



Dpto. Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur

# ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2015

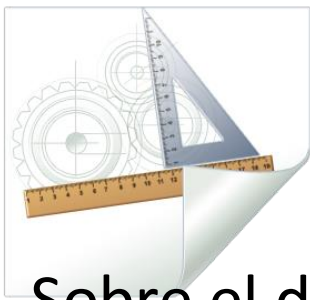
**Clase 4:**

**Decisiones de diseño**

**Pasaje a Tablas: modelo relacional**

Mg. María Mercedes Vitturini  
[mvitturi@uns.edu.ar]





# Diseño del Modelo de Datos

Sobre el diseño del esquema de Base de Datos de un problema:

1. Se consulta a usuarios y expertos del dominio para que planteen sus requerimientos y restricciones.
2. Enfocar la atención en *caracterizar los datos*.
3. Dependiendo del problema, el **modelo de datos puede ser más o menos complejo**.
  - Los modelos reales generalmente son grandes y complejos!
4. Un modelo se construye para iterar sobre él, analizarlo, discutir y corregir.
5. El modelo de datos debe ser tan general como para que no se requieran cambios frecuentes.



# Guía para construir un MER

1. Identificar las *entidades* y definir los conjuntos de entidades.
2. Identificar los *atributos* que describen las entidades.
3. Identificar las *relaciones* entre las entidades.
4. Identificar *atributos asociados con las relaciones* (si existen).
5. Definir la *cardinalidad* de las relaciones.
6. Identificar la oportunidad de *usar conceptos avanzados* (entidades débiles, generalización-especialización, agregación).

# Diseñando el MER

Algunas decisiones a tomar:

- ¿un objeto real se representa como un **atributo** o una **entidad**?
- ¿un concepto del mundo real se expresa mediante un **conjunto entidad** o un **conjunto relación**?
- ¿usar una **relación ternaria** o un descomponerla en **pares de relaciones binarias**?
- ¿es **conjunto entidad fuerte** o se trata de un **conjunto entidad débil**?
- ¿conviene usar **generalización-especialización**?
- ¿conviene usar **agregación**?

**RECORDAR:** se modelan entidades del negocio y sus relaciones (modelo de datos) no "acciones".

# Decisiones de Diseño

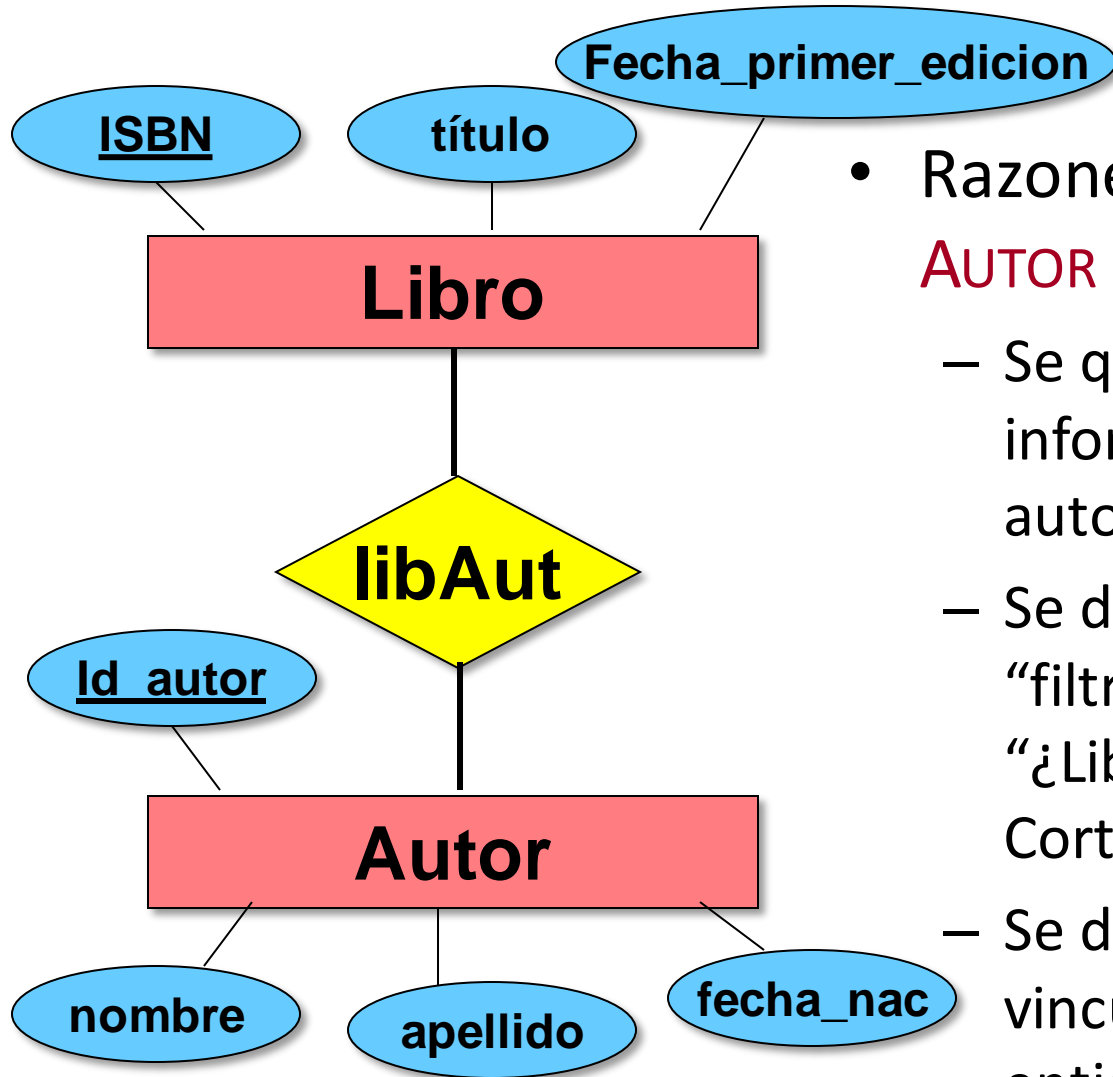
## *¿Atributo o Entidad?*

- Un aspecto importante en el Modelo E-R es distinguir cuándo “un dato” constituye un atributo y cuándo una entidad.
- **Depende del problema que se está modelando** y sus restricciones:
  - Relevancia de la información en el contexto del problema.
  - Información adicional que acompaña al atributo.
  - Si la información se vincula por relaciones con otras entidades.

# ¿Atributo o entidad?

- Ejemplos para discutir:
  1. Una compañía desea mantener información de los *departamentos* en que se organiza y sus *gerentes*.
  2. Una agencia de viajes ofrece distintos *viajes*, la *localidad* desde donde salen/arriban los viajes.
  3. El *número de teléfono* de un *cliente*.
  4. El *número de teléfono* de una *guía on-line*.
  5. Las *palabras claves* para un *libro*.

# Ejemplo



- Razones para modelar a **AUTOR** como entidad:
  - Se quiere mantener mucha información propia de cada autor.
  - Se desea poder “consultar” o “filtrar” información por autor. “¿Libros que escribió Cortázar?”
  - Se desea expandir el modelo y vincular Autor con una entidad **PREMIO**

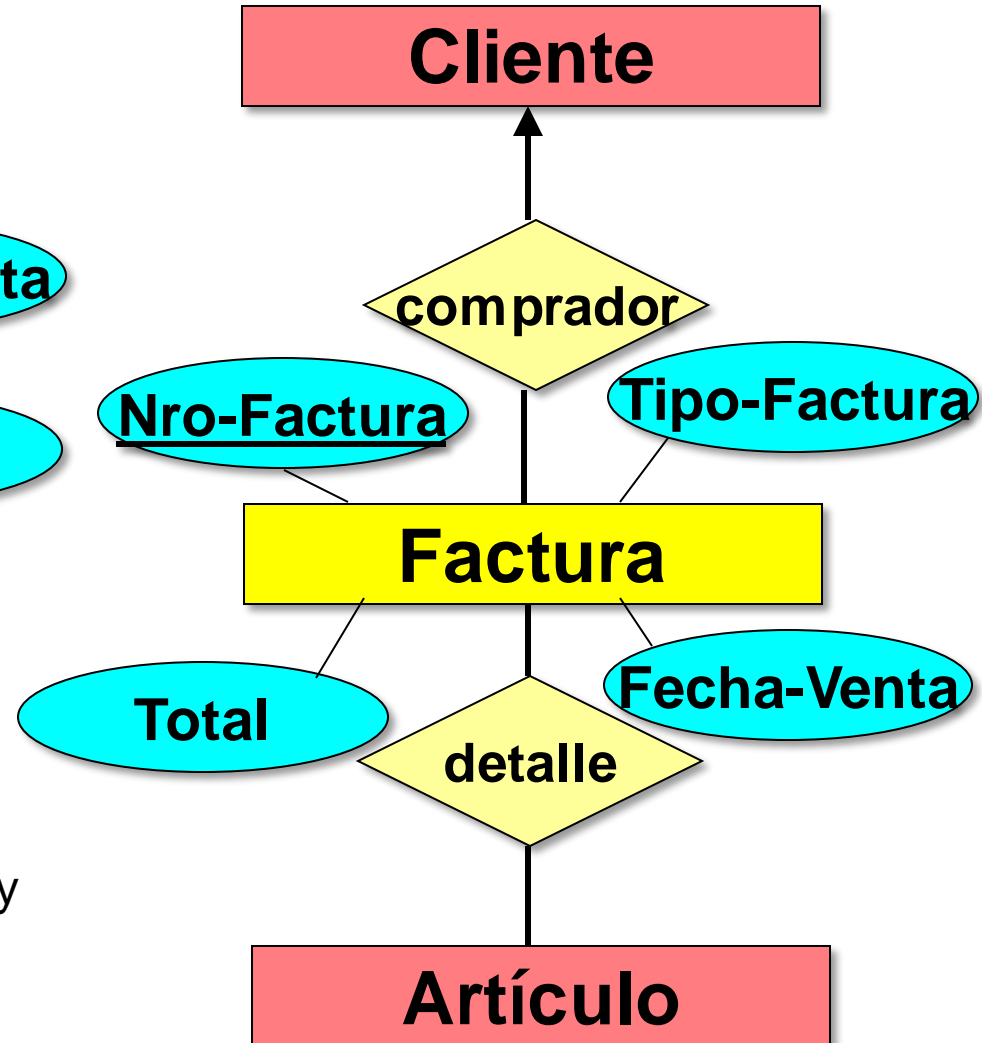
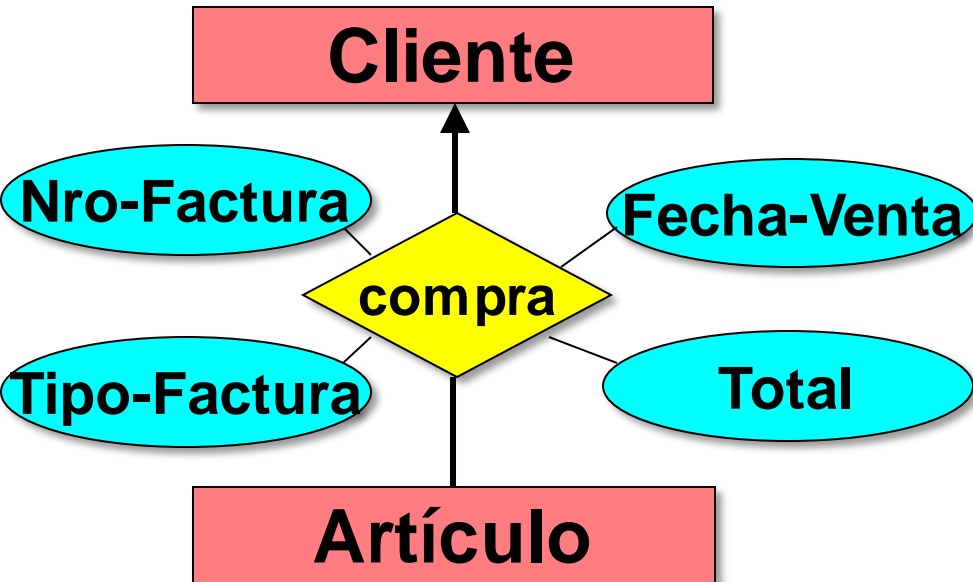
# ¿Usar una Entidad o una Relación?

