciudad)
Admiten NULL: **cod_post**
Clave primaria: **nombre**

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
Admiten NULL: **DNI,fin_pena**
Clave primaria: **codigo**
Clave alternativa: **DNI**
Derivado: **fin_pena** [...cálculo...]

AUTOR(id,nombre,pais,año_nacim, num_premios,nombre,ape1,ape2)
Admiten NULL: **ape2**
Clave primaria: **id**

LIBRO(ISBN,título,año,edicion, num_copias, editorial)
Admiten NULL: Ninguno
Clave primaria: **ISBN**
Clave ajena: **editorial**
Referencia_a **EDITORIAL**(nombre)
Derivado: **num_copias** [para cada...]

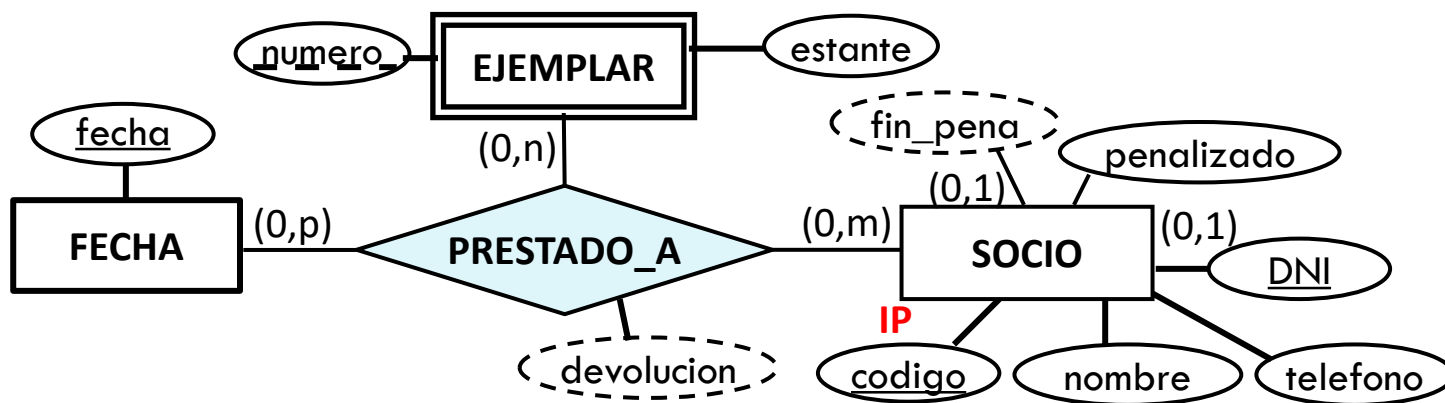
EJEMPLAR(numero, estante, ISBN)
Admiten NULL: Ninguno
Clave primaria: (**ISBN, numero**)
Clave ajena: **ISBN**
Referencia_a **LIBRO**(ISBN)

ESCRITO POR(libro,autor)
Admiten NULL: Ninguno
Clave primaria: (**libro, autor**)
Clave ajena: **libro** Referencia_a **LIBRO**(ISBN)
Clave ajena: **autor** Referencia_a **AUTOR**(id)

PRESTAMO(socio,libro,ejemplar,fecha,devolucion)
Admiten NULL: Ninguno
Clave primaria: (**socio,libro,ejemplar,fecha**)
Clave ajena: **socio** Referencia_a **SOCIO**(codigo)
Clave ajena: (**libro, ejemplar**)
Referencia_a **EJEMPLAR**(ISBN, numero)
Derivado: **devolucion** [fecha+15dias]

1.8. Tipo de relación n-aria

84



PRESTAMO(*libro*, *ejemplar*, socio, fecha, devolucion)

Admiten NULL: Ninguno

Clave primaria: (*libro*, *ejemplar*, socio, *fecha*)

Clave ajena: socio Referencia_a SOCIO(codigo)

Clave ajena: (*libro*, *ejemplar*) Referencia_a EJEMLAR(ISBN, numero)

Valor por defecto calculado: devolucion=fecha+15días

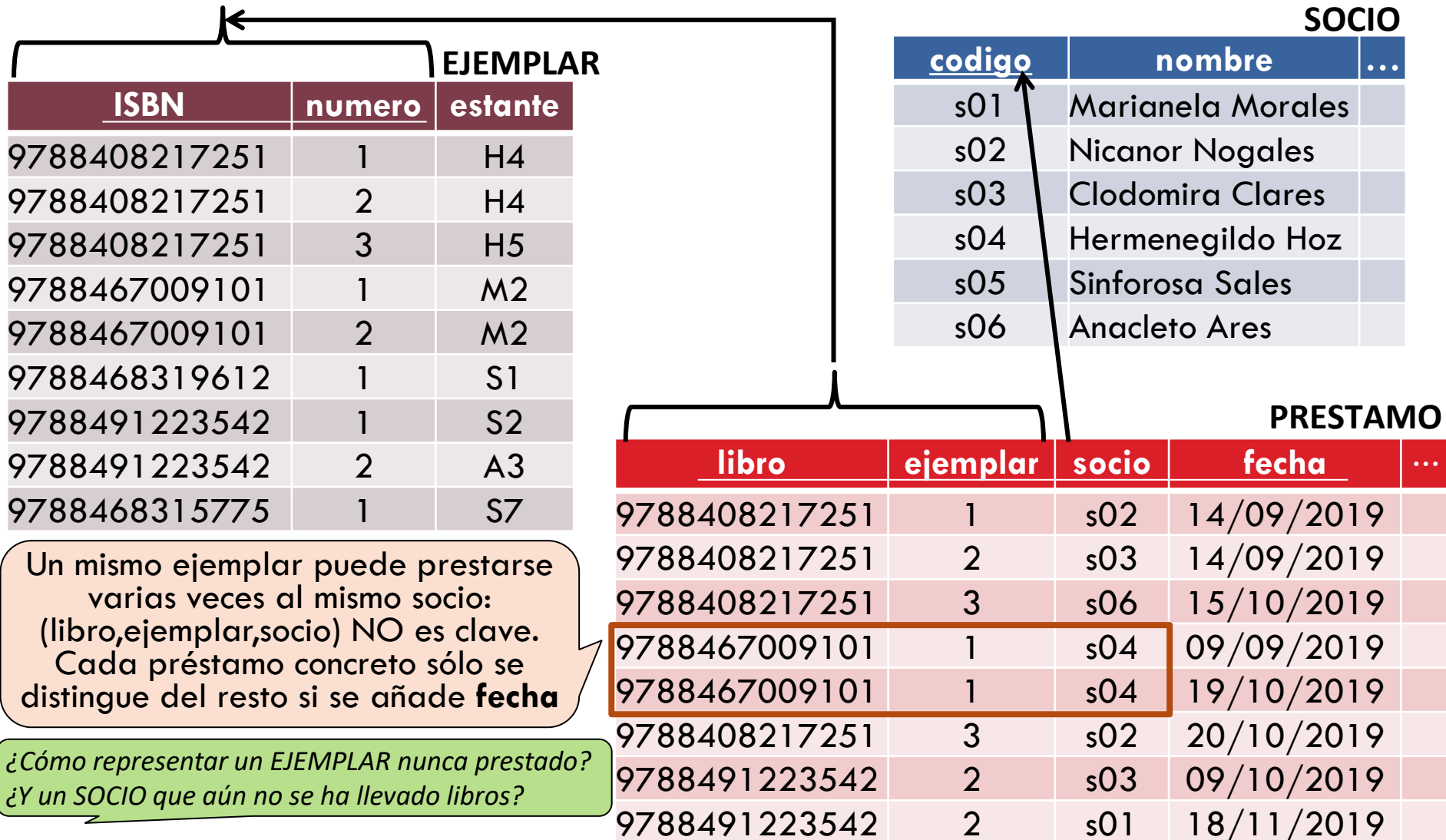
¿Merece la pena **crear** la relación (tabla) **FECHA**?

¿Sobra 'socio' de la Clave Primaria?

