



Unitat Didàctica 4: Disseny de Bases de Dades Relacionals

Part 3: Disseny lògic

(Doc. UD4.3)

UD 4-3 Disseny lògic

1 Introducció

2 Transformació de les classes

- Classes fortes
- Classes dèbils
- Classes especialitzades

3 Transformació de les associacions

- Binaries
- Reflexives
- Atributs en les associacions (d'enllaç)
- Associacions amb atributs
- Transformació de la associació quan s'associa amb altres classes
- Elecció de les directrius de les claus alienes

4 Exemple

5 Teoria de la Normalització

1 Disseny lògic

1ª FASE: Análisis

Investigación

Req. de
información

Req. de
procesos

2ª FASE: Diseño

Modelo semántico

Diseño conceptual

Esquema conceptual

Estática

Dinámica

Tecnología de gestión
de datos

Diseño lógico

Esquema
lógico

Esquemas de
transacciones

SGBD

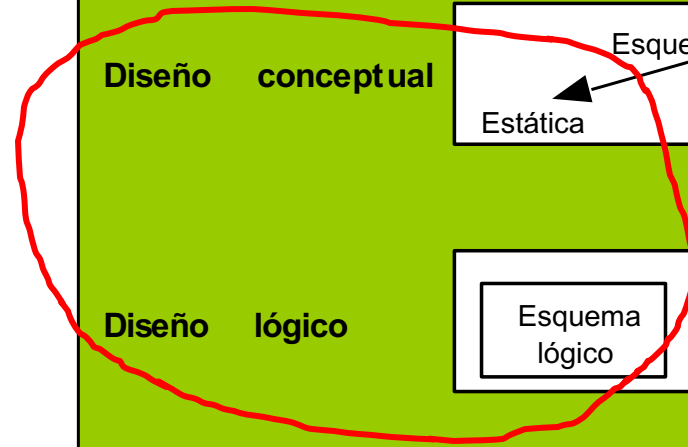
Diseño físico

Esquema
físico

Diseño y
desarrollo de
Programas

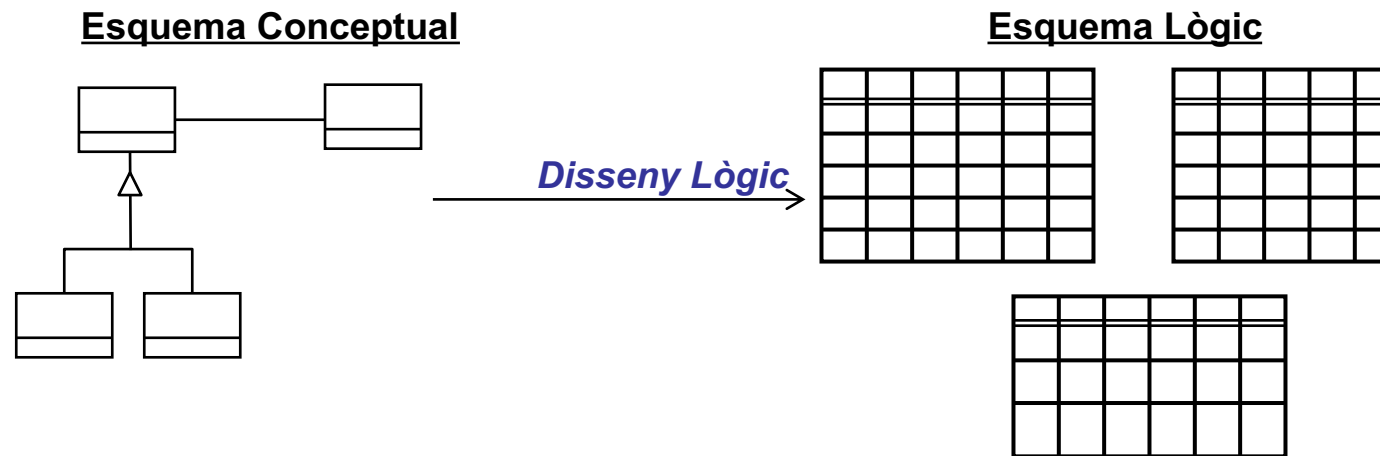
3ª FASE: Implantación

Carga de la
base de datos



1 Disseny lògic

Disseny lògic: transformació de l'esquema conceptual, que està descrit amb un cert model de dades (UML en el nostre cas), en estructures descrites en termes del model de dades en el qual es base el sistema de gestió de bases de dades que s'utilitzarà (model relacional en el nostre cas).