## ¿Entidad o relación?

- No siempre está claro si es mejor expresar un objeto del mundo a modelar como un *conjunto entidad* o como un *conjunto relación*.
- Algunas consideraciones:
  - Atributos propios de la relación.
  - Cardinalidad de la relación (una a una, una a muchas, etc.) entre las relaciones.



# Entidad o Relación

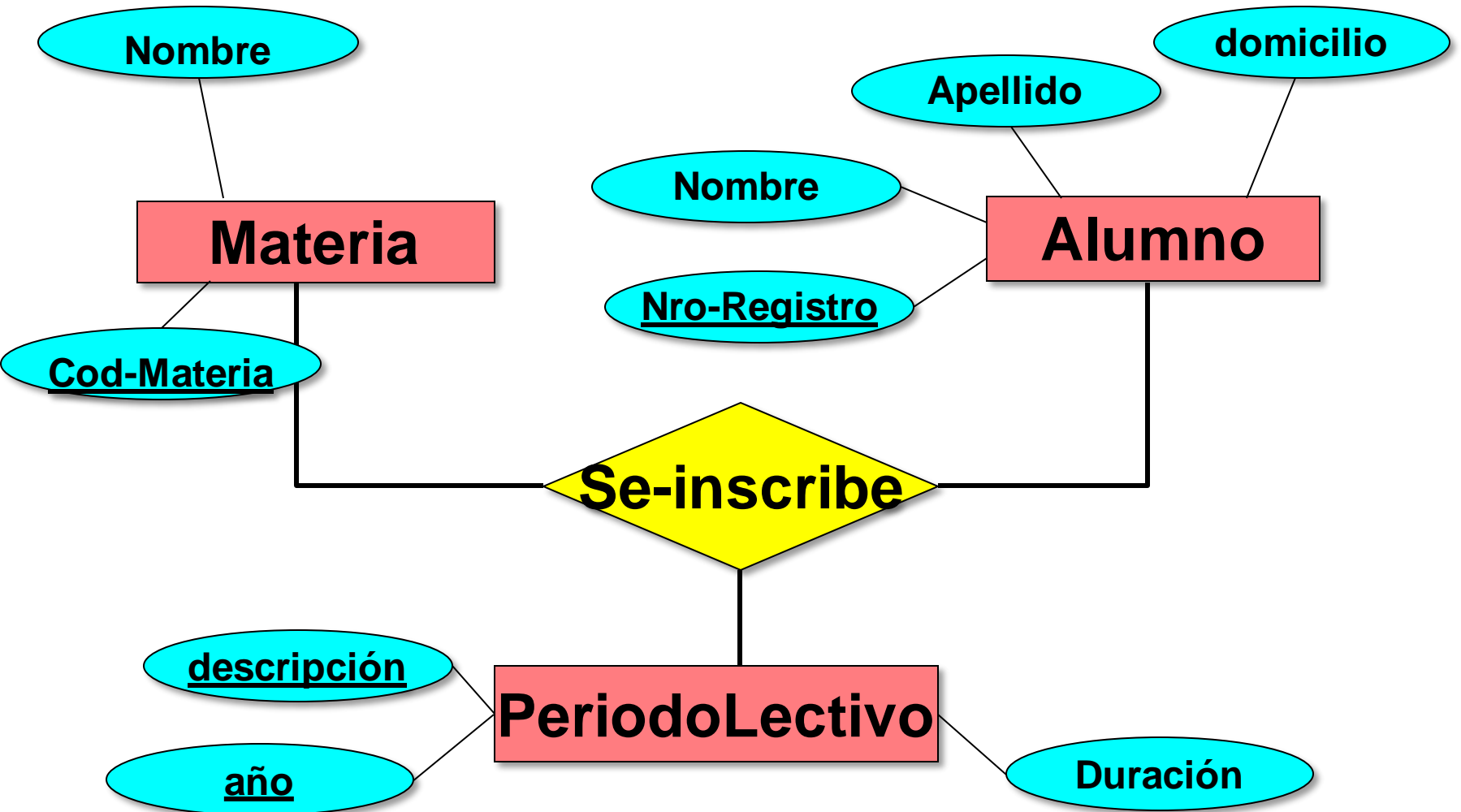


**Obs:** por simplicidad no se incluyeron atributos en las entidades **ARTÍCULOS** y **CLIENTES**

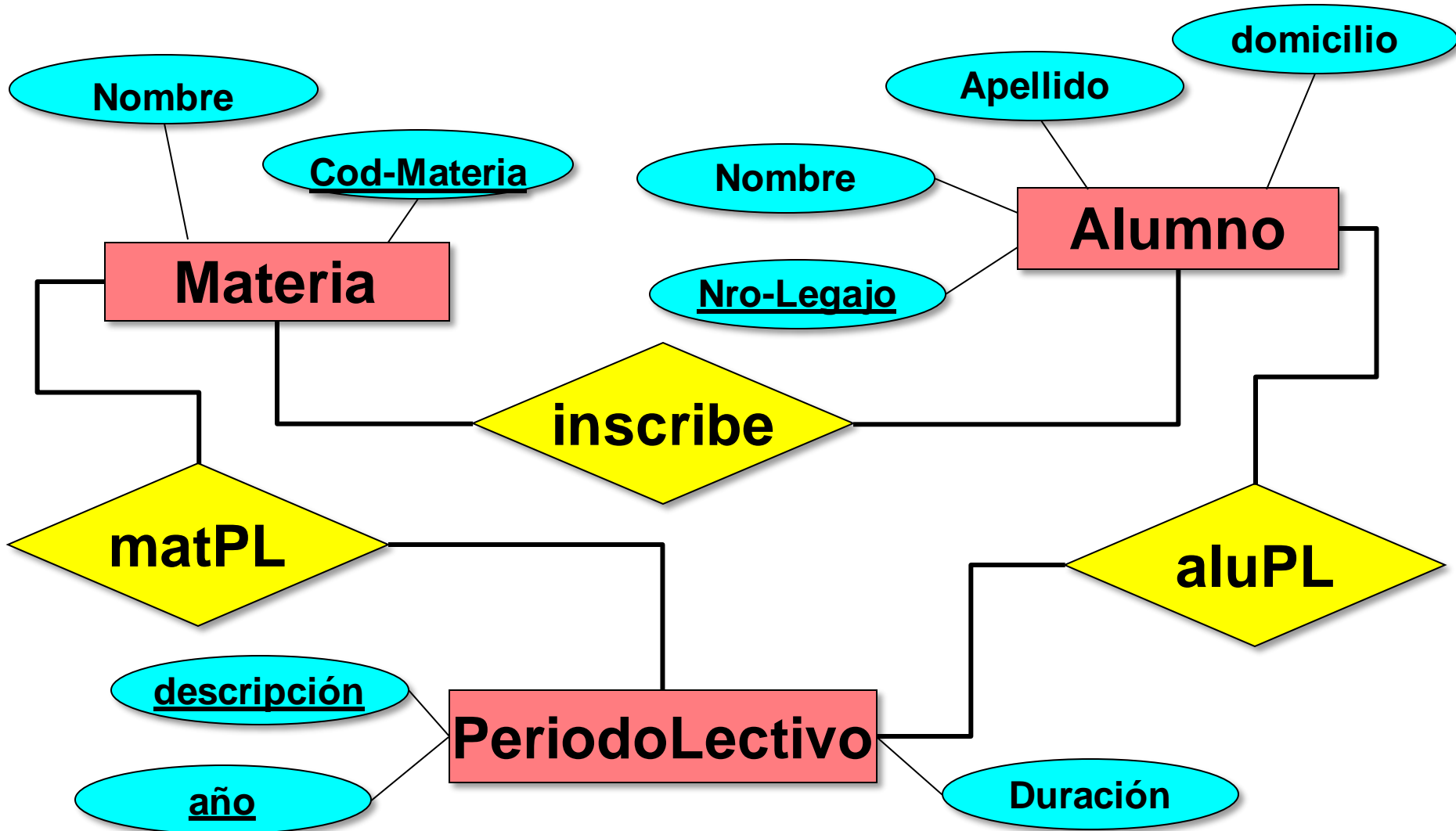
# ¿Relaciones ternarias o binarias ?

- Algunas relaciones que *parecen* mejor representadas como relaciones ternarias que binarias.
- Elegir la representación que *modelo mejor la semántica* del problema.
- Siempre se puede encontrar una representación usando relaciones binarias para las relaciones ternarias. *Depende del gusto del diseñador*.
- Lo *importante* es verificar que *el diseño represente los requerimientos del problema!*.