Recordamos: esta **clave ajena** debe ser **compuesta**, porque lo es la clave primaria a la que referencia (la relación EJEMLAR viene de un tipo de entidad débil)

1.8. Tipo de relación n-aria

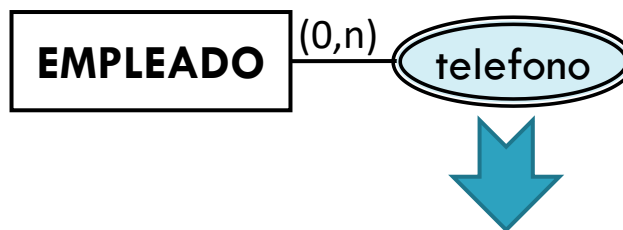
85



1.9. Atributo multivalorado

86

- Crear una **nueva relación (tabla)**
 - ▣ Incluir un **atributo** (columna) con el mismo nombre que el atributo multivalorado
 - ▣ Añadir **una copia de la clave primaria** del tipo de entidad a la que está conectado: será una **clave ajena**
 - ▣ La **clave primaria** de la nueva relación será uno de estos:
 - La **concatenación de la clave ajena y el atributo**
 - El **atributo** “en solitario”



La clave primaria es **compuesta** porque es posible que varios empleados tengan el número de teléfono (el fijo del despacho compartido). Si cada teléfono sólo pudiera pertenecer a un empleado, la clave primaria sería “telefono”

TELEFONO_EMPLEADO(*empleado*, telefono)

Admiten NULL: ninguno

Clave primaria: (**empleado**, **telefono**)

Clave ajena: empleado Referencia_a EMPLEADO(code)

1.9. Atributo multivalorado

87

EMPLEADO

<u>codE</u>	nombre	...
e1	Laura López	
e2	María Martínez	
e3	Abel Ayala	
e4	Pedro Palao	
e5	Simona Sánchez	

TELEFONO_EMPLEADO

empleado	telefono
e2	555111222
e2	555333666
e3	555111222
e4	555777888
e4	555999888
e5	555000444
e4	555777666



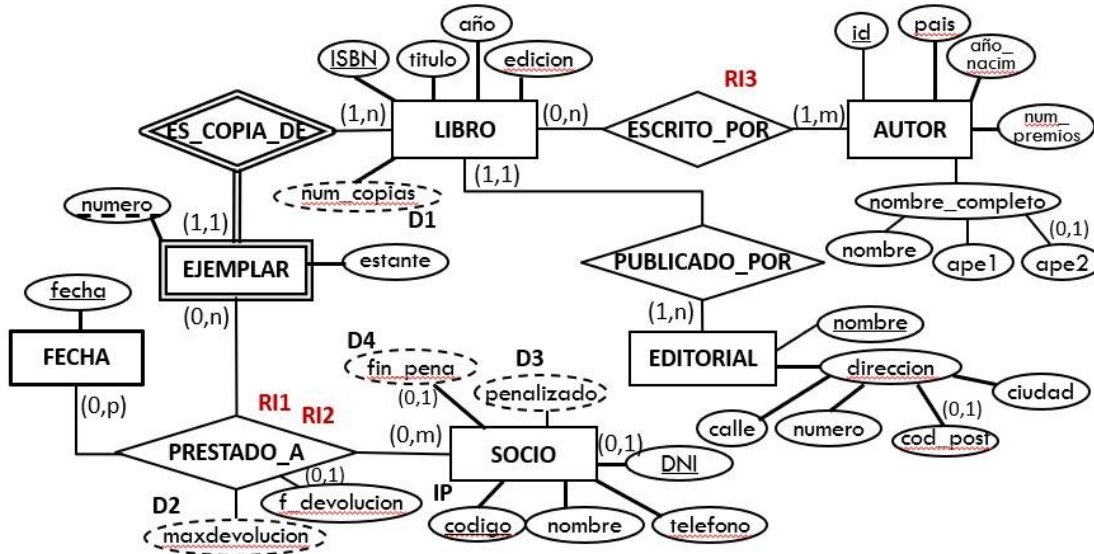
Un empleado puede tener varios teléfonos distintos: si tiene 3 teléfonos, habrá 3 filas para ese empleado en TELEFONO_EMPLEADO

Varios empleados pueden tener el mismo teléfono (por ejemplo si comparten despacho, y en él hay un teléfono fijo)

Un empleado puede NO tener teléfono (e1, por ejemplo): será un empleado cuyo código NO aparece en la relación TELEFONO_EMPLEADO. Ninguna fila le hace referencia

Ejemplo. Resultado de la Traducción

88



ESCRITO_POR(libro,autor)

Admiten NULL: Ninguno

Clave primaria: (libro, autor)

Clave ajena: libro Referencia_a LIBRO(ISBN)

Clave ajena: autor Referencia_a AUTOR(id)

PRESTAMO(socio,libro,ejemplar,fecha,devolucion)

Admiten NULL: Ninguno

Clave primaria: (socio,libro,ejemplar,fecha)

Clave ajena: socio Referencia_a SOCIO(codigo)

Clave ajena: (libro, ejemplar)

Referencia_a EJEMPLAR(ISBN, numero)

Derivado: devolucion [fecha+15dias]

EDITORIAL(nombre,calle, numero,cod_post,ciudad)
Admiten NULL: cod_post
Clave primaria: nombre

SOCIO(codigo,DNI,nombre, telefono,penalizado, fin_pena)
Admiten NULL: DNI,fin_pena
Clave primaria: codigo
Clave alternativa: DNI
Derivados: fin_pena [...cálculo...]

AUTOR(id,nombre,pais,año_nacim, num_premios,nombre,ape1,ape2)
Admiten NULL: ape2
Clave primaria: id

LIBRO(ISBN,título,año,edicion, num_copias, editorial)
Admiten NULL: Ninguno
Clave primaria: ISBN
Clave ajena: editorial
Referencia_a EDITORIAL(nombre)
Derivado: num_copias [para cada...]

EJEMPLAR(numero, estante, ISBN)
Admiten NULL: Ninguno
Clave primaria: (ISBN, numero)
Clave ajena: ISBN
Referencia_a LIBRO(ISBN)

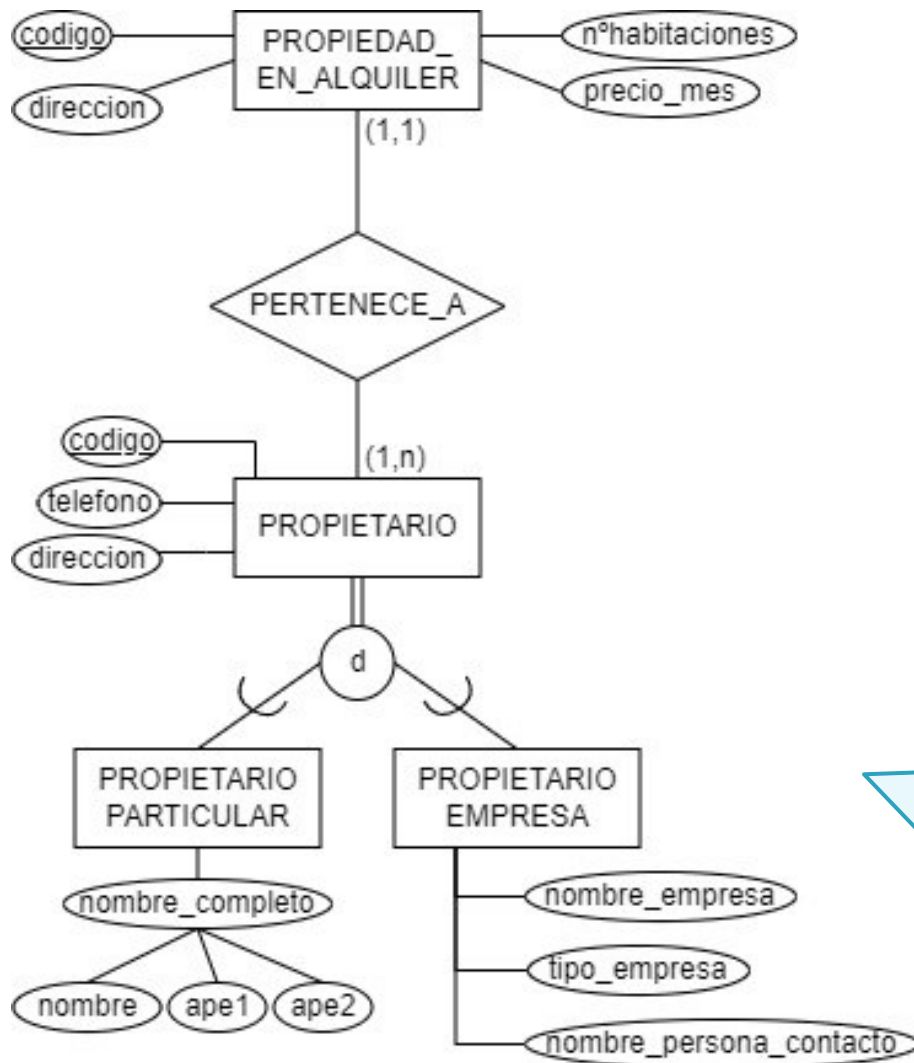
Reajustes de la traducción

89

- Estas son las pautas generales que podemos aplicar para traducir esquemas conceptuales a esquemas relacionales
- Pero siempre hay que tener en cuenta que, según los tipos de relación en los que participan los tipos de entidad y la existencia de jerarquías, a veces habrá que **adaptar la traducción** para conseguir un resultado (esquema lógico) correcto
- Veámoslo con un ejemplo...