1 Disseny lògic

- Transformacions: es basen en la definició de CP, CAj, VNN, Únic
- Aquelles propietats expressades en el diagrama que no es puguin representar en l'esquema relacional s'hauran d'incloure en una *llista de restriccions d'integritat* que s'hauran de controlar amb restriccions generals (assertions, triggers o programa).
- Quan tinguem més d'un esquema relacionals possible:

“Elegirem l'esquema amb menys restriccions d'integritat afegides. Davant la igualtat de restriccions, elegirem l'esquema amb menys relacions”.

2 Transformació de classes (fortes)

| A |
|---|
| $a_0: \{id\}: t_{a_0}$ |
| $a_1: \{\text{únic}_1\}: \{0..1\}: t_{a_1}$ |
| $a_2: \{1..1\}: t_{a_2}$ |
| $a_3: \{0..1\}: t_{a_3}$ |
| $a_4: \{1..*\}: t_{a_4}$ |
| $a_5: \{0..*\}: t_{a_5}$ |
| $a_6: \{0..1\}: t_{a_6}$ |
| $a_{61}: t_{a_{61}}$ |
| $a_{62}: t_{a_{62}}$ |

$A(a_0: t_{a_0}, a_1: t_{a_1}, a_2: t_{a_2}, a_3: t_{a_3}, a_{61}: t_{a_{61}}, a_{62}: t_{a_{62}})$

CP: $\{a_0\}$

Único: $\{a_1\}$

VNN: $\{a_2\}$

$A4(a_0: t_{a_0}, a_4: t_{a_4})$

CP: $\{a_0, a_4\}$

CAj: $\{a_0\} \rightarrow A(a_0)$

$A5(a_0: t_{a_0}, a_5: t_{a_5})$

CP: $\{a_0, a_5\}$

CAj: $\{a_0\} \rightarrow A(a_0)$

RI1: Tot valor que aparega en l'atribut a_0 d'A ha d'aparèixer en l'atribut a_0 de A4.

| Persona |
|---|
| DNI: {id}: char NSS: {unico ₁ }: {1..1}: char nombre: {1..1}: propio:{1..1}: char apellidos:{1..1}: char edad: {0..1}: int teléfonos: {0..*}: char |

Persona (DNI:char, NSS:char, nom_propio:char, apellidos:char, edad:int)

CP:{DNI}

Único:{NSS}

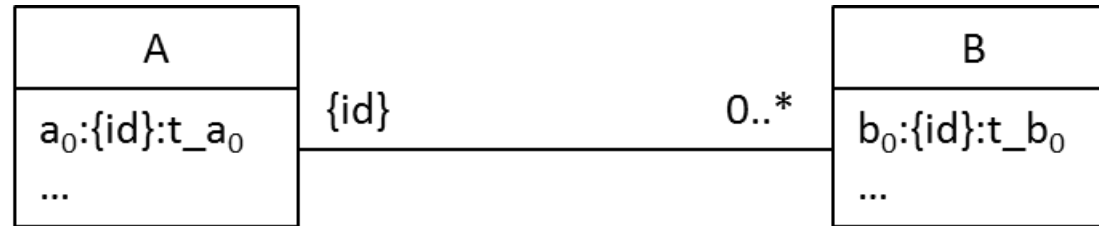
VNN:{NSS,nom_propio,apellidos}

Telefons_Persona (DNI:char, teléfono:char)

CP:{DNI,teléfono}

CAj:{DNI}->Persona(DNI)

2 Transformació de classes (dèbils)



$A(a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$

$B(b_0:t_{b_0}, a_0:t_{a_0}, \dots)$
 $CP: \{a_0, b_0\}$
 $CAj: \{a_0\} \rightarrow A(a_0)$

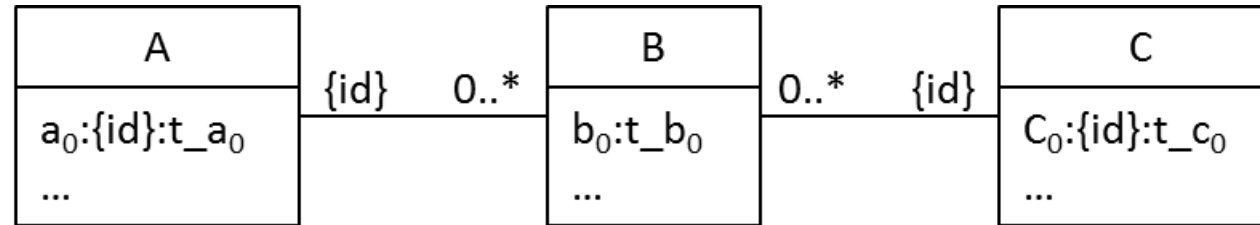
1.1 Transformació de classes (dèbils)