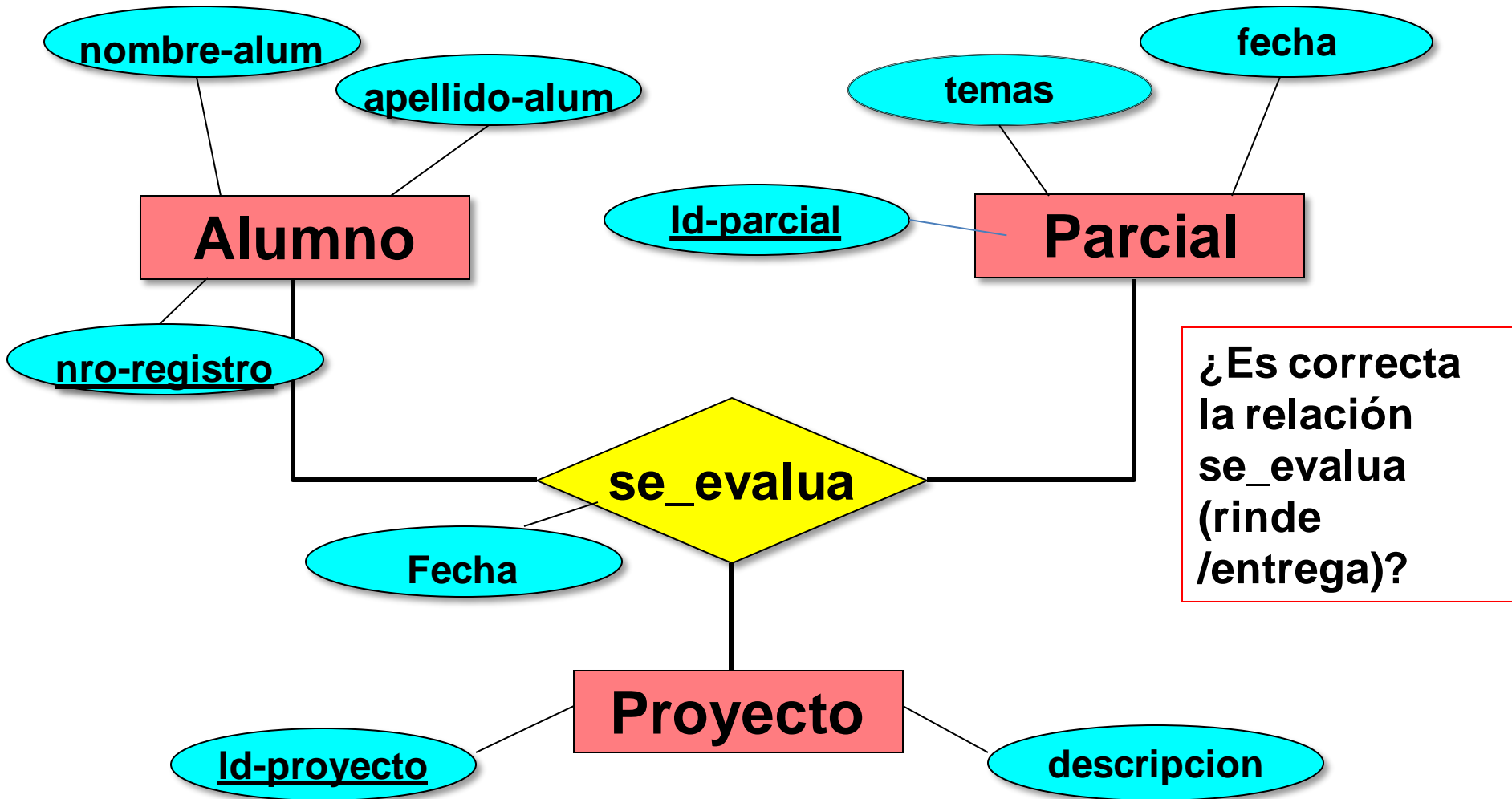
# Relaciones Ternaria – Ejemplo



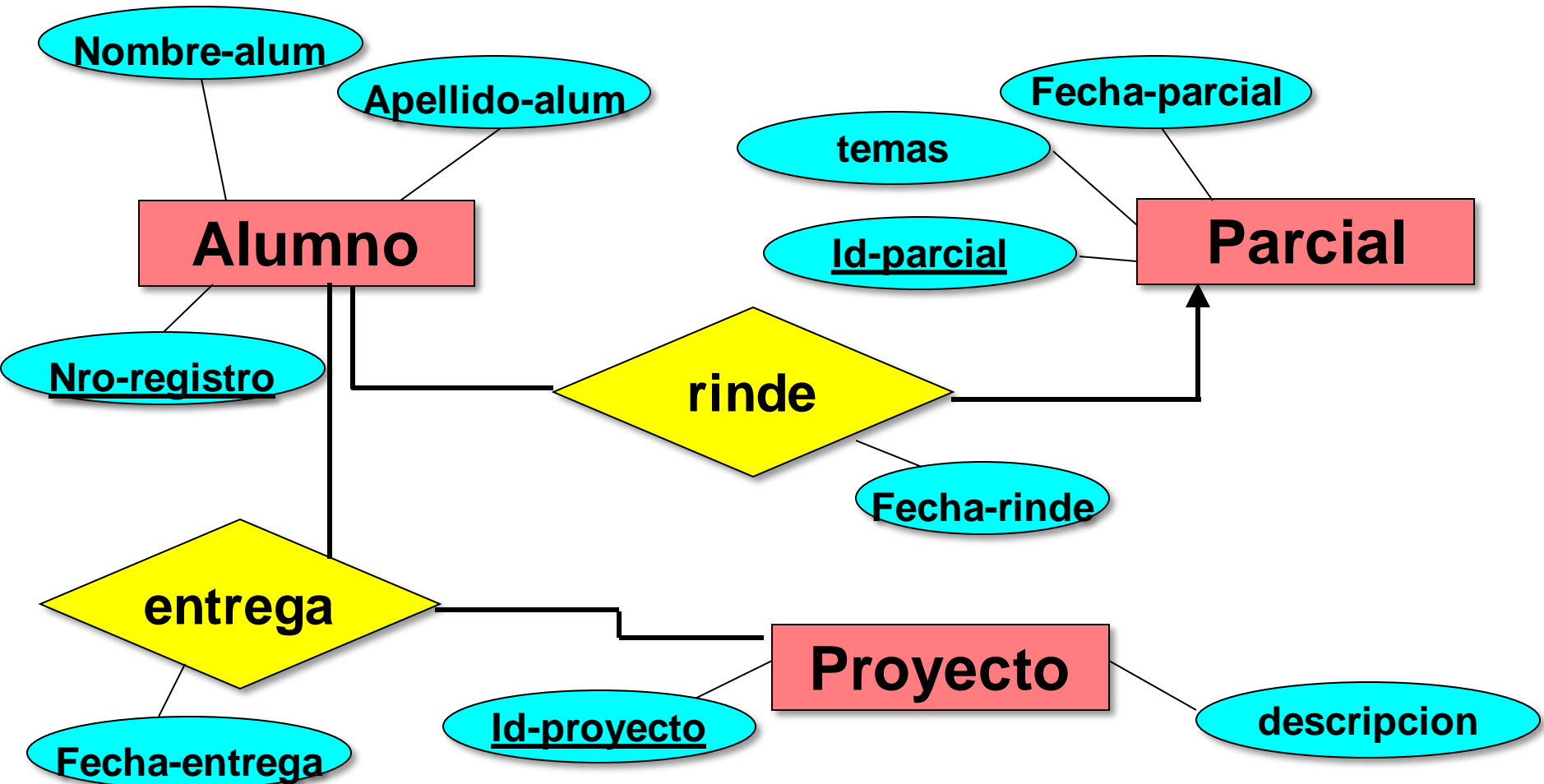
# Vs: Relaciones Binarias



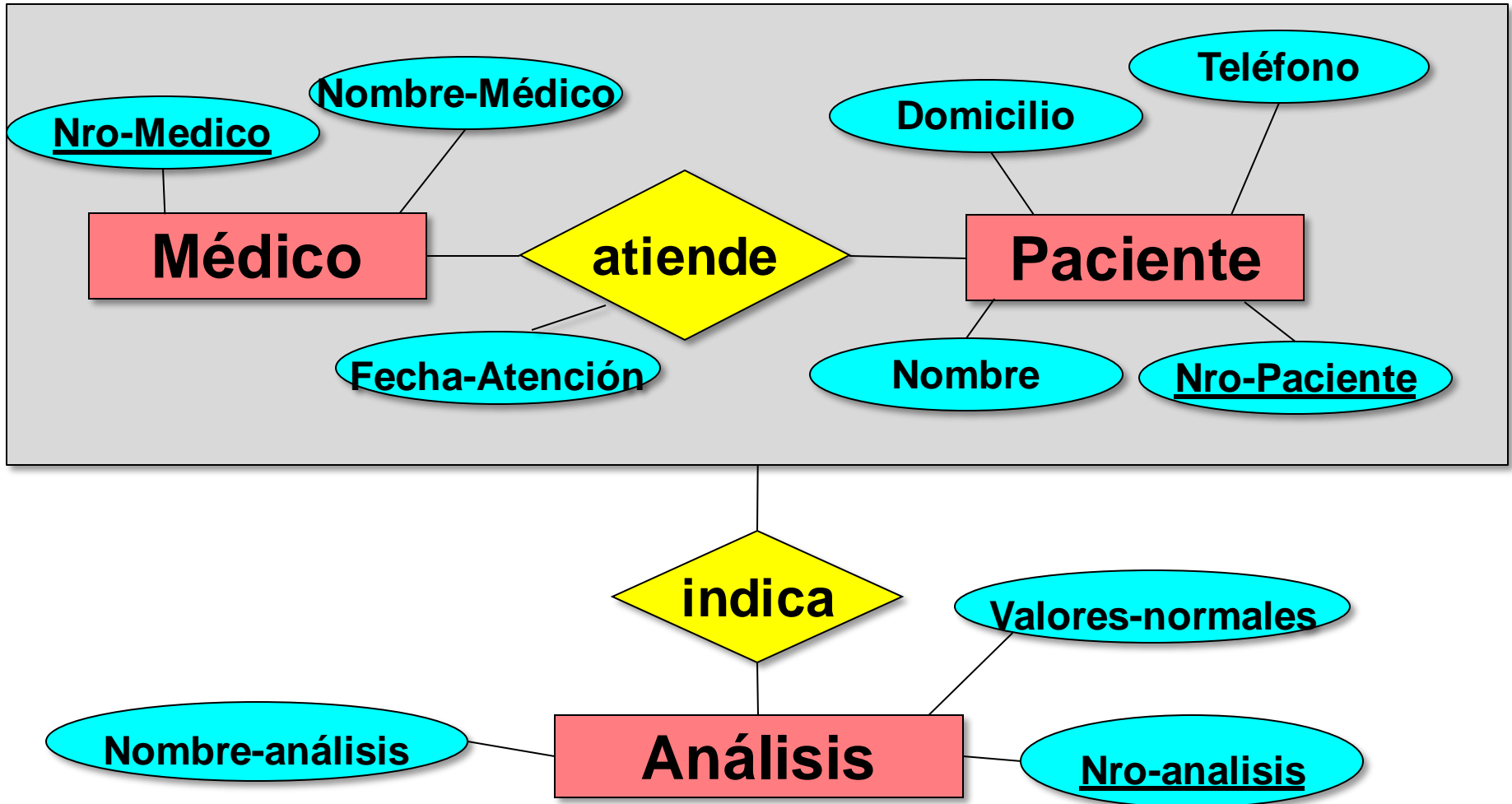
# Problemas con Relaciones no binarias

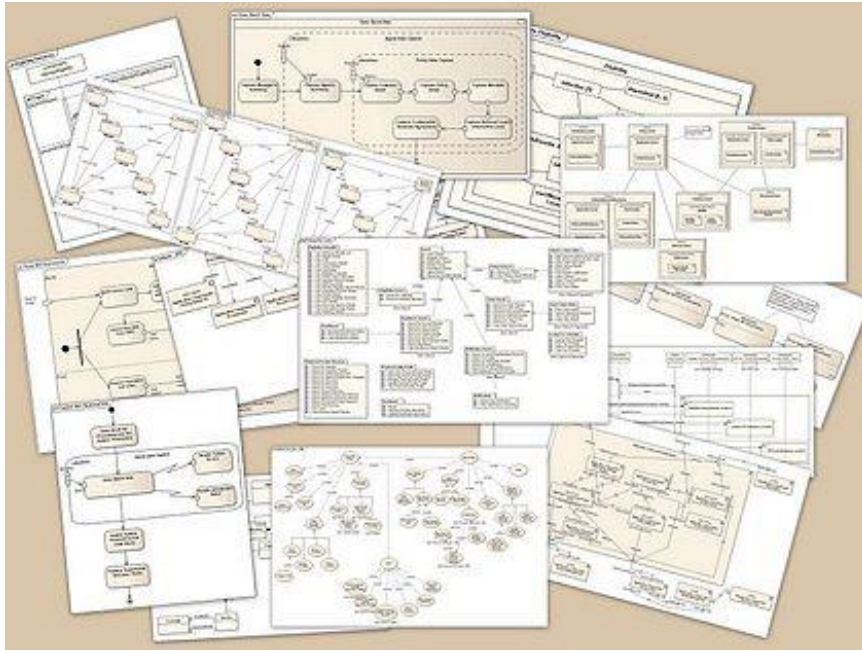


# Binarias vs. Ternaria



# Uso de Agregados





# UML y MER

**Correspondencia entre el diagrama de clases de UML**



UNIFIED MODELING LANGUAGE™





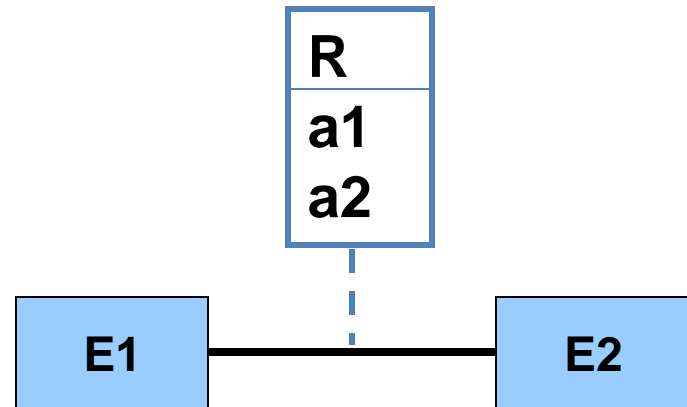
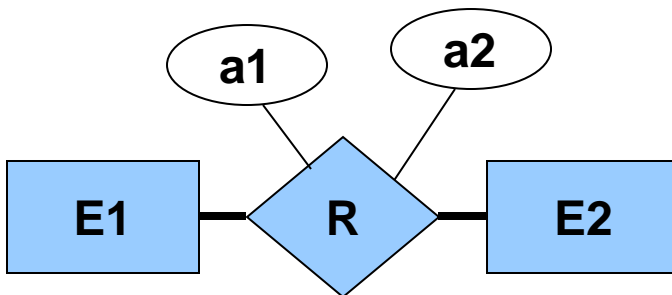
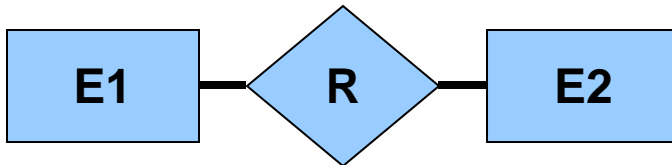
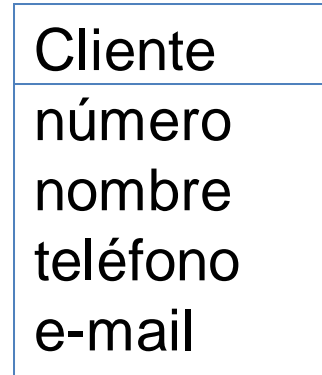
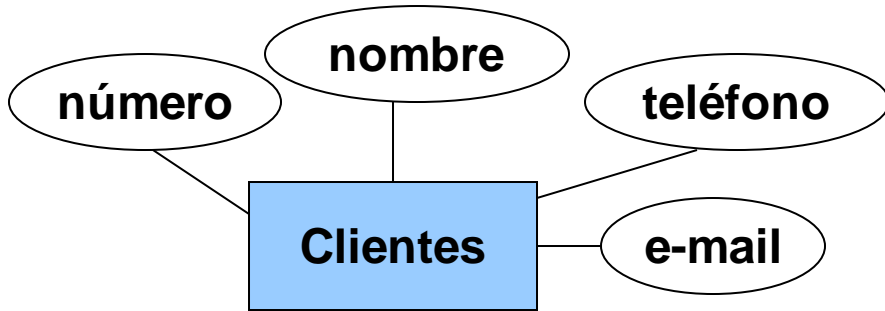
# UML

- **UML** (*Unified Modeling Language*) Lenguaje de Modelado Unificado
- UML provee varios componentes para modelar gráficamente diferentes aspectos de un sistema de software.
- El **Diagrama de Clases UML** en cierta forma se corresponde con el **Diagrama E-R**, aunque ambos modelos presentan algunas diferencias.

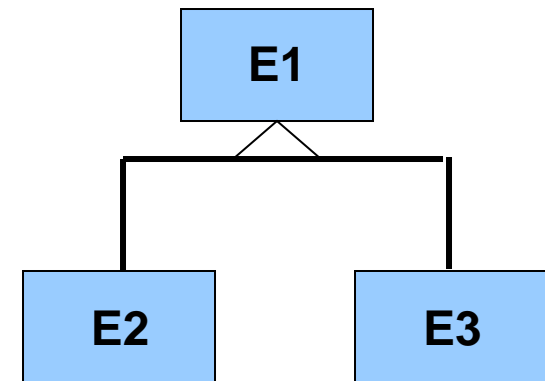
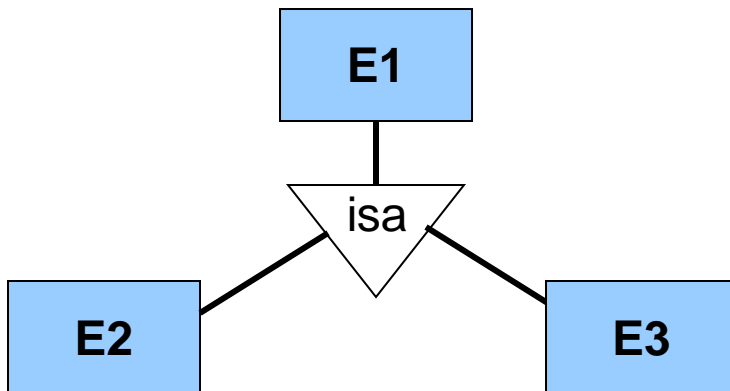
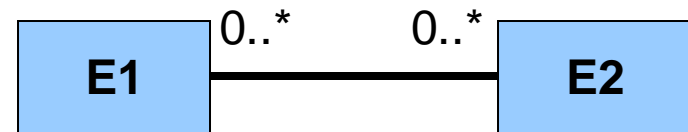
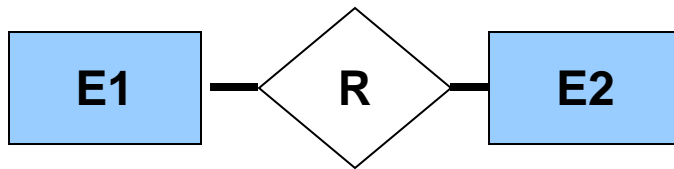
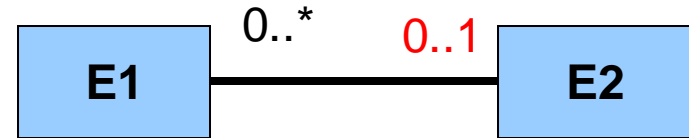
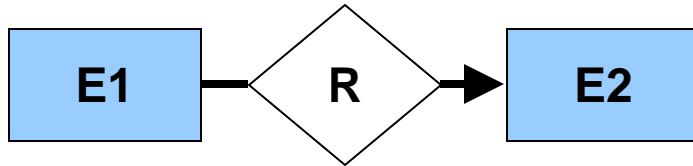
# UML vs. DER - Representación

- Los **conjuntos de entidades** (clases en UML) se representan como **Cajas** y los atributos están incluidos dentro de las cajas.
- Los **conjuntos de relaciones** (asociaciones en UML) se dibujan como una **línea** conectando clases.
- El nombre de la relación se escribe sobre la asociación.
- Las **relaciones con atributos** se dibujan como una **caja** con los atributos que se vincula con la asociación.
- Las **restricciones sobre cardinalidad** se especifican en la forma  $i..s$  donde  $i$  representa el límite inferior y  $s$  el límite superior.

# DER vs. UML



# DER vs. UML: Restricciones



# UML vs. DER

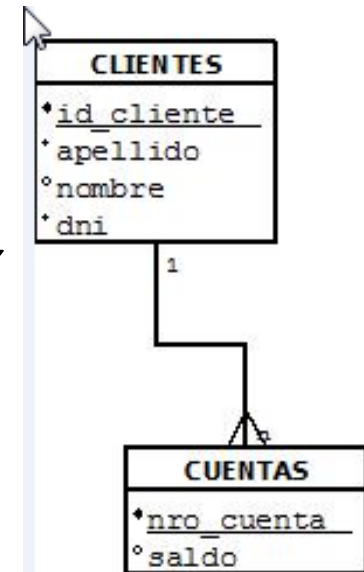