Reajuste de la traducción

90



Traducimos los tipos de entidad:

PROPIEDAD_EN_ALQUILER (codigo, direccion, num_habitaciones, precio_mes)

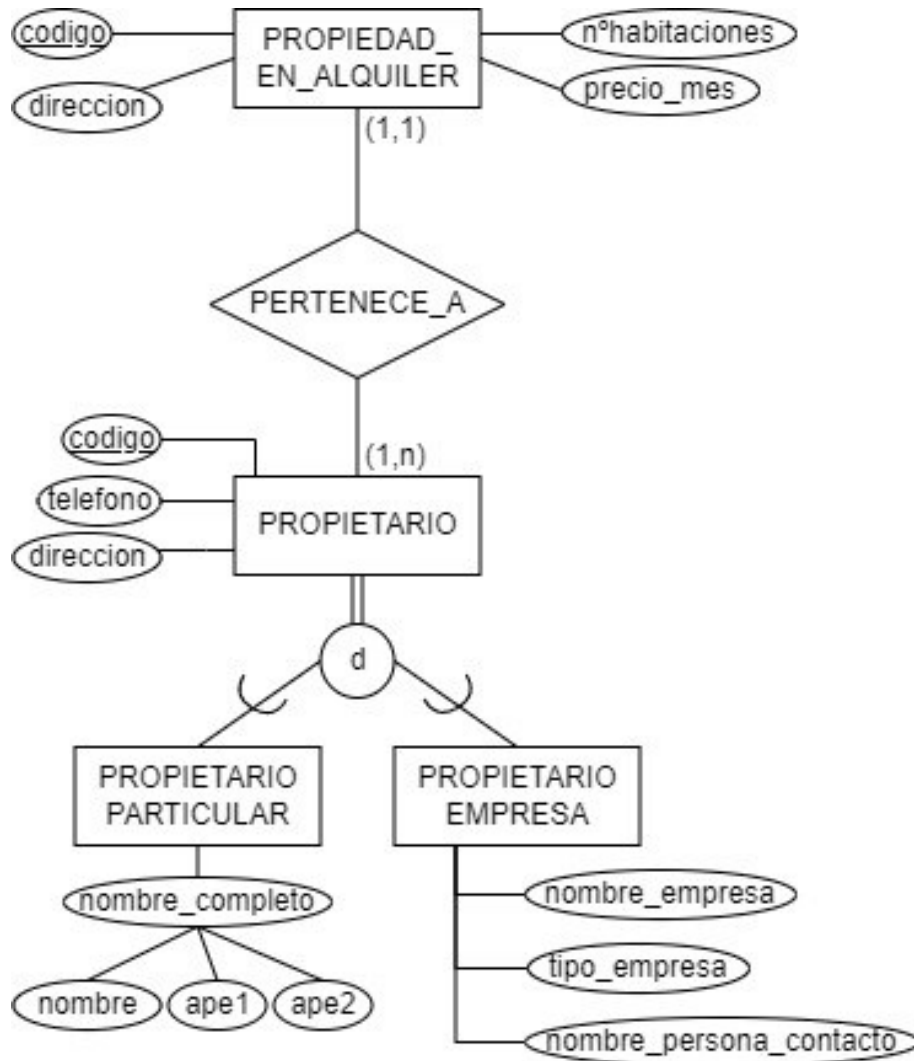
PROPIETARIO PARTICULAR(codigo, telefono, direccion, nombre, ape1, ape2)

PROPIETARIO EMPRESA(codigo, telefono, direccion, nombre_empresa, tipo_empresa, nombre_persona_contacto)

Al ser disjunta y total, la jerarquía se traduce a una relación por cada combinación **supertipo/subtipo**, más un **ASERTO** que asegure que un **PROPIETARIO PARTICULAR** no puede tener el mismo código que un **PROPIETARIO EMPRESA**

Reajuste de la traducción

91



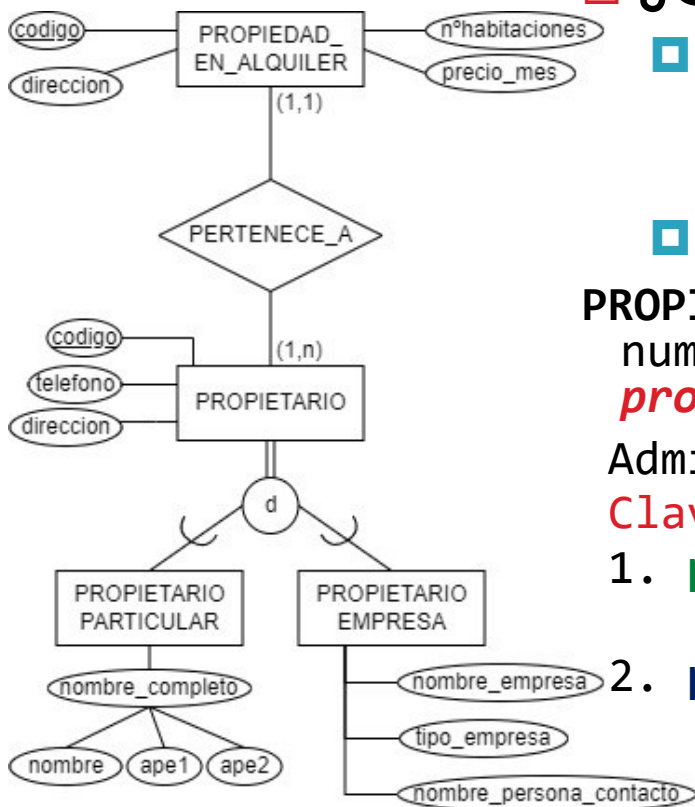
- Traducimos el **tipo de relación 1:N**, y vemos que
 - ▣ Hijo: PROPIEDAD_EN_ALQUILER
 - ▣ Padre: PROPIETARIO

PROPIEDAD_EN_ALQUILER (codigo, direccion, num_habitaciones, precio_mes, *propietario*)
 Clave ajena: propietario
 REFERENCIA_A *¿¿??*

- ¿A qué tabla debe referenciar? ¡Tenemos dos tipos de propietarios!
 - ▣ Una clave ajena sólo puede referenciar a **UNA** tabla y no a DOS

Reajuste de la traducción

92



□ ¿Cómo lo hacemos?