$A(a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$

$B(b_0:t_{b_0}, a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$
 $CAj: \{a_0\} \rightarrow A(a_0)$

1.1 Transformació de classes (dèbils)

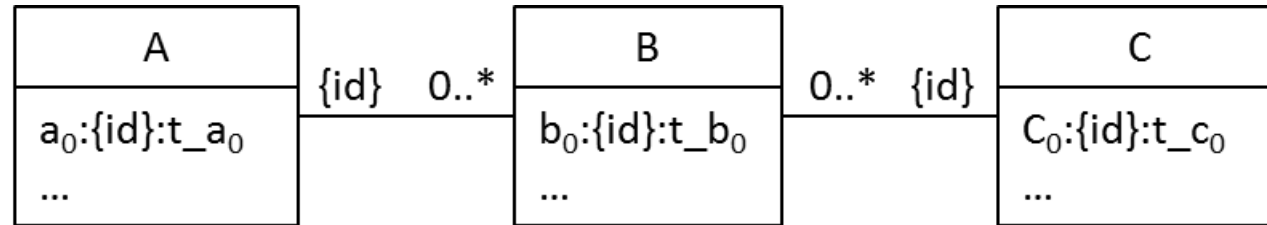


$A(a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$

$C(c_0:t_{c_0}, \dots)$
 $CP: \{c_0\}$

$B(a_0:t_{a_0}, c_0:t_{c_0}, b_0:t_{b_0}, \dots)$
 $CP: \{a_0, c_0\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$
 $CA_j: \{c_0\} \rightarrow C(c_0)$