- Una comparación más exhaustiva se puede encontrar en el libro “Database Systems – The complete book” 2nd ed.– Molina, Ullman, Widom (capítulo 4)

UML	E/R Model
Class	Entity set
Association	Binary relationship
Association Class	Attributes on a relationship
Subclass	Isa hierarchy
Aggregation	Many-one relationship
Composition	Many-one relationship with referential integrity

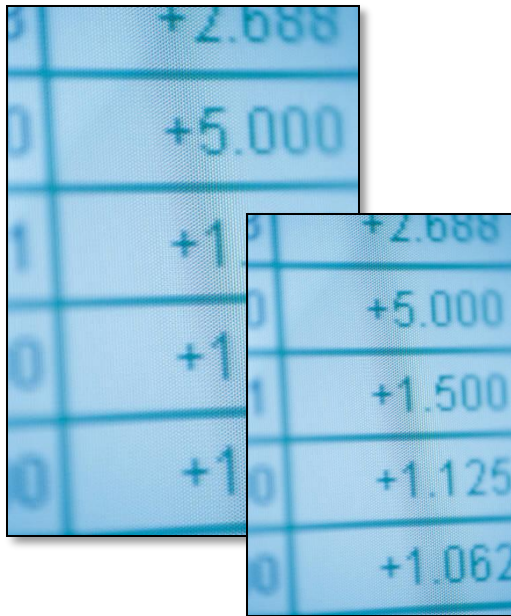
Figure 4.34: Comparison between UML and E/R terminology

# Notaciones Alternativas

- Existen notaciones alternativas.
- Queda propuesto como ejercicio investigar alternativas de notación así como editores gráficos para construir DER.



# Representación en Tablas



The image shows two overlapping tables with a light blue background and a grid pattern. The top table has the following visible data:

0	+2.688
0	+5.000
1	+1
0	+1
0	+1

The bottom table has the following visible data:

0	+2.688
0	+5.000
1	+1.500
0	+1.125
0	+1.062

**Dado un DER existe una representación en tablas que aproxima a la vista lógica del modelo de datos**

# Representación Tabular

- Una **modelo MER** se puede traducir **como un conjunto de *tablas***.
- Este proceso de conversión se conoce como *“pasaje a Tablas del DER”*.
- Vamos a estudiar:
  - Cómo se define el proceso de la conversión.
  - Como representar las restricciones del MER en las “tablas”.



# Representación como Tablas

- Entidades
  - Entidades Fuertes
  - Entidades débiles
  - Subentidades (is\_a)
- Relaciones
  - Relaciones sin atributos
    - M:N ; M:1 ; 1:1
  - Relaciones con atributos
  - IS\_A y Agregados



→ y para cada tabla definir los atributos que forma la llave primaria

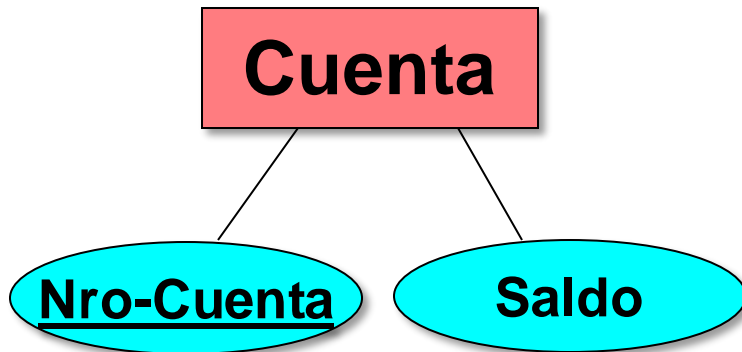
# Conjuntos de Entidades

- Sea  $E$  un conjunto de **entidades fuertes** con  $n$  atributos descriptivos  $a_1, a_2, \dots, a_n$ .