- ▣ Cada propiedad en alquiler debe tener un propietario que será o bien un particular, o bien una empresa
- ▣ Hemos de **definir DOS claves ajenas**

PROPIEDAD_EN_ALQUILER (codigo, direccion, num_habitaciones, precio_mes, *prop_particular, prop_empresa*)

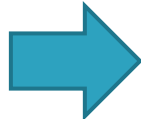
Admiten NULL: prop_particular, prop_empresa

Claves ajenas:

1. *prop_particular* REFERENCIA_A PROPIETARIO PARTICULAR(codigo)
2. *prop_empresa* REFERENCIA_A PROPIETARIO EMPRESA(codigo)

Comprobar:

- (prop_particular IS NOT NULL AND prop_empresa IS NULL)
- OR (prop_particular IS NULL AND prop_empresa IS NOT NULL)



Resumen de la traducción MER → MR

93

Elemento del MER	Elemento(s) del Modelo Relacional en el que se traduce
Tipo de entidad fuerte	Relación (tabla)
Tipo de entidad débil	Relación (tabla) . Tras traducir el tipo de relación identificador, se podrá formar la clave primaria como la unión de la clave ajena y el discriminante
Tipo de relación 1:N	Clave ajena (en la relación correspondiente al tipo de entidad “hijo” -- con cardinalidad máxima 1)
Tipo de relación 1:N recursiva	Clave ajena en la relación que referencia a la misma relación (al “padre”). La relación juega el papel de “hijo”
Tipo de relación 1:1 con participación opcional en un lado	Clave ajena (en la relación correspondiente al tipo de entidad “hijo” -- con cardinalidad mínima 1) ▶ La clave ajena también es clave alternativa
Tipo de relación 1:1 con participación obligatoria en ambos lados	Clave ajena en una de las dos relaciones ▶ La clave ajena también es clave alternativa O , si los tipos de entidad son “sinónimos”, integrar todo en una sola relación (tabla)
Tipo de relación 1:1 con participación opcional en ambos lados	Clave ajena en una de las dos relaciones ▶ La clave ajena también es clave alternativa

Resumen de la traducción MER → MR

94

Elemento del MER	Elemento(s) del Modelo Relacional en el que se traduce
Tipo de relación 1:1 recursiva con participación obligatoria en ambos lados	Clave ajena que referencia a la propia relación ▶ La clave ajena también es clave alternativa
Tipo de relación 1:1 recursiva con participación opcional en ambos lados	Relación (tabla) con 2 claves ajenas que referencian a la misma relación (la correspondiente al tipo de entidad) ▶ Una clave ajena es clave primaria; la otra, clave alternativa
Tipo de relación 1:1 recursiva con participación opcional en un lado	Elegir una de las opciones anteriores (clave ajena o nueva relación)
Tipo de relación M:N	Relación (tabla) con 2 claves ajenas: una para cada relación correspondiente a cada tipo de entidad conectado ▶ La clave primaria es la combinación de las claves ajenas, o quizá haya que añadir un atributo (si error de diseño concep.)
Tipo de relación n-aria	Relación (tabla) con n claves ajenas: una para cada relación correspondiente a cada tipo de entidad conectado ▶ La clave primaria es la concatenación de las claves ajenas, o de algunas de ellas, o quizá haya que añadir un atributo (si error de diseño conceptual)
Atributo multivalorado	Relación (tabla) con el atributo y con una clave ajena hacia la relación correspondiente al tipo de entidad al que pertenece el atributo ▶ La clave primaria es la combinación de la clave ajena y el atributo, o bien sólo el atributo

Resumen de traducción MERE → MR

95

Elemento del MERE	Elemento(s) del Modelo Relacional en el que se traduce
Jerarquía Disjunta y Total	Una relación (tabla) por cada combinación tipo/subtipo + Restricción de integridad general (aserto) para garantizar la disyunción
Jerarquía Disjunta y Parcial	Una relación para el supertipo Una relación para <u>cada</u> subtipo con... <ul style="list-style-type: none"> ▶ ... una clave ajena que referencia a la relación supertipo ▶ La clave primaria es esa clave ajena
Jerarquía Solapada y Total	Una sola relación (tabla) con atributo(s) discriminante(s) + Restricciones de comprobación
Jerarquía Solapada y Parcial	2 Relaciones (tablas): una para el supertipo y otra para <u>todos</u> los subtipos; ésta última contiene... <ul style="list-style-type: none"> ▶ Uno o más discriminantes ▶ Restricciones de comprobación ▶ Clave ajena que referencia a la relación del supertipo ▶ La clave primaria es dicha clave ajena

2. Validar las relaciones contra las transacciones de usuario

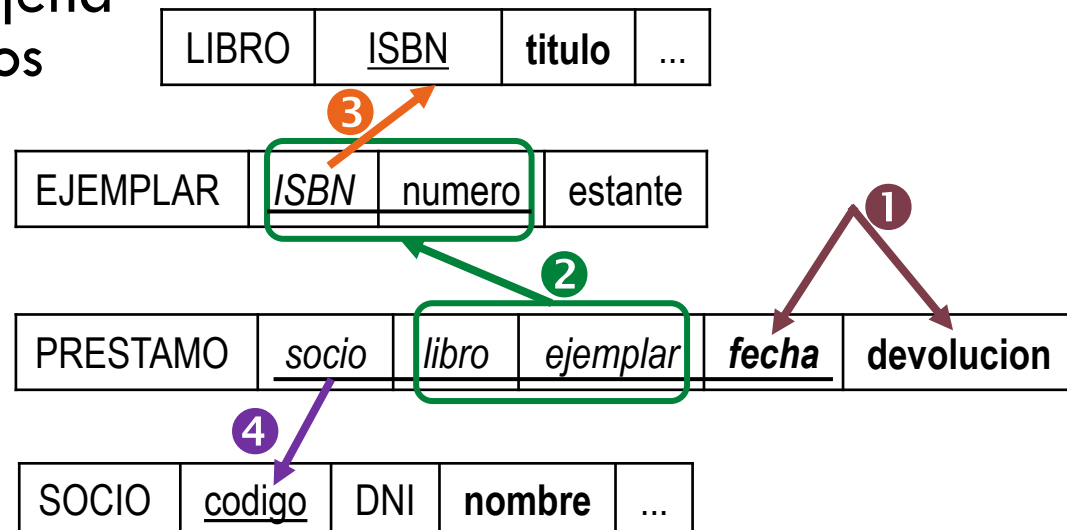
96

- Objetivo: asegurar que el **esquema lógico (EL)** **soporta** las **transacciones** requeridas
- “Ejecutar manualmente” las transacciones usando las relaciones (tablas), vínculos clave-primaria/clave ajena y el diccionario de datos
- Si alguna no se puede resolver
 - ▣ Se ha cometido algún **error** en el DC o en la traducción al MR
 - ▣ Retroceder y comprobar a qué áreas del EC accede la transacción para **identificar** y **resolver** el problema

...

C6. Listar detalles de los préstamos en curso (título del libro, nombre del socio y fechas).

...



3. Revisar las restricciones de integridad

97

- ❑ **Objetivo: evitar la pérdida de semántica inherente a la traducción** Esquema Conceptual → Esquema Lógico
- ❑ Asegurar que se ha documentado toda restricción de integridad necesaria para **impedir que la BD** pueda llegar a estar **incompleta, imprecisa o incoherente**
- ❑ Comprobar estos tipos de Restricciones de Integridad
 1. Datos requeridos u obligatorios
 2. Restricciones de dominio de atributos
 3. Integridad de entidad
 4. Integridad referencial (claves ajenas)
 5. Restricciones generales

Revisión
de lo que
ya hemos
hecho

3.1. Datos requeridos

98

- Comprobar que los **atributos de tipos de entidad** que **no admiten NULL** en el Esquema Conceptual se han traducido correctamente a atributos de relaciones (tablas) que no admiten NULL
 - ▣ Ídem para los atributos que sí lo admiten
- Comprobar que los **atributos de tipos de relación** han sido traducidos correctamente, de forma que admiten nulos o no en función de las cardinalidades mínimas
- Revisar el campo “**Admite nulos**” de las fichas de descripción correspondientes



SOCIO(codigo,DNI,nombre,telefono,penalizado,fin_pena)

Admite nulos: DNI, fin_pena

...