1.1 Transformació de classes (dèbils)

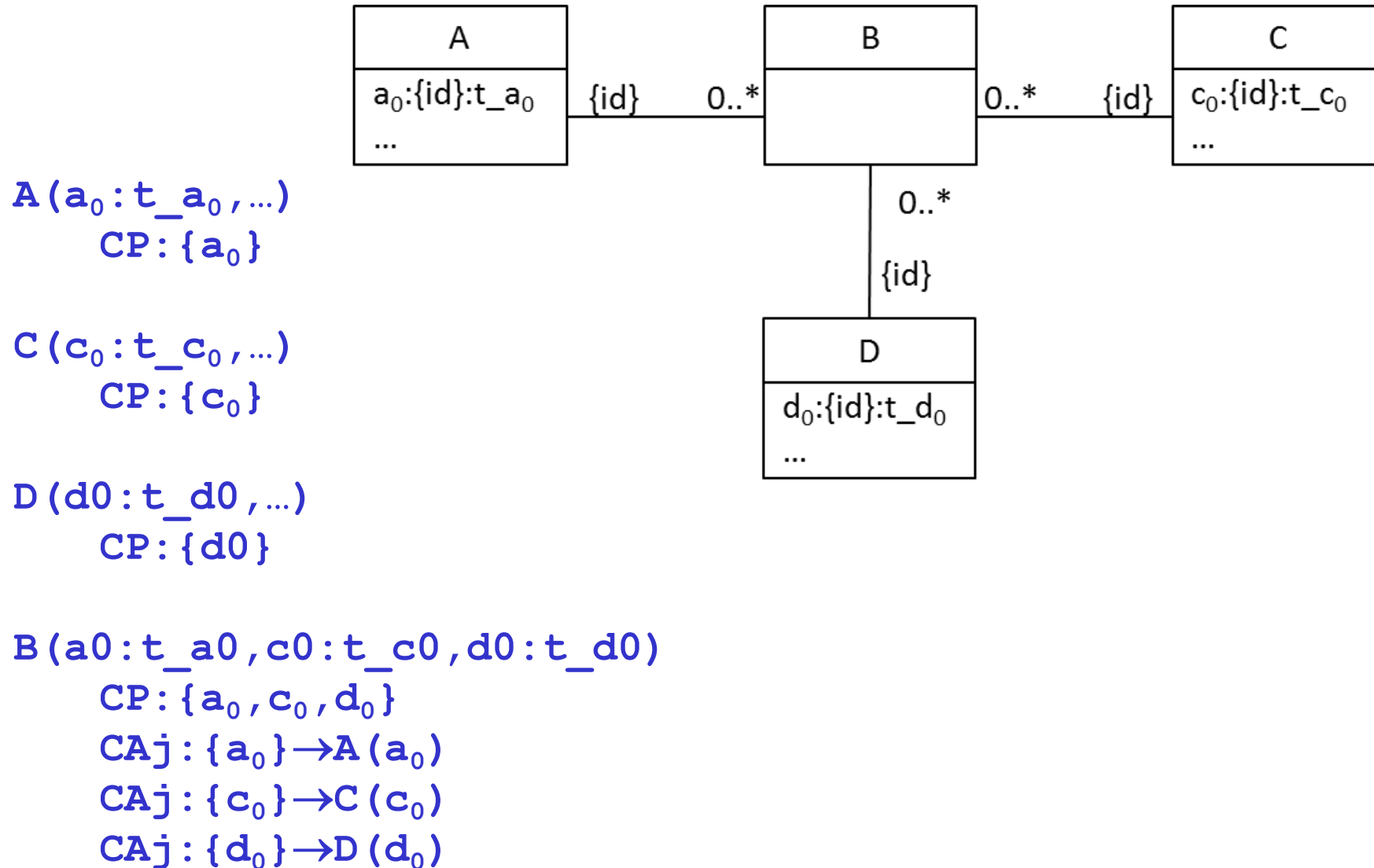


$A(a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$

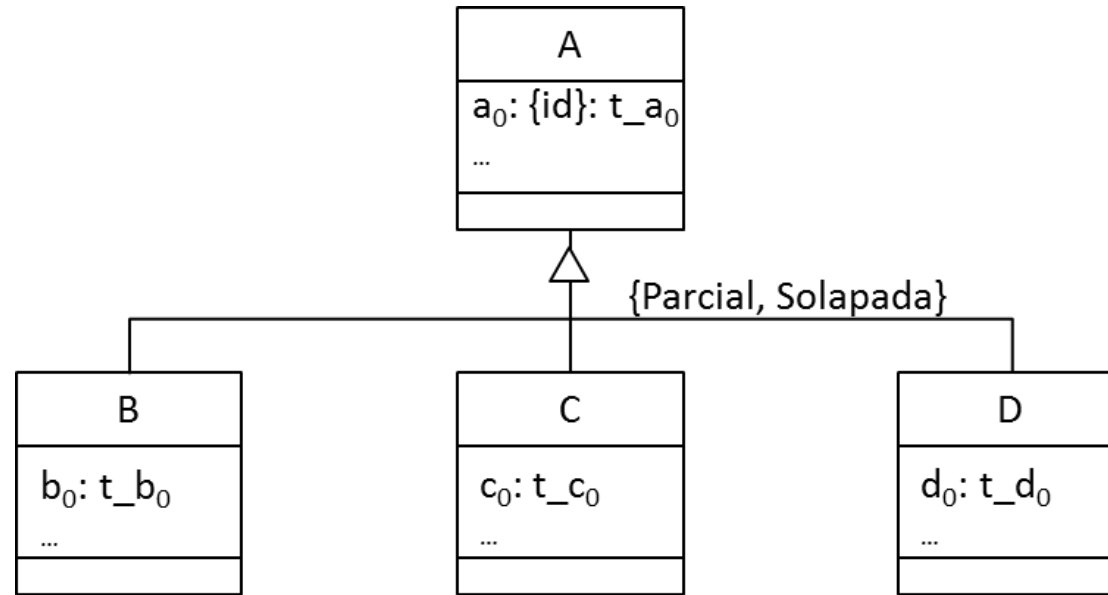
$C(c_0:t_{c_0}, \dots)$
 $CP: \{c_0\}$

$B(a_0:t_{a_0}, c_0:t_{c_0}, b_0:t_{b_0}, \dots)$
 $CP: \{a_0, c_0, b_0\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$
 $CA_j: \{c_0\} \rightarrow C(c_0)$

2 Transformació de classes (dèbils)



2 Transformació de classes (Gen/Esp)



$A(a_0:t_{a_0}, \dots)$
 $CP: \{a_0\}$

$B(a_0:t_{a_0}, b_0:t_{b_0}, \dots)$
 $CP: \{a_0\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$

$C(a_0:t_{a_0}, c_0:t_{c_0}, \dots)$
 $CP: \{a_0\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$
 $D(a_0:t_{a_0}, d_0:t_{d_0}, \dots)$
 $CP: \{a_0\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$

2 Transformació de classes (Gen/Esp)

RI_{Total}: Tot valor que aparega en l'atribut a_0 de A ha d'aparèixer en l'atribut a_0 de B , de C o de D .