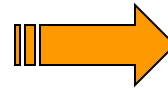
$$E(a_1, a_2, \dots, a_n)$$

- Una *tabla para*  $E$  constará de  $n$  columnas, una por cada atributo.
- Cada *fila* de la tabla representa a una *entidad* del conjunto entidad  $E$ .
- La llave primaria de  $E$  representa *la llave primaria de la tabla*.

# Entidad fuerte - Ejemplo



1- vista del ME-R



## Cuenta

<b>Nro-Cuenta</b>	<b>Saldo</b>
12345	120
12456	1000
20321	500
31124	2500
32100	170
54881	3000

Tabla para la entidad fuerte Cuentas.

**CUENTA (Nro-Cuenta, Saldo)**

La llave primaria es el número de cuenta.

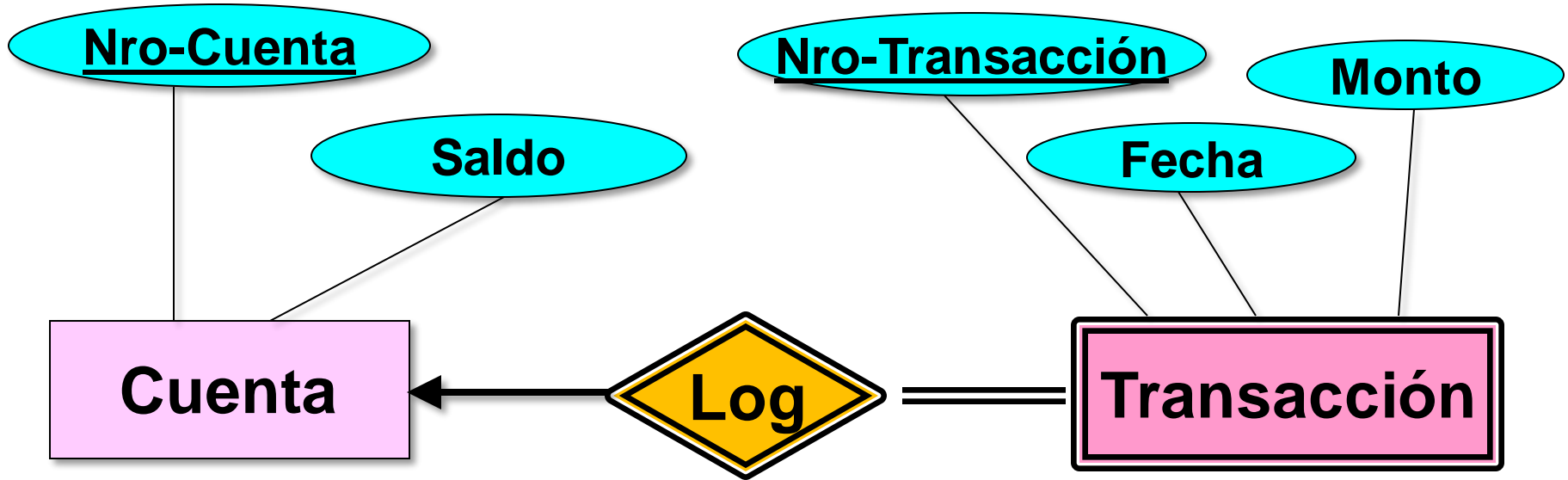
# Conjunto Entidad Débil

- Sea **A** un conjunto **entidad débil** con los atributos  $n$  descriptivos  $a_1, a_2, \dots, a_n$ .
- Sea **B** el conjunto **entidad fuerte** que domina a **A** con atributos  $b_1, b_2, \dots, b_m$ , y sea  $pk(b_1, b_2, \dots, b_j)$  la los atributos que forma la llave primaria para B-
- La **tabla para la entidad débil A** constará de  $n+j$  columnas, una para cada atributo del conjunto:

$$\{a_1, a_2, \dots, a_n\} \cup \{b_1, b_2, \dots, b_j\}$$

- La llave *primaria de la tabla para la entidad débil A* está formada por la  $pk(B) \cup$  discriminador (A)

# Entidades débiles



**Transacción**

<b>Nro-Cuenta</b>	<b>Nro-Transacción</b>	<b>Fecha</b>	<b>Monto</b>
1126	5	12/02/2001	100
1126	6	20/03/2001	-50
3270	5	14/05/2001	90
3270	6	02/06/2001	-600
1691	6	05/06/2001	1000
5501	7	15/07/2001	3000
7632	8	23/08/2001	-200

**TRANSACCIÓN**  
**(nro-cuenta, nro-transacción, fecha, monto)**

Tabla que representa a la entidad débil **Transacción**

# Conjuntos Relación

- Sea  $R$  un conjunto relación que vincula a los conjuntos entidad  $E_1, E_2, \dots, E_n$  ( $n > 1$ ), con  $pk(E_1), \dots, pk(E_n)$  las llaves primarias de  $E_1, E_2, \dots, E_n$  respectivamente.
- Si  $R$  no tiene atributos propios entonces se crea una tabla con una columna por cada elemento del conjunto:

$$pk(E_1) \cup \dots \cup pk(E_n)$$

- Si  $R$  tiene como atributos propios,  $\{a_1, \dots, a_m\}$  entonces se crea una tabla con una columna por cada elemento del conjunto:

$$pk(E_1) \cup \dots \cup pk(E_n) \cup \{a_1, \dots, a_m\}$$

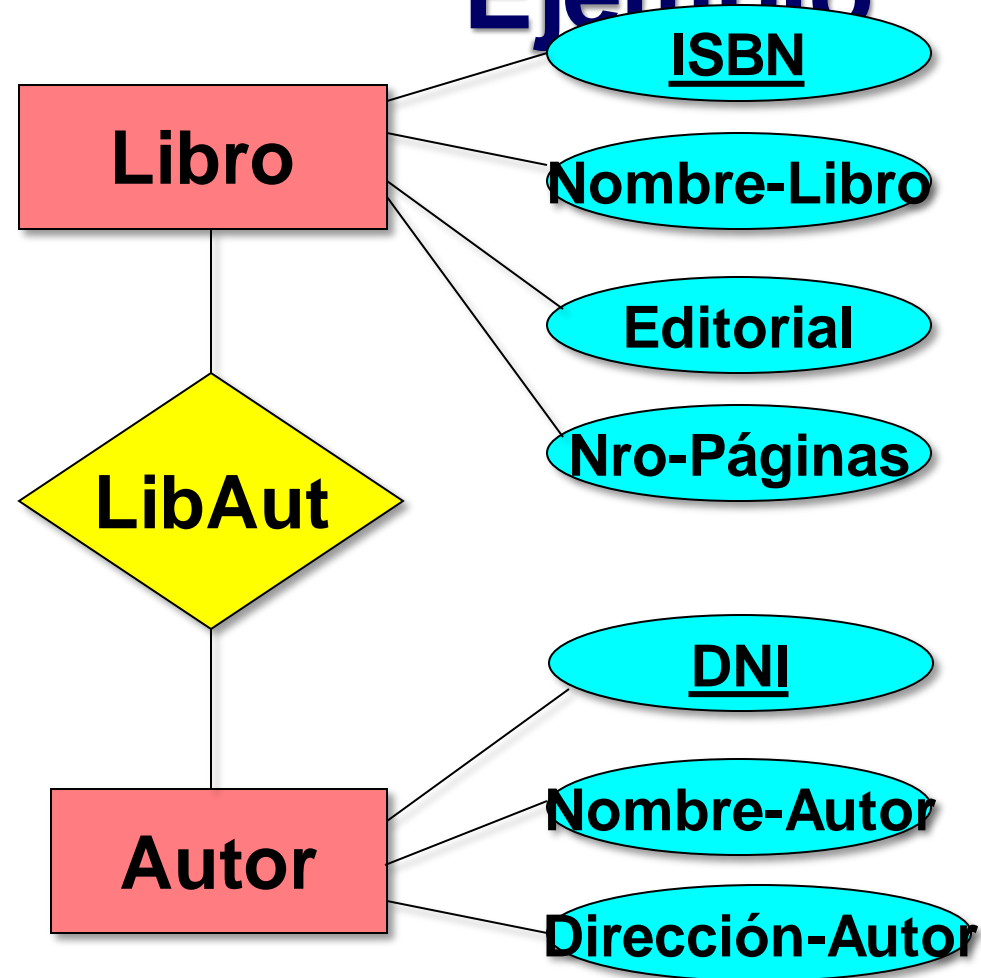
# Relaciones sin atributos - Ejemplo

**LibAut**

<b>ISBN</b>	<b>DNI</b>
84-481-0079-4	22222222
96-665-1238-4	32000123
12-236-9812-1	45678678
44-128-9365-8	20666999
51-326-4588-7	18987345
87-123-9699-0	5698772
71-439-2541-1	25006897