3.2. Restricciones de dominio

99

- Recordatorio: en el paso 3 del Diseño Conceptual se documentó el **tipo de datos y longitud** de cada atributo
- Para algunos de ellos pueden existir **restricciones que afectan a sus valores legales**
 - ▣ Restricciones que pudieron identificarse en el Diseño Conceptual
 - ▣ Ejemplos
 - “penalizado” de SOCIO sólo permite los valores ‘SÍ’ y ‘NO’
 - “num_premios” de AUTOR, permite n°s sin decimales entre 0 y 99
 - “num_copias” de LIBRO, admite números iguales o superiores a 1
 - “salario” de EMPLEADO, admite números con dos decimales superiores o iguales a 600,00
- Ahora deben expresarse en el Esquema Lógico de datos

3.2. Restricciones de dominio

100

- Indicar estas restricciones en el campo “**Comprobar**” de las fichas de descripción correspondientes



SOCIO(codigo,DNI,nombre,telefono,penalizado,fin_pena)

Admite nulos: DNI, fin_pena

Clave primaria: codigo

Clave alternativa: DNI

Valor por defecto calculado:

fin_pena = PRESTAMO.devolucion+10 días

Comprobar: penalizado IN ('SI', 'NO')



- Una restricción de comprobación debe ser **cierta para todas y cada una de las tuplas de la relación**



3.3. Integridad de Entidad

101

- ❑ Verificar que ningún atributo componente de una **clave primaria** de cualquier relación (tabla) **admite el nulo**
- ❑ Comprobar cada **clave alternativa** para asegurar que se indicó correctamente si puede o no contener nulo
- ❑ Revisar el campo “**Admite nulos**” de las fichas de descripción correspondientes

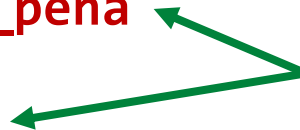


SOCIO(codigo,DNI,nombre,telefono,penalizado,fin_pena)

Admite nulos: DNI, fin_pena

Clave primaria: codigo

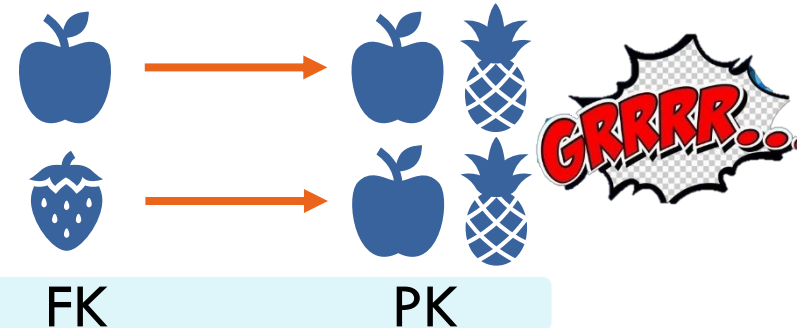
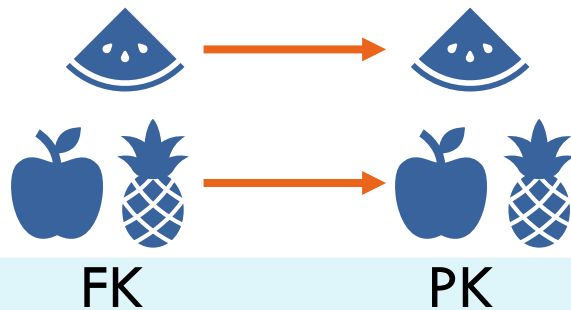
Clave alternativa: **DNI**



3.4. Integridad Referencial

102

- Comprobar que **cada clave ajena** está bien definida
 - ▣ Debe tener **tantos atributos como la clave primaria** a la que referencia
 - Si la clave primaria es compuesta entonces la clave ajena también debe serlo y tener el mismo número de atributos



- ▣ Confirmar que está bien indicado si **admite nulos o no**
 - Con base en la cardinalidad mínima del tipo de entidad hijo en la relación que se ha traducido a la clave ajena
- Revisar cada campo “**Clave ajena**” de las fichas de descripción correspondientes



3.5. Restricciones Generales (Asertos)

103

- Comprobar si las **restricciones de integridad generales** incluidas en los requisitos, o de forma textual en el EC o Diccionario de datos, **ya están redactadas** utilizando como vocabulario los nombres de **relaciones y atributos del esquema lógico**
- Si no es así, **definir los asertos** necesarios
 - ▣ *Ejemplo: Impedir que un socio penalizado tenga préstamos en curso (todos deben ser pasados)*
ASERTO RI_socio_penalizado COMPROBAR_QUE (
 NO EXISTE (una tupla en SOCIO
 donde “penalizado”=‘SÍ’
 y el valor de “código” ESTÉ_ENTRE
 (valores de “socio”
 de tuplas en PRESTAMO
 donde “devolucion”>=fecha actual))));

4. Validar el esquema lógico con los usuarios

104

- ❑ Objetivo: Asegurar que el **esquema lógico estándar (ELS)** y el **diccionario de datos** están **completos y documentados**
- ❑ Hay que confirmar que es así, repasándolos con los usuarios para que verifiquen que lo consideran una **representación verdadera** de los requisitos de datos
 - ▣ Si no les satisface, hay que volver atrás y repetir algunos pasos para modificarlo convenientemente



6.3 Diseño Lógico Específico

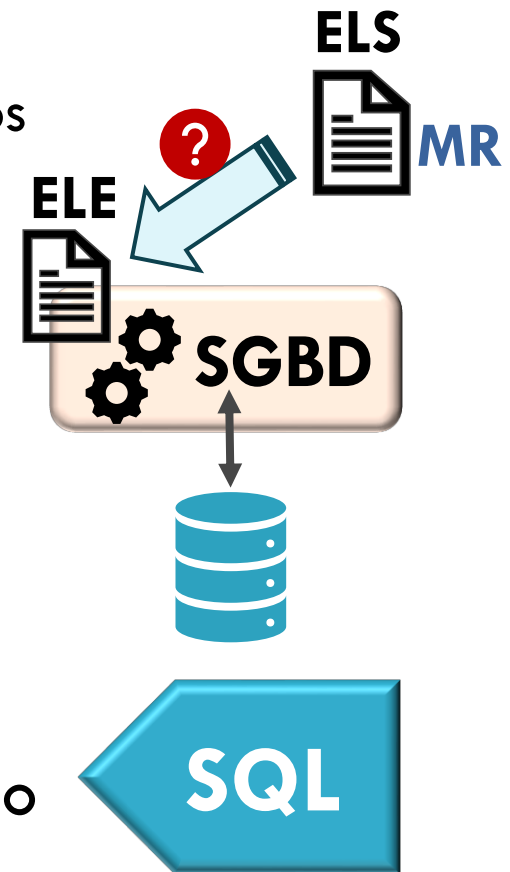
Especificación del Esquema Lógico en un **SGBD real**

105

- Ya tenemos el **Esquema Lógico eStándar** definido en el Modelo Relacional
 - ▣ Cada **relación** descrita en una **ficha**, y los **asertos** y resto de información en el Diccionario de Datos
- Y ha sido elegido el **SGBD Relacional comercial** para la implementación
 - ▣ Oracle, MySQL, PostgreSQL, etc.



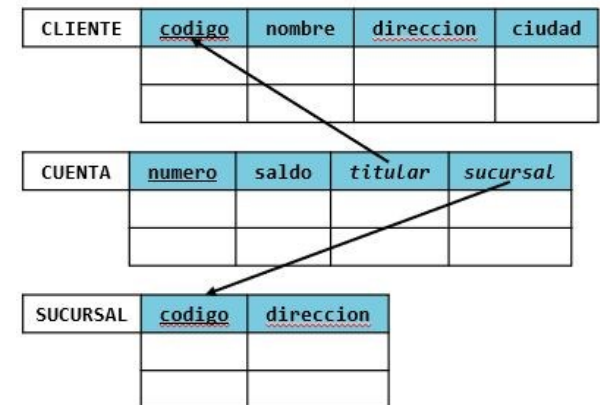
- ¿Cómo creamos las tablas en la BD?
- Pues hay que especificar el Esquema Lógico con la sintaxis propia del modelo de datos específico del SGBD: el ELE



Especificación del Esquema Lógico en un SGBD real

106

- ❑ Los SGBD relacionales comerciales (*Oracle, MySQL, MariaDB, MS Access, SQL Server, PostgreSQL, etc.*) ofrecen un dialecto particular del **lenguaje SQL**
- ❑ El **SQL permite crear** esquemas lógicos de datos compuestos por una **colección de tablas vinculadas entre sí mediante claves ajenas**
- ❑ En el tema siguiente estudiaremos el **estándar ANSI SQL**, y en prácticas programaremos con **Oracle SQL**
 - ▣ Utilizaremos sentencias **CREATE TABLE** para definir/crear cada una de las tablas componentes de un esquema lógico de BD



```
CREATE TABLE CUENTA (
  numero NUMERIC(20),
  saldo   NUMERIC(9,2),
  titular  CHAR(12),
  sucursal CHAR(4),
  PRIMARY KEY(numero),
  FOREIGN KEY(titular)
    REFERENCES CLIENTE(DNI)...,
  [ ... ] );
```