RI_{Disjunta}: No pot haver-hi un mateix valor en l'atribut a_0 de B i en el a_0 de C ; ni en el a_0 de B i en el a_0 de D ; ni en l'atribut a_0 de C i en el a_0 de D .

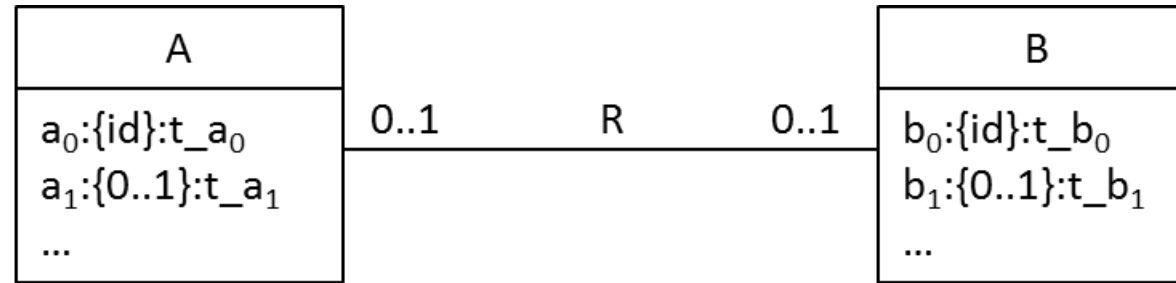
$A(a_0:t_{a_0}, \dots)$
CP: $\{a_0\}$

$B(a_0:t_{a_0}, b_0:t_{b_0}, \dots)$
CP: $\{a_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$

$C(a_0:t_{a_0}, c_0:t_{c_0}, \dots)$
CP: $\{a_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$

$D(a_0:t_{a_0}, d_0:t_{d_0}, \dots)$
CP: $\{a_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$

3 Transf. d'associacions binàries 0..1:0..1

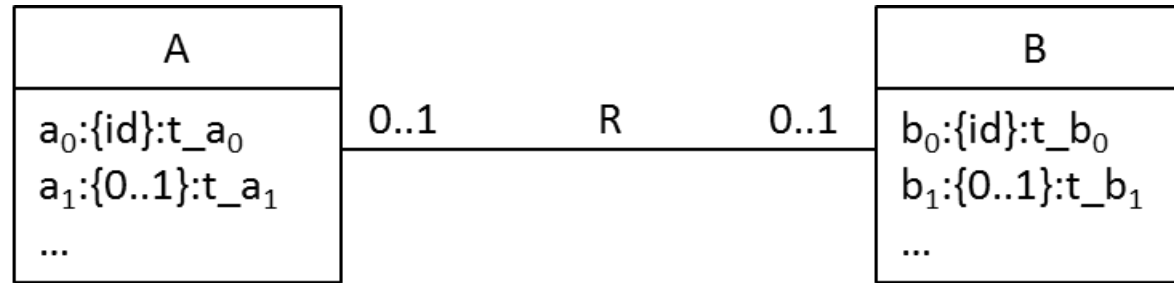


Esquema 1

$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: $\{b_0\}$
Único: $\{a_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$

3 Transf. d'associacions binàries 0..1:0..1



Esquema 2

$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, b_0:t_{b_0})$

CP: $\{a_0\}$

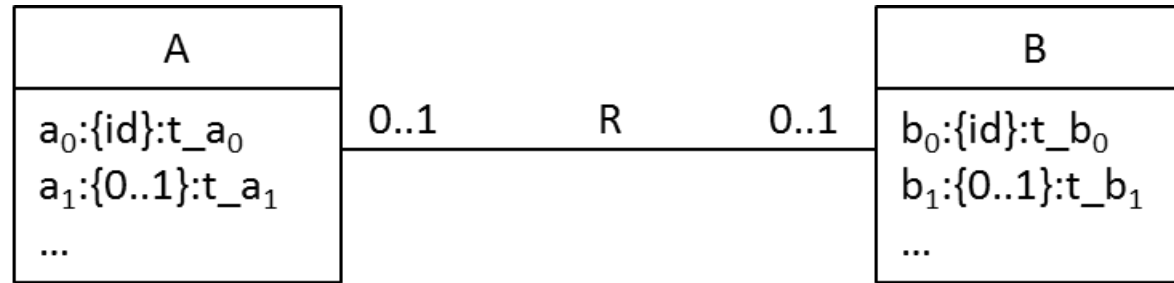
Único: $\{b_0\}$

CAj: $\{b_0\} \rightarrow B(b_0)$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots)$