Tabla que representa a la relación  
**LibAut**

**LibAut** (ISBN, DNI)



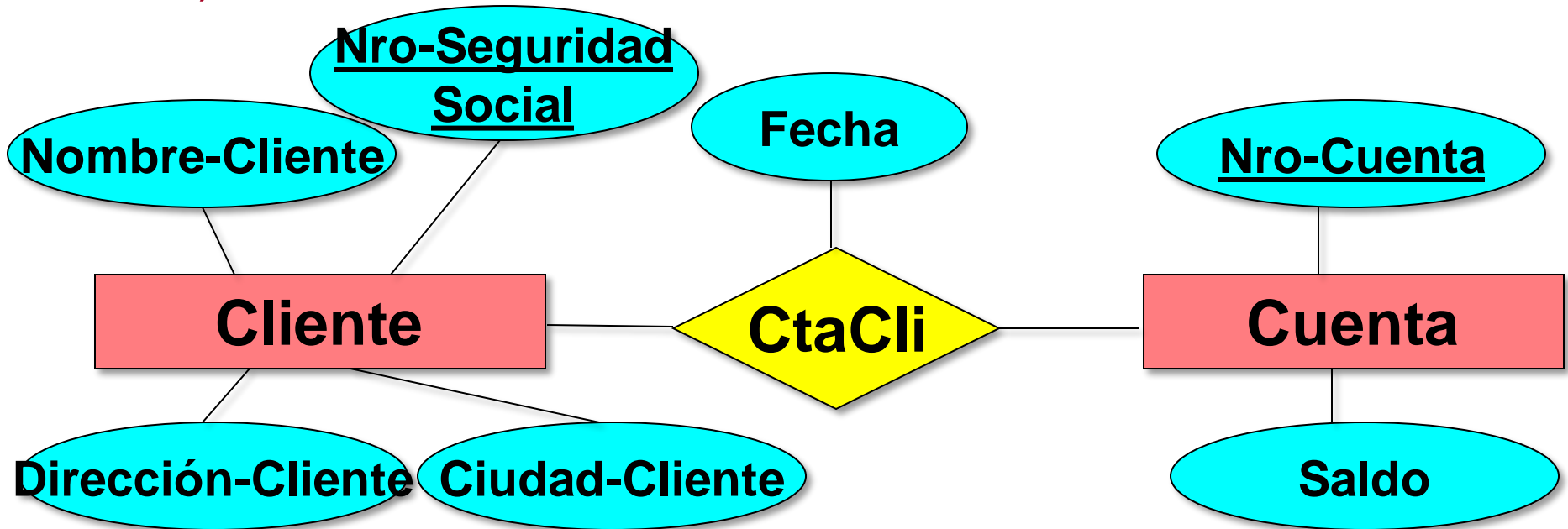
# Con Atributos

**CtaCli**  
(nro-seguridad-social, nro-cuenta, fecha)

**CtaCli**

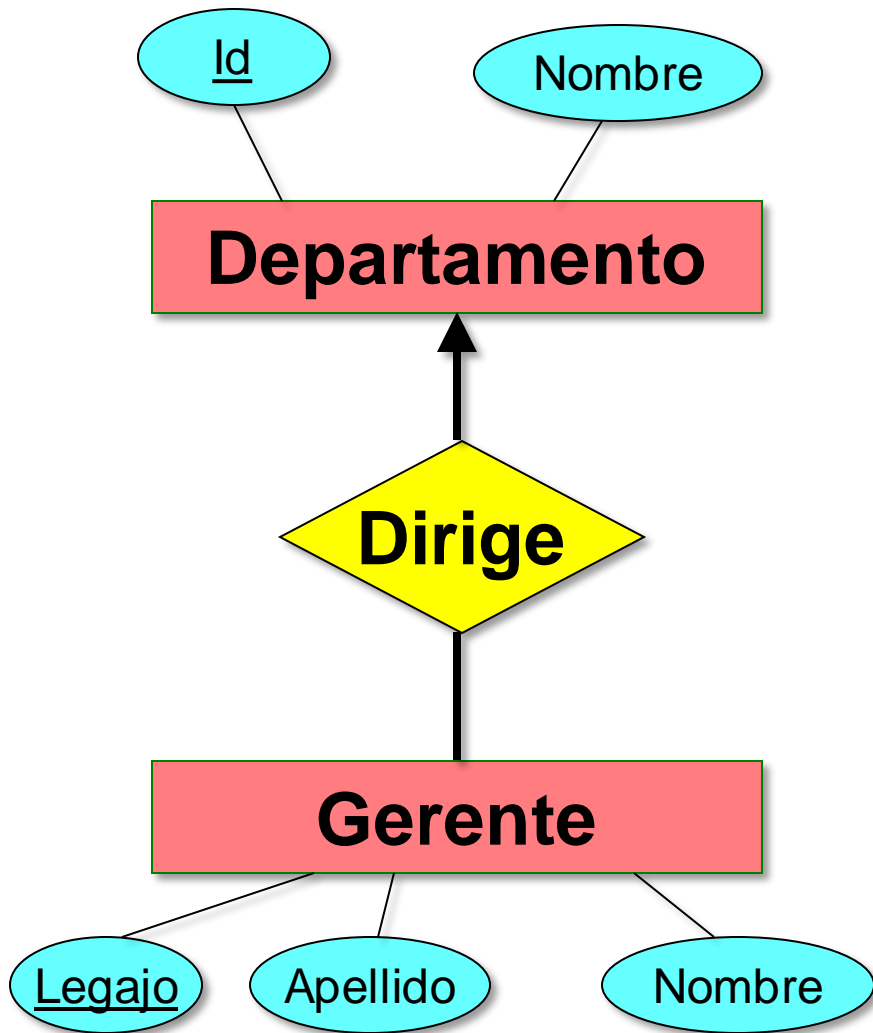
<b>Nro-Seguridad-Social</b>	<b>Nro-Cuenta</b>	<b>Fecha</b>
8448100794	1111	17/01/2003
9666512384	2222	24/01/2001
1223698121	3146	04/02/2002
4412893658	2222	12/02/2001
5132645887	5987	03/01/2003
8712396990	4329	18/03/2002
7143925411	7452	05/08/2003

Tabla que representa a la relación CtaCli





# Caso particular: Relaciones “1:m”

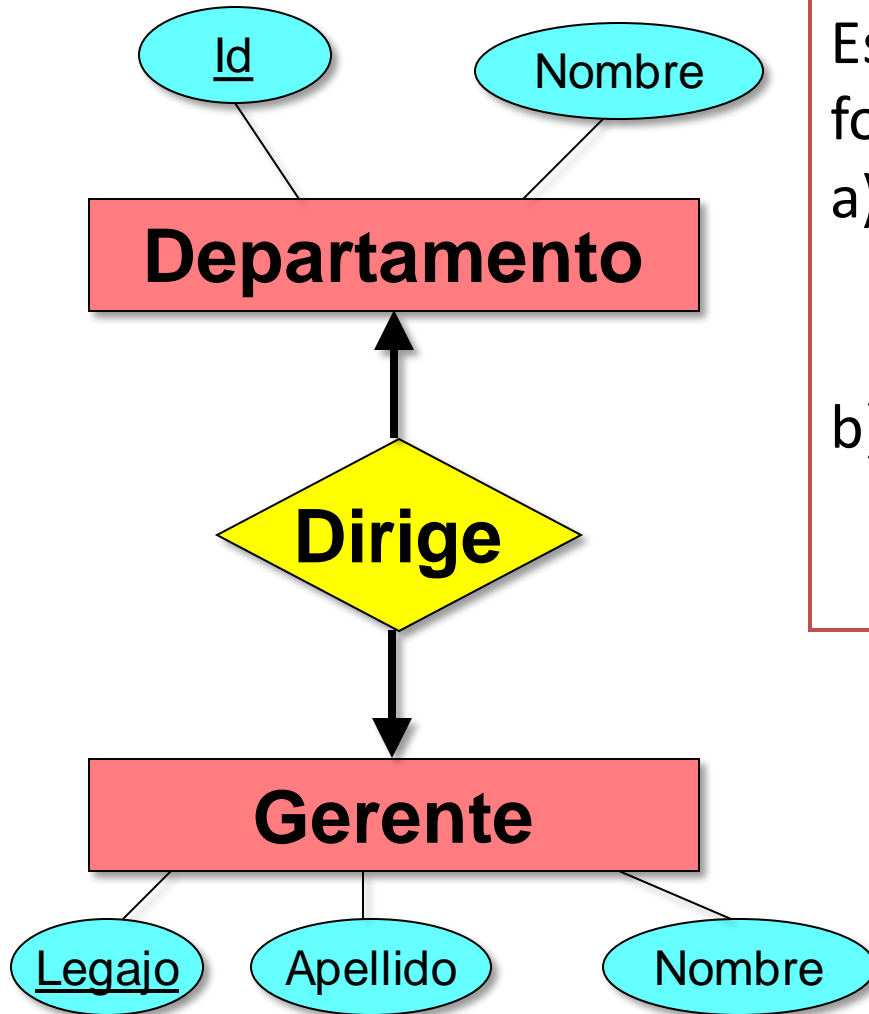


## Caso 1: relaciones m:1 sin atributos

La relación podría eliminarse:  
a) Agregando el atributo ID-Departamento en el conjunto de entidades Gerente (*del lado del muchos*).

**Gerente + Dirige:**  
**GERENTE** (legajo, apellido, nombre, *id-dpto*)

# Relaciones “1:1”



Esta relación podría eliminarse de 2 formas:

- Agregando el atributo ID-Departamento en el conjunto de entidades Gerente.
- Agregando el atributo Legajo en el conjunto de entidades Departamento.

**Gerente + Dirige:**

a) DEPARTAMENTO (Id-dpto, nombre, *legajo*)

ó

b) GERENTE (legajo, apellido, nombre, *id-dpto*)

# Relaciones IS-A

- Existen distintas alternativas para reducir al esquema tabular una **relación “es-un”**:

## Opción 1

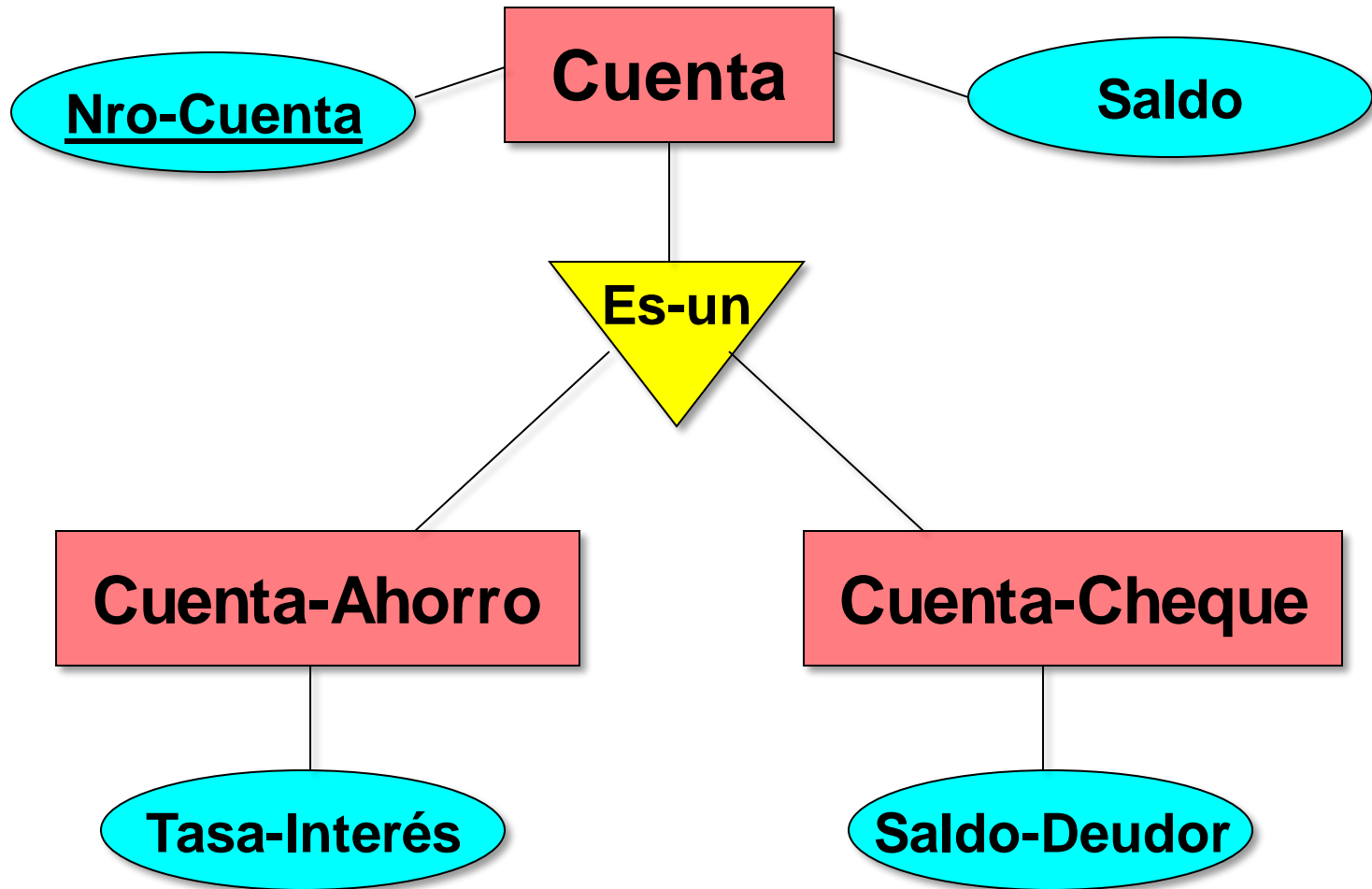
- Se crea una única tabla *para el conjunto de entidades de nivel más alto*.
- Para cada *conjunto de entidades de nivel más bajo se crea* una tabla con los atributos de ese conjunto de entidades más una columna por cada atributo que es clave primaria del conjunto de entidades de nivel más alto.
- Es válida para generalizaciones solapadas o disjuntas y parciales o totales.

# Relaciones IS-A

## Opción 2

- No se crea una tabla para el conjunto de entidades de nivel más alto.
- Para cada conjunto de entidades de nivel más bajo se crea una tabla que con los atributos de ese conjunto de entidades más una columna por cada atributo del conjunto de entidades de nivel más alto.
- Esta conversión es válida para generalizaciones disjuntas y totales.