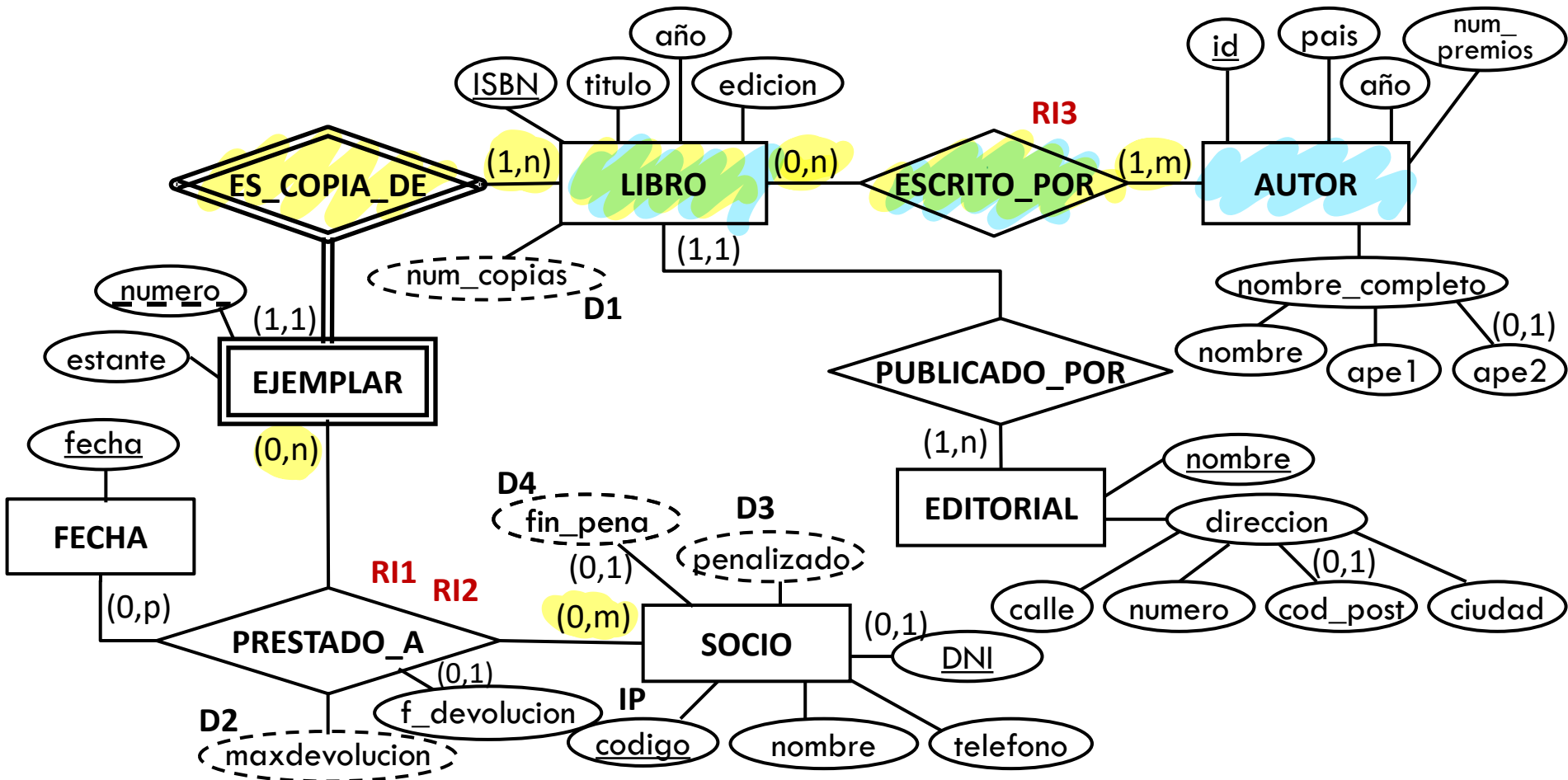
Ejemplos

- ❑ Esquema Lógico Estándar del ejemplo
- ❑ Más ejemplos de Diseño Lógico

Esquema Conceptual de partida

108

□ Diagrama Entidad-Relación (en el MER)



Esquema Lógico Estándar obtenido

109

EDITORIAL (nombre, calle, numero, cod_post, ciudad)

Admiten NULL: codpost

Clave primaria: nombre

LIBRO (ISBN, titulo, año, edicion, num_copias, *editorial*)

Admiten NULL: Ninguno

Clave primaria: ISBN

Clave ajena: editorial **Referencia_a** EDITORIAL(nombre)

Derivado: num_copias [para cada valor 'i' de ISBN contar tuplas de EJEMPLAR cuyo valor de "ISBN" es igual a 'i']

EJEMPLAR (numero, estante, *ISBN*)

Admiten NULL: Ninguno

Clave primaria: (ISBN, numero)

Clave ajena: ISBN **Referencia_a** LIBRO(ISBN)

Esquema Lógico Estándar obtenido

110

SOCIO (codigo, nombre, telefono, DNI, penalizado, fin_pena)

Admiten NULL: DNI, fin_pena

Clave primaria: codigo

Clave alternativa: DNI

Atributo Derivado:

fin_pena = PRESTAMO.devolucion + 10

...

Comprobar: penalizado IN ('SI', 'NO')

PRESTAMO (*libro, ejemplar, socio, fecha, maxdevolucion, f_devolucion*)

Admiten NULL: f_devolucion

Clave primaria: (libro, ejemplar, socio, fecha)

Clave ajena: (libro, ejemplar) **Referencia_a** EJEMPLAR(ISBN, numero)

Clave ajena: socio **Referencia_a** SOCIO(codigo)

Atributo Derivado: maxdevolucion = fecha + 15

Esquema Lógico Estándar obtenido

111

AUTOR (id, nombre, apellido1, apellido2, año_nacim, pais, num_premios)

Admiten NULL: apellido2

Clave primaria: id

Comprobar: (num_premios>0 AND num_premios<99)

ESCRITO_POR (libro, autor)

Admiten NULL: Ninguno

Clave primaria: (libro, autor)

Clave ajena: libro **Referencia_a** LIBRO(ISBN)

Clave ajena: autor **Referencia_a** AUTOR(id)

Esquema Lógico Estándar obtenido

112

Restricciones surgidas de la Traducción

ASERTO RI_editorial_publica_libros

COMPROBAR_QUE (

NO_EXISTE (una tupla en EDITORIAL
donde el valor de “nombre” NO_ESTÉ_ENTRE
(valores de “editorial” en LIBRO)));

ASERTO RI_libro_tiene_copias

COMPROBAR_QUE (

NO_EXISTE (una tupla en LIBRO
donde el valor de “ISBN” NO_ESTÉ_ENTRE
(valores de “ISBN” en EJEMPLAR)));

ASERTO RI_autor_escribe_libros

COMPROBAR_QUE (

NO_EXISTE (una tupla de AUTOR
donde el valor de “id” NO_ESTÉ_ENTRE
(valores de “autor” en ESCRITO_POR)));

Esquema Lógico Estándar obtenido

113

Restricciones detectadas en el Diseño Conceptual

ASERTO RI_socio_penalizado COMPROBAR_QUE

(**NO EXISTE** (una tupla en SOCIO
donde “penalizado”=‘SÍ’
y el valor de “codigo” ESTÉ_ENTRE
(valores de “socio”
de tuplas en PRESTAMO
donde “f_devolucion” ES_NULL
y “maxdevolucion”>=fecha_actual)));