CP: $\{b_0\}$

3 Transf. d'associacions binàries 0..1:0..1



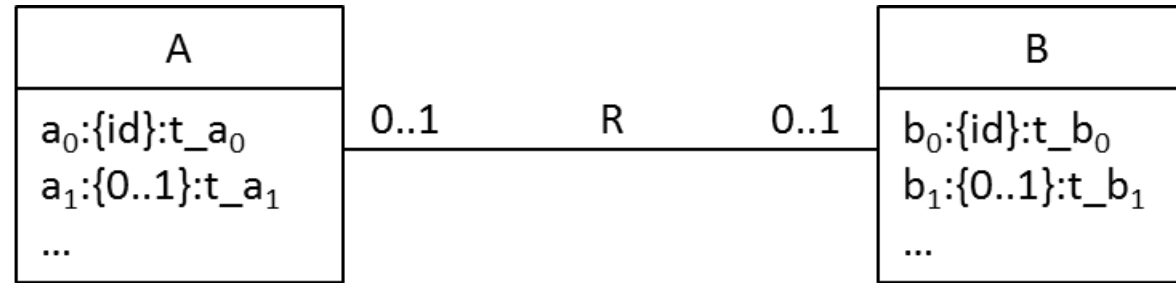
Esquema 3

$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots)$
CP: $\{b_0\}$

$R(b_0:t_{b_0}, a_0:t_{a_0})$
CP: $\{b_0\}$
Único: $\{a_0\}$
VNN: $\{a_0\}$
 $CAj: \{a_0\} \rightarrow A(a_0)$
 $CAj: \{b_0\} \rightarrow B(b_0)$

3 Transf. d'associacions binàries 0..1:0..1



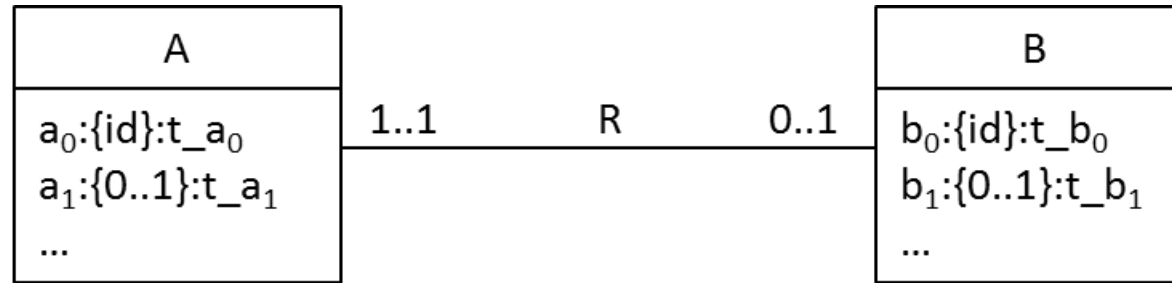
Esquema 4

$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots)$
CP: $\{b_0\}$

$R(b_0:t_{b_0}, a_0:t_{a_0})$
CP: $\{a_0\}$
Único: $\{b_0\}$
VNN: $\{b_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$
CAj: $\{b_0\} \rightarrow B(b_0)$

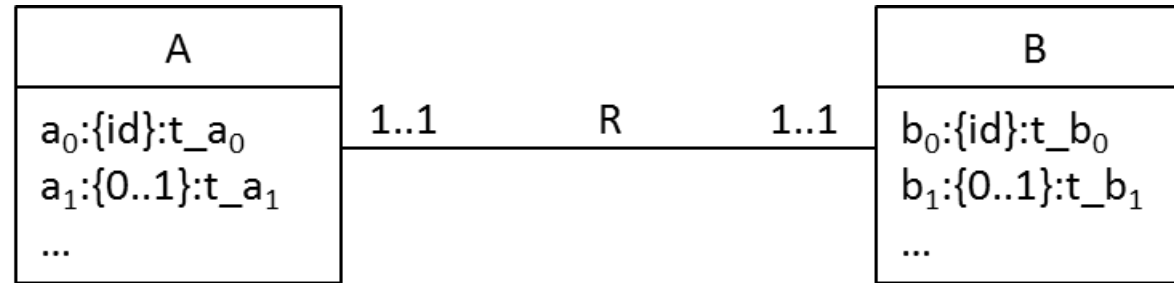
3 Transf. d'associacions binàries 1..1:0..1