# Especialización y Generalización



# Tablas para Diagramas E-R Extendidos

## Opción 1

### Cuenta

Nro-Cuenta	Saldo

### Cuenta-Ahorro

Nro-Cuenta	Tasa-Interés

### Cuenta-Cheque

Nro-Cuenta	Saldo-Deudor

## Opción 2

### Cuenta-Ahorro

Nro-Cuenta	Saldo	Tasa-Interés

### Cuenta-Cheque

Nro-Cuenta	Saldo	Saldo-Deudor

## Ejercicio

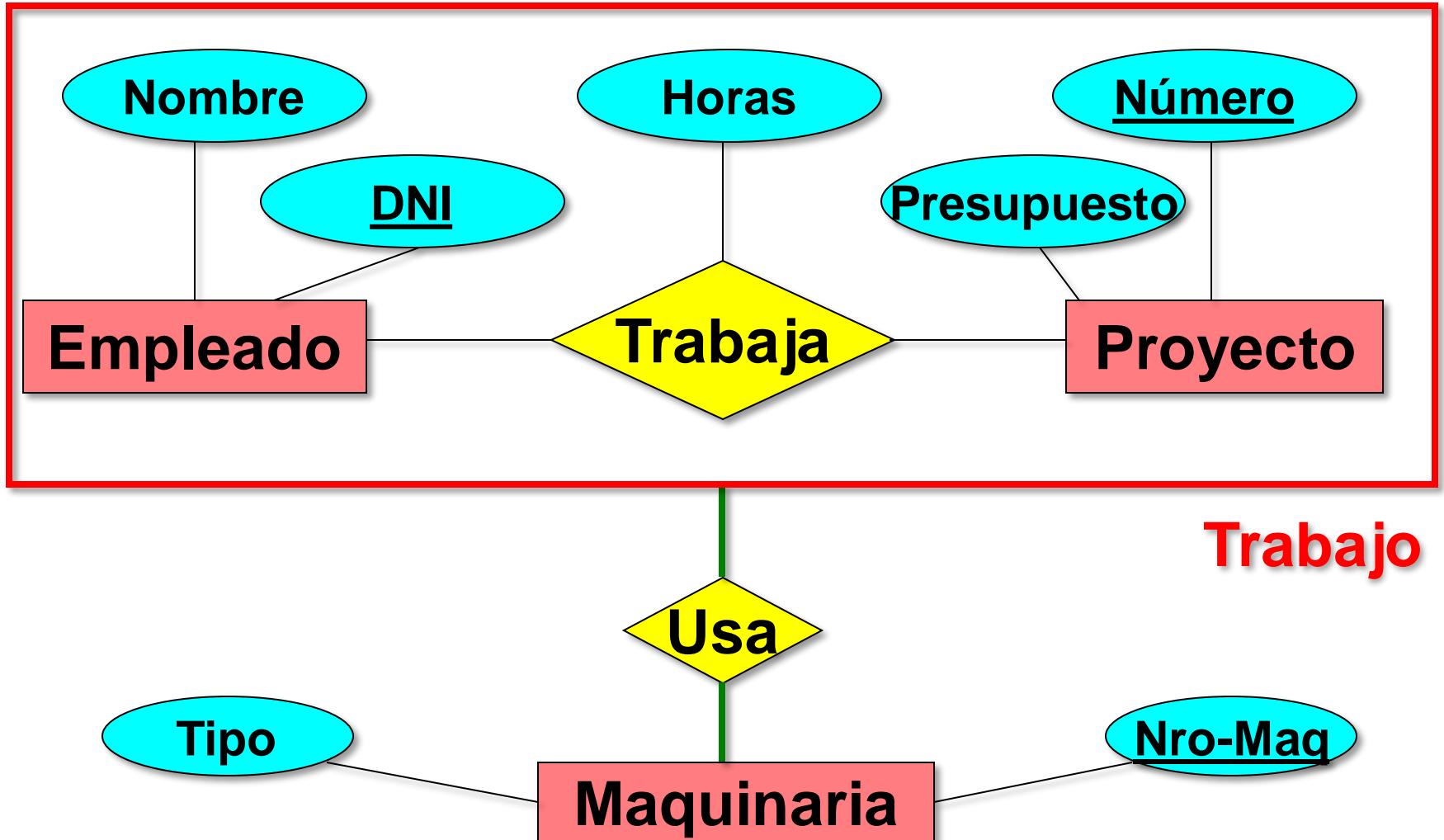
- De acuerdo a la clasificación de las relaciones “IS-A” en
  - Por Nivel inferior
    - Solapadas
    - Disjuntas
  - Distribución
    - Totales
    - Parciales
- Identificar la/s representaciones como tabla más adecuadas. Justificar

# Agregación

- La transformación de una **relación de agregación** es directa, según las reglas vistas.
- Consideremos el ejemplo, se necesitan tablas para las entidades fuertes:
  - **Empleado, Proyecto y Maquinaria.**
- Y para las relaciones:
  - **Trabaja.**
  - **Usa.**
    - Incluye una columna para cada atributo en la clave primaria del conjunto de entidades **maquinaria** y para cada atributo de la clave primaria de **trabaja.**



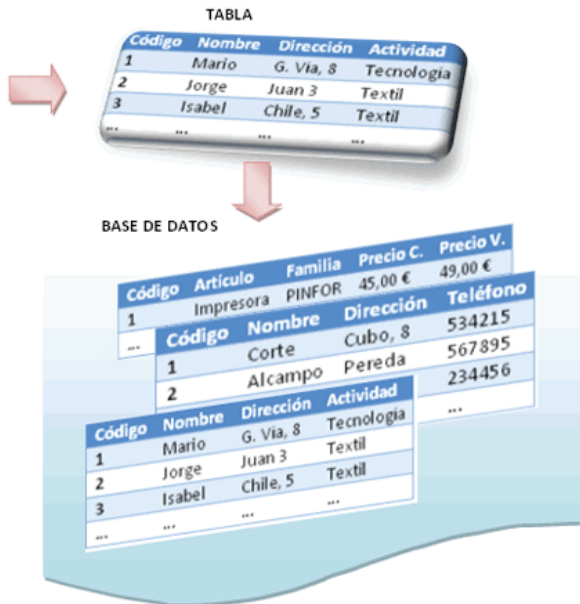
# Agregación



# Ejemplo

- **EMPLEADO** (DNI, nombre).
- **PROYECTO** (número, presupuesto).
- **MAQUINARIA** (tipo-máquina, nro-máquina)
  
- **TRABAJA** (DNI, número, horas)
- **USA** (tipo-máquina, DNI, número)

# Modelo de Datos Relacional



Conceptos  
teóricos

# Modelo Relacional

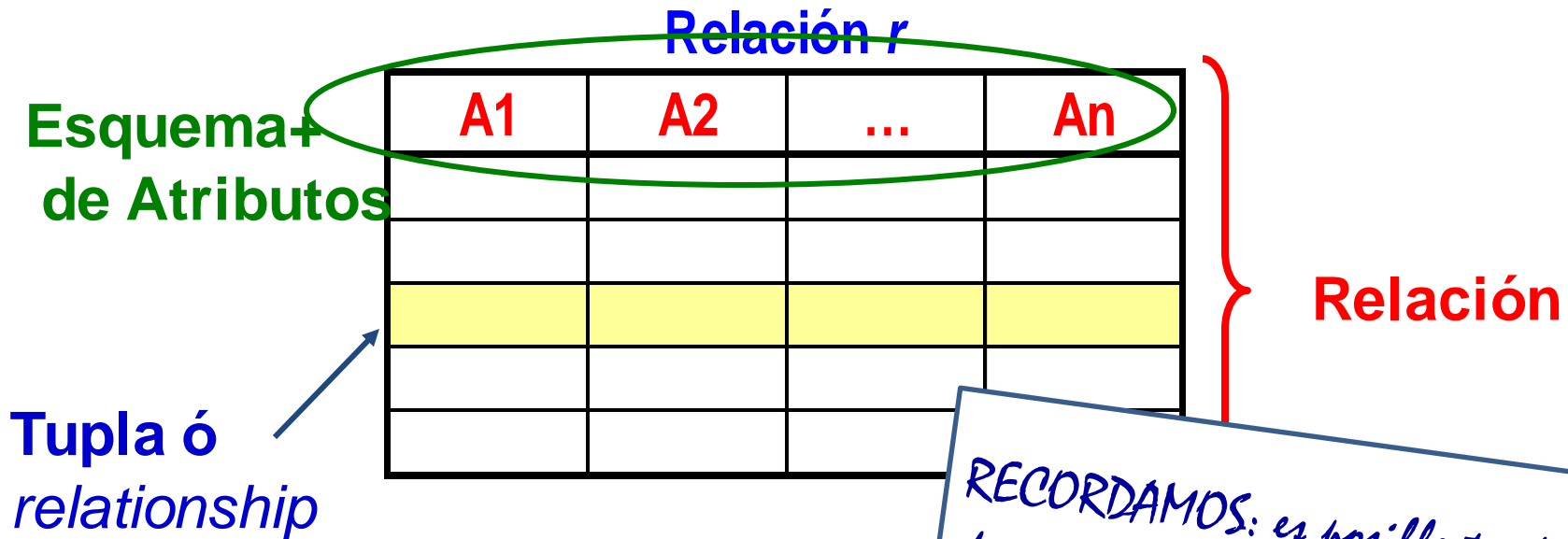
- El **modelo relacional** se basa en definir una colección de tablas para representar las entidades y las relaciones entre los datos.
  - Está basado en el uso de **relaciones** (*relations*).
  - Las **relaciones** representan a **los conjuntos entidad** y los **conjuntos relación** del modelo E-R.
  - Cada relación puede pensarse como una **tabla** compuesta por *filas o tuplas*.
  - Los valores de atributo de una **tupla** tienen asociados un *dominio de atributo*.

# Estructura básica

- Las relaciones son conjuntos de tuplas. *No interesa el orden* en el que las tuplas aparecen dentro de la misma.
- Se requiere que *los dominios* de los atributos sean atómicos.
- Varios atributos pueden tener el mismo dominio asociado.
- Un valor que es miembro de múltiples dominios es el *null*.

# Modelo Relacional

**Modelo Relacional:** define una colección de tablas para representar datos y relaciones entre ellos.



# Modelo Relacional - Definiciones

**Esquema de Base de Datos** – es el diseño lógico de la base de datos.

- El esquema de base de datos se compone de varios *esquemas de relación*

**Instancia de Base de Datos** – el contenido de la base de datos en un instante de tiempo. El contenido queda distribuido en distintas ‘tablas’ o relaciones.