Un socio
penalizado no
puede tener
préstamos activos

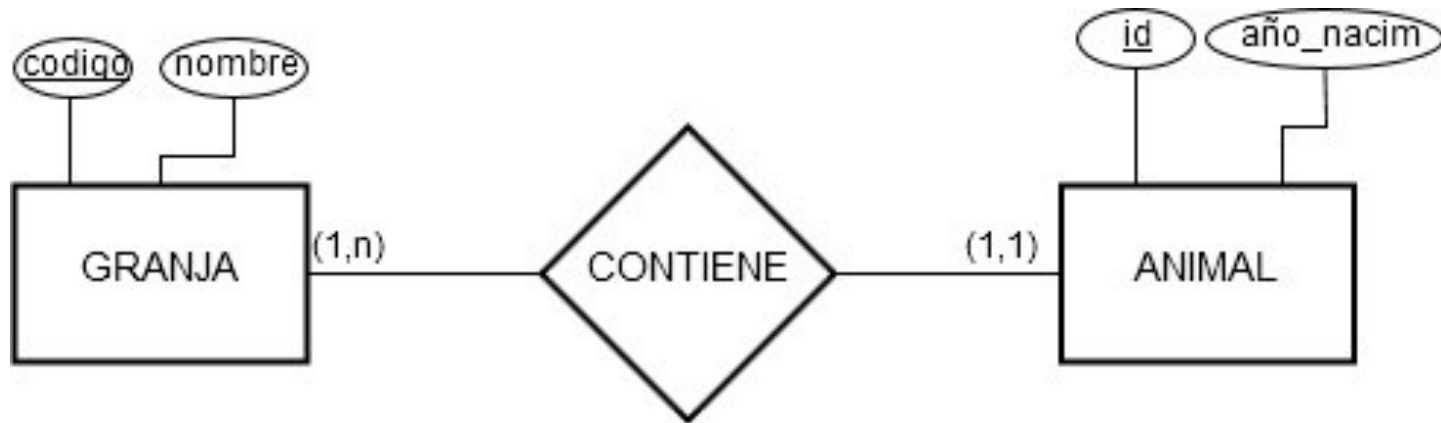
ASERTO RI_socio_maximo_prestamos COMPROBAR_QUE

(4 >= MÁXIMO(CUENTA PARA_TODA_TUPLA_DE SOCIO S
(tuplas en PRESTAMO
donde “socio” = S.codigo
y “f_devolucion” ES_NULL
y “maxdevolucion”>=fecha_actual)));

Un socio puede
tener activos un
máximo de 4
préstamos

Ejemplos de diseño lógico

114



GRANJA (codigo, nombre)

Admiten NULL: ninguno

Clave primaria: codigo

ANIMAL (id, año_nacim, granja)

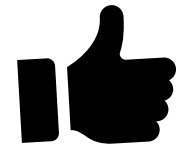
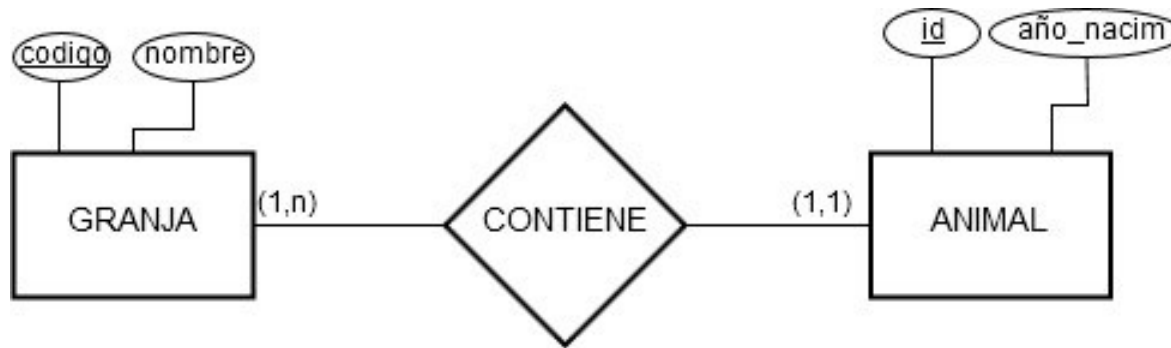
Admiten NULL: ninguno

Clave primaria: id

Clave ajena: granja Referencia_a GRANJA(codigo)

Ejemplos de diseño lógico

115



GRANJA

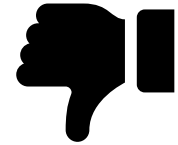
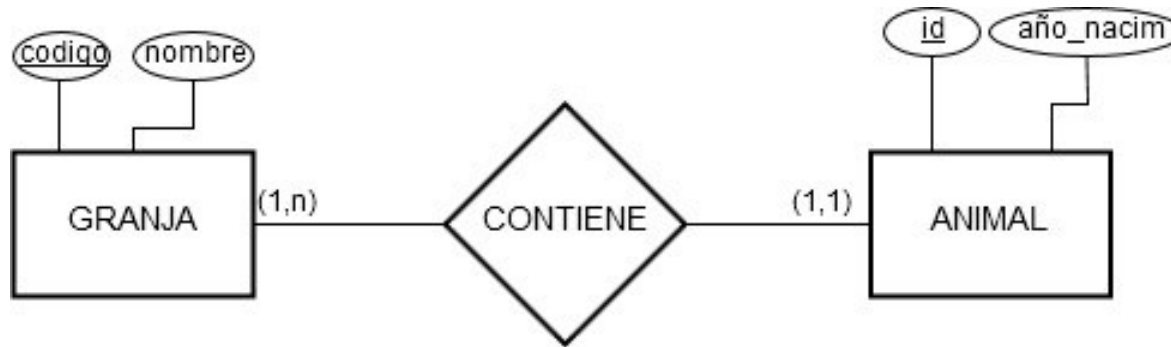
<u>codigo</u>	nombre
g1	Los pioneros
g2	La solana
g3	El girasol

ANIMAL

<u>id</u>	año_nacim	granja
a1	2013	g1
a2	2016	g1
a3	2014	g2
a4	2016	g3
a5	2018	g2

Ejemplos de diseño lógico

116



GRANJA

<u>codigo</u>	nombre	animal
g1	Los pioneros	a1, a2
g2	La solana	a3, a5
g3	El girasol	a4

KO!

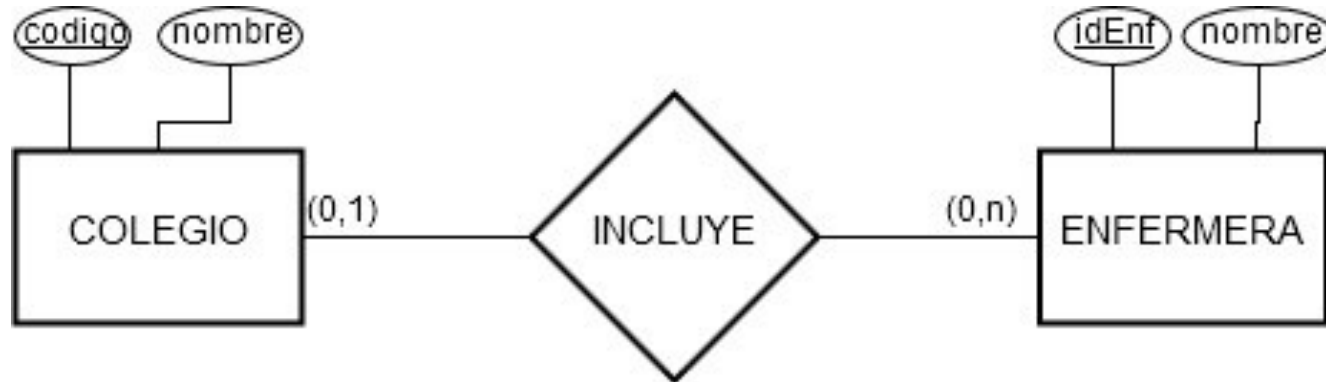
ERROR

ANIMAL

<u>id</u>	año_nacim
a1	2013
a2	2016
a3	2014
a4	2016
a5	2018

Ejemplos de diseño lógico

117



COLEGIO(codigo, nombre, enfermera)

Admiten NULL: enfermera

Clave primaria: codigo

Clave ajena: enfermera

Referencia_a ENFERMERA(idEnf)

ENFERMERA (idEnf, nombre)

Admiten NULL: ninguno

Clave primaria: DNI

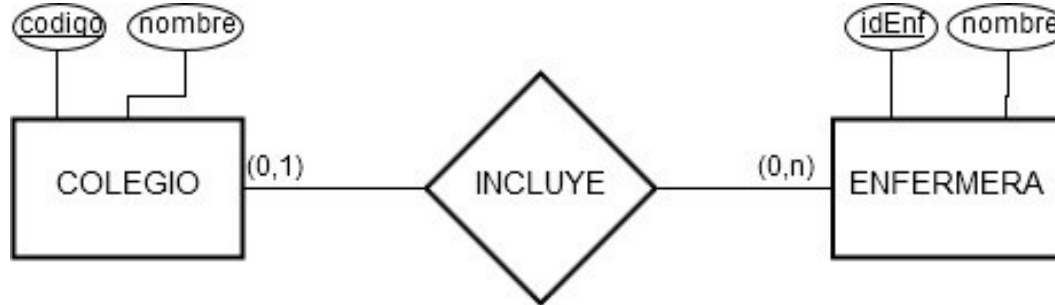
*La mayoría de los colegios tienen una enfermera.
Hay colegios sin enfermera.
No todas las enfermeras están asignadas a colegios.*