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: { a_0 }

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: { b_0 }
Único: { a_0 }
VNN: { a_0 }
CAj: { a_0 } $\rightarrow A(a_0)$

3 Transf. d'associacions binàries 1..1:1..1



Esquema 1

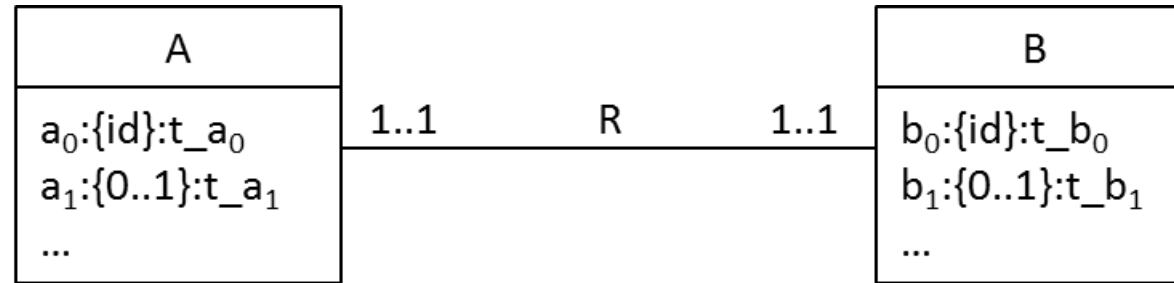
$A-B(a_0:t_{a_0}, a_1:t_{a_1}, \dots, b_0:t_{b_0}, b_1:t_{b_1}, \dots)$

CP: $\{a_0\}$

Único: $\{b_0\}$

VNN: $\{b_0\}$

3 Transf. d'associacions binàries 1..1:1..1



Esquema 2

$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, b_0:t_{b_0})$

CP: $\{a_0\}$

Único: $\{b_0\}$

VNN: $\{b_0\}$

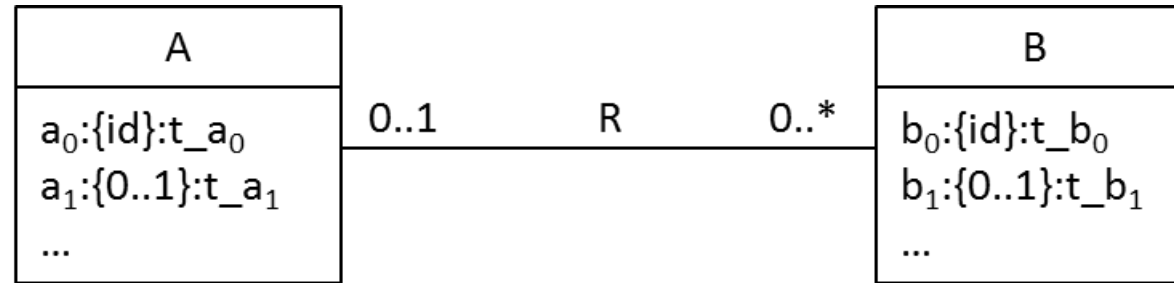
CAj: $\{b_0\} \rightarrow (b_0)$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots)$