# Modelo Relacional – Definiciones

**Esquema de Relación** – es el diseño lógico de una relación.

– Esquema-Cuenta (nro-cuenta, saldo)

**Relación** – es el contenido de un esquema de relación.

– **cuenta** (Esquema-Cuentas)



# Ejemplo



- **Esquema-libro** (ISBN, título, editorial)
- **Esquema-autor** (id, apellido, nombre)
- **Esquema-escrito-por** (ISBN, ID)
  
- **libro** (Esquema-libro).
- **autor** (Esquema-autor)
- **escrito-por** (Esquema-escrito-por)

# Ejemplo

ISBN	Título	Editorial
111-2222	Databases: From Relational to Object-Oriented Systems	McGrawHill
111-3333	Database System Concepts	McGrawHill
111-4444	Operating System Concepts	Amazon

## *libros*

## *autores*

id	Apellido	nombre
A-11	Delobel	C
A-12	Decluse,	Ch
A-13	Phillippe	R.
A-16	Silberschatz	A.
A-15	Korth	H
A-22	Sudarshan	S

ISBN	id
111-2222	A-11
111-2222	A-12
111-2222	A-13
111-3333	A-16
111-3333	A-15
111-3333	A-22
111-4444	A-16
111-4444	A-19

## *escrito-por*

# Modelo E-R a Modelo Relacional

- Para las siguientes definiciones sobre relaciones binarias definidas en el modelo ER asumimos los conjuntos de entidades:

$A (A_1, \dots, A_m)$  y  $B (B_1, \dots, B_n)$

- Con *llaves primarias* (pk):

$(A_1, \dots, A_i)$  para A y  $(B_1, \dots, B_j)$  para B

- Notación:

–  $(\underline{A_1}, \dots, \underline{A_i}, \dots, A_m)$  para A

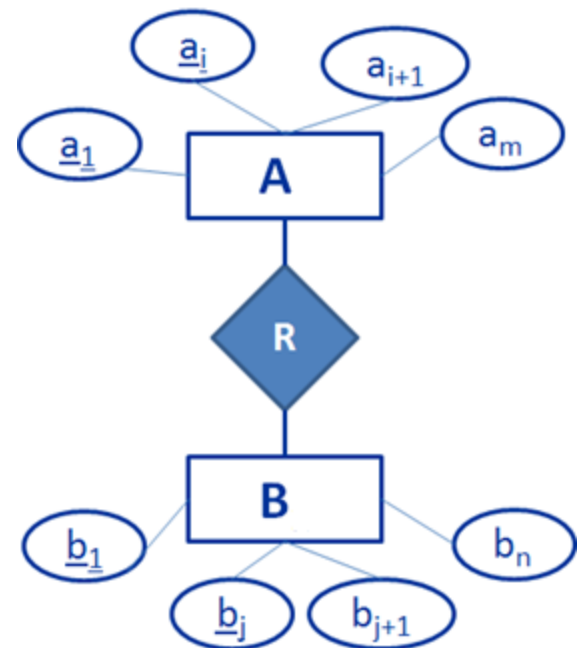
–  $(\underline{B_1}, \dots, \underline{B_j}, \dots, B_n)$  para B

# Modelo E-R a Modelo Relacional

- Sea R una relación “*binaria*” del modelo E-R que la vincula A y B con cardinalidad *muchos a muchos sin atributos*:

– Solución (general):

- $A(\underline{A_1}, \dots, \underline{A_i}, \dots, A_m)$ .
- $B(\underline{B_1}, \dots, \underline{B_j}, \dots, B_n)$ .
- $R(\underline{A_1}, \dots, \underline{A_i}, \underline{B_1}, \dots, \underline{B_j})$ .



# Modelo E-R a Modelo Relacional

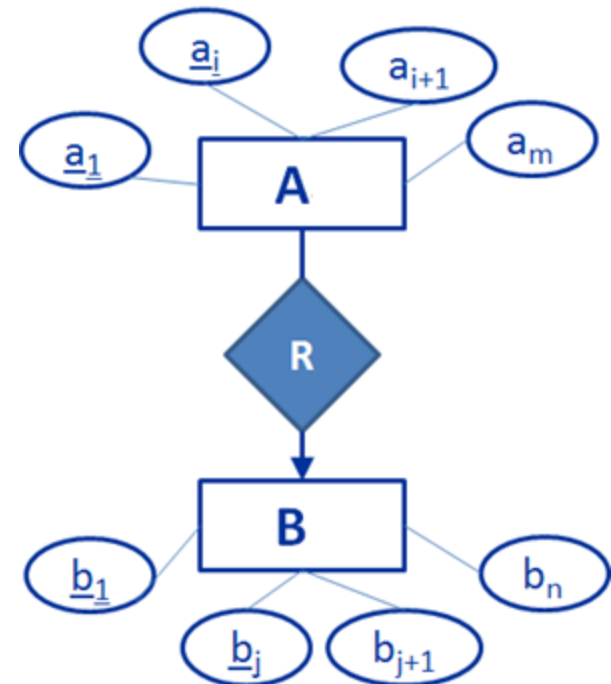
- Sea R una relación “binaria” del modelo E-R que la vincula A y B con cardinalidad *muchos a uno*:

– Solución Costosa (general):

- A (A<sub>1</sub>, ..., A<sub>j</sub>, ..., A<sub>m</sub>).
- B (B<sub>1</sub>, ..., B<sub>j</sub>, ..., B<sub>n</sub>).
- R (A<sub>1</sub>, ..., A<sub>j</sub>, B<sub>1</sub>, ..., B<sub>j</sub>).

– Solución Económica:

- A (A<sub>1</sub>, ..., A<sub>j</sub>, ..., A<sub>m</sub>, B<sub>1</sub>, ..., B<sub>j</sub>).
- B (B<sub>1</sub>, ..., B<sub>j</sub>, ..., B<sub>n</sub>).



# Modelo E-R a Modelo Relacional

- Sea R una relación “binaria” del modelo E-R que la vincula A y B, con cardinalidad *uno a uno*:
  - Como caso particular de relación muchos a uno.
    - A (A<sub>1</sub>, ..., A<sub>i</sub>, ..., A<sub>m</sub>, B<sub>1</sub>, ..., B<sub>j</sub>).
    - B (B<sub>1</sub>, ..., B<sub>j</sub>, ..., B<sub>n</sub>).

ó

- A (A<sub>1</sub>, ..., A<sub>i</sub>, ..., A<sub>m</sub>).
- B (B<sub>1</sub>, ..., B<sub>j</sub>, ..., B<sub>n</sub>, A<sub>1</sub>, ..., A<sub>i</sub>).

# Generalizando

- Sean los conjuntos de entidades  $E_1, E_2, \dots, E_n$ , con llaves  $k_1, \dots, k_n$  respectivamente.
- Sea  $R$  una **relación n-aria** del modelo E/R que vincula  $E_1, E_2, \dots$  y  $E_n$ :
  - Solución General:
    - $E_1 = (\underline{A_1}, \dots, \underline{A_{i_1}}, \dots, A_{m_1})$
    - $E_2 = (\underline{B_1}, \dots, \underline{B_{i_2}}, \dots, B_{m_2})$
    - ...
    - $E_n = (\underline{N_1}, \dots, \underline{N_{j_n}}, \dots, N_{m_n})$
    - $R = (A_1, \dots, A_{i_1}, B_1, \dots, B_{i_2}, \dots, N_1, \dots, N_{j_n})$ .

La PK de  $R$  depende de la semántica del problema

# Modelo E-R a Modelo Relacional

- Sea R una relación *n-aria* del modelo E/R, *con atributos propios*, se agregan los atributos a la relación:

$$- E_1 = (\underline{A_1}, \dots, \underline{A_{i_1}}, \dots, A_{m_1})$$

$$- E_2 = (\underline{B_1}, \dots, \underline{B_{i_2}}, \dots, B_{m_2})$$

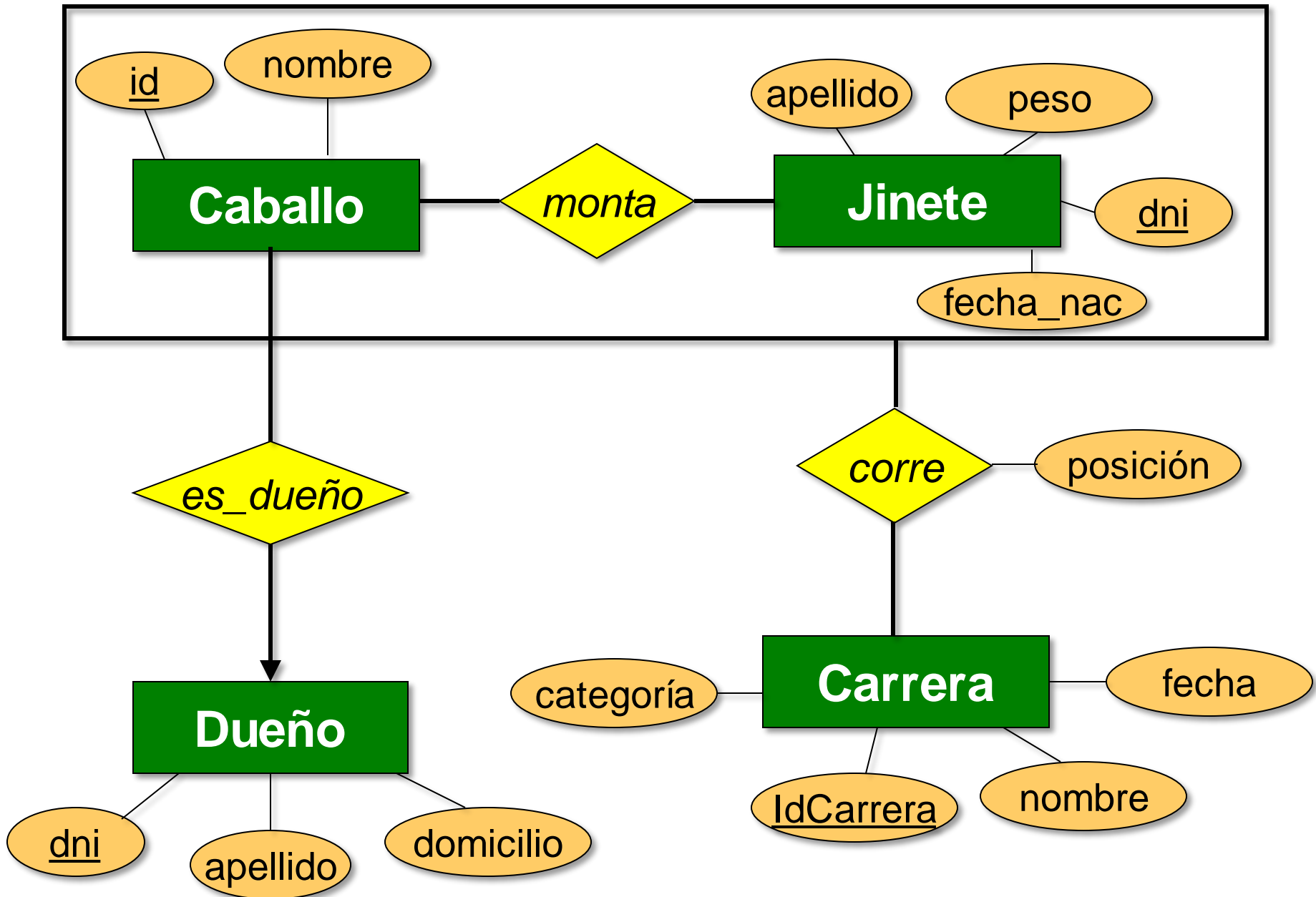
...

$$- E_n = (\underline{N_1}, \dots, \underline{N_{j_n}}, \dots, N_{m_n})$$

$$- R = (A_1, \dots, A_{i_1}, B_1, \dots, B_{i_2}, \dots, N_1, \dots, N_{j_n}, a_1, \dots, a_n).$$

La PK de R depende del problema.  
Podría o no involucrar a uno o más de los atributos  $a_i$ .





# Temas de la clase de hoy

- Modelo Entidad – Relación
  - Consideraciones de diseño.
- El ME-R y UML – comparación.
- Pasaje a tablas.
- Modelo Relacional

## Bibliografía:

- *“Conceptos de Sistemas de Bases de Datos”* – Abraham Silberschatz 5ta. Ed – Capítulos 7, 8 (Ed.2010)
- *“Principles of Database and Knowledge-Base Systems”* – J. Ullman. Capítulo 2.