❗ La cardinalidad mínima 0 ENFERMERA (tipo de entidad padre) no afecta

❗ La cardinalidad mínima 0 de COLEGIO (tipo de entidad hijo) hace que la clave ajena pueda contener NULL

Ejemplos de diseño lógico

118



COLEGIO

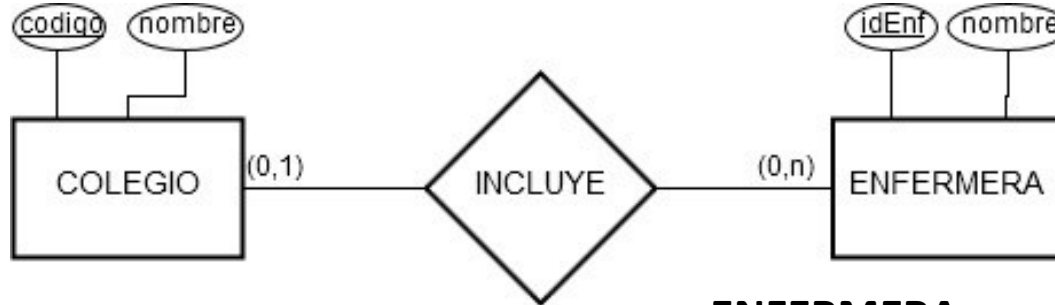
<u>codigo</u>	nombre	enfermera
c1	CEEIP Giner de los ríos	null
c2	CEEIP Antonio Machado	enf3
c3	CEEIP Rosalía de Castro	enf1
c4	CEEIP María Montessori	enf3

ENFERMERA

<u>idEnf</u>	nombre
enf1	Ana Ayala
enf2	Juan Jara
enf3	Luisa Lira
enf4	Paula Pina

Ejemplos de diseño lógico

119



COLEGIO

<u>codigo</u>	nombre
c1	CEEIP Giner de los ríos
c2	CEEIP Antonio Machado
c3	CEEIP Rosalía de Castro
c4	CEEIP María Montessori

ENFERMERA

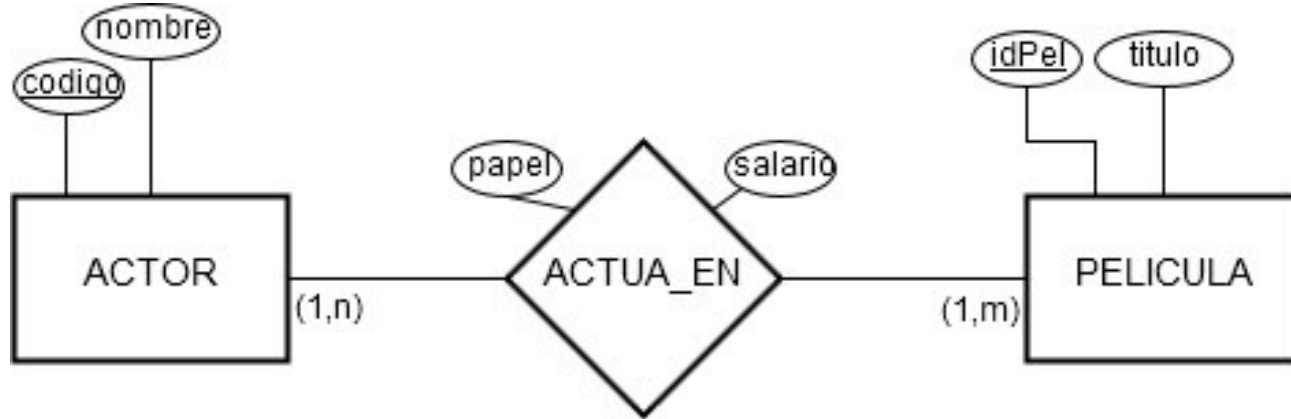
<u>idEnf</u>	nombre	colegio
enf1	Ana Ayala	c3
enf2	Juan Jara	NULL
enf3	Luisa Lira	c2, c4
enf4	Paula Pina	NULL

KO!

ERROR

Ejemplos de diseño lógico

120



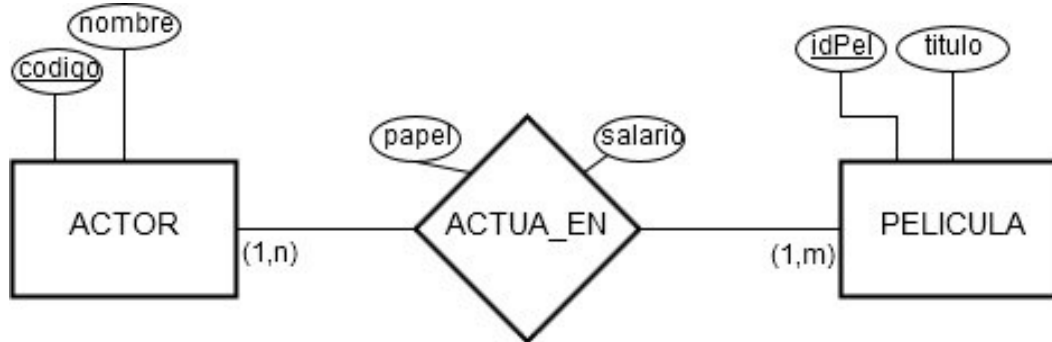
ACTOR(codigo, nombre)
Admiten NULL: ninguno
Clave primaria: codigo

PELICULA(idPel, titulo)
Admiten NULL: ninguno
Clave primaria: idPel

ACTUA_EN(actor, pelicula, papel, salario)
Admiten NULL: ninguno
Clave primaria: (actor, pelicula)
Clave ajena: actor
Referencia_a ACTOR(codigo)
Clave ajena: pelicula
Referencia_a PELICULA(idPel)

Ejemplos de diseño lógico

121



ACTOR

<u>codigo</u>	nombre
a1	Richard Armitage
a2	Sandra Bullock
a3	Melissa McCarthy
a4	Bill Murray

<u>actor</u>	<u>película</u>	papel	salario
a1	p1	secundario	350000
a2	p1	principal	600000
a2	p2	principal	400000
a3	p3	secundaria	200000
a4	p3	principal	600000

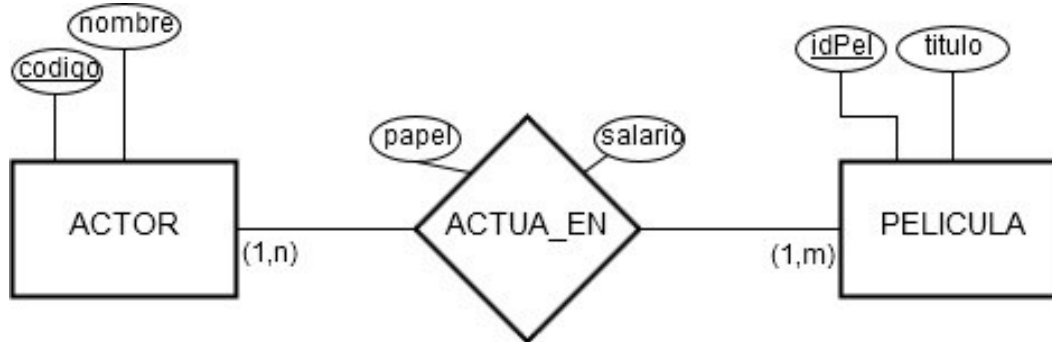
ACTUA_EN

<u>idPel</u>	titulo
p1	Ocean's 8
p2	Gravity
p3	St. Vincent

PELICULA

Ejemplos de diseño lógico

122



ACTOR

<u>codigo</u>	nombre
a1	Richard Armitage
a2	Sandra Bullock
a3	Melissa McCarthy
a4	Bill Murray

PELICULA

<u>idPel</u>	titulo	actor
p1	Ocean's 8	a1, a2
p2	Gravity	a2
p3	St. Vincent	a3, a4

KO!

ERROR