CP: $\{b_0\}$

CAj: $\{b_0\} \rightarrow A(b_0)$

3 Transf. d'associacions binàries 0..1:0..*

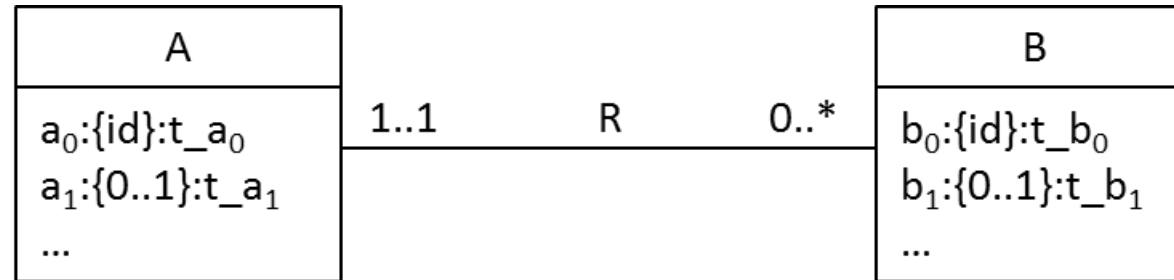


$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: { a_0 }

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: { b_0 }

CAj: { a_0 } $\rightarrow A(a_0)$

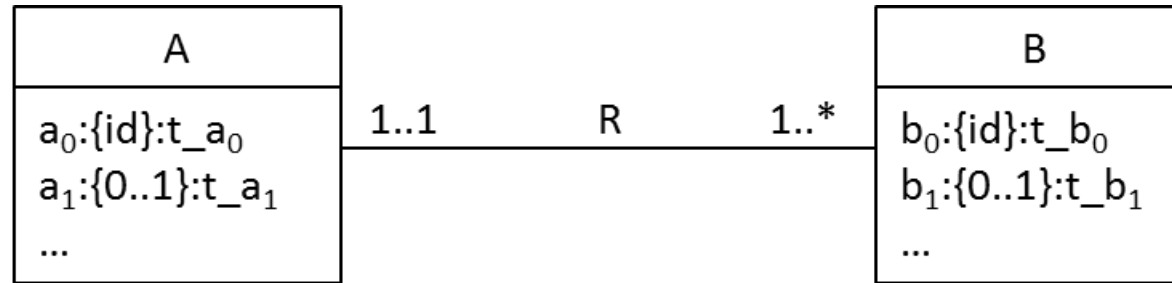
3 Transf. d'associacions binàries 1..1:0..*



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: $\{b_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$
VNN: $\{a_0\}$

3 Transf. d'associacions binàries 1..1:1..*

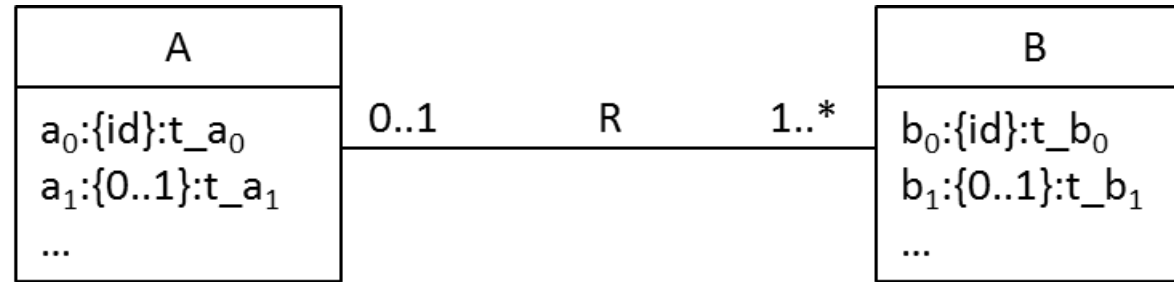


$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: $\{b_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$
VNN: $\{a_0\}$

RI1: Tot valor que aparega en l'atribut a_0 de A ha d'aparèixer en l'atribut a_0 de B.

3 Transf. d'associacions binàries 0..1:1..*

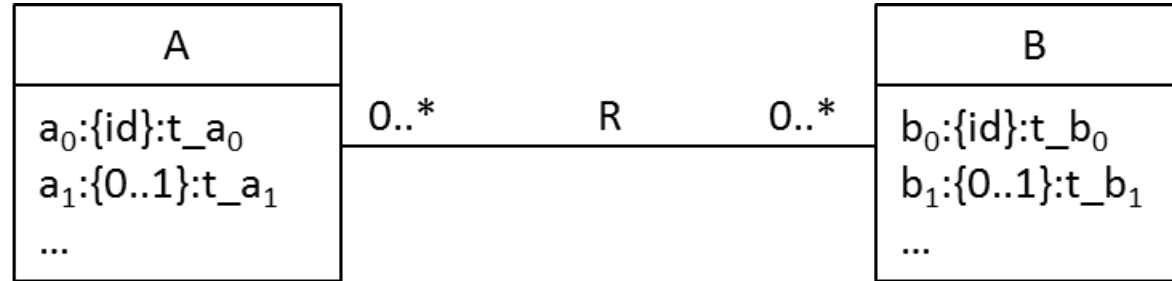


$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$
CP: $\{b_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$

RI1: Tot valor que aparega en l'atribut a_0 de A ha d'apareixer en l'atribut a_0 de B.

3 Transf. d'associacions binàries 0..*:0..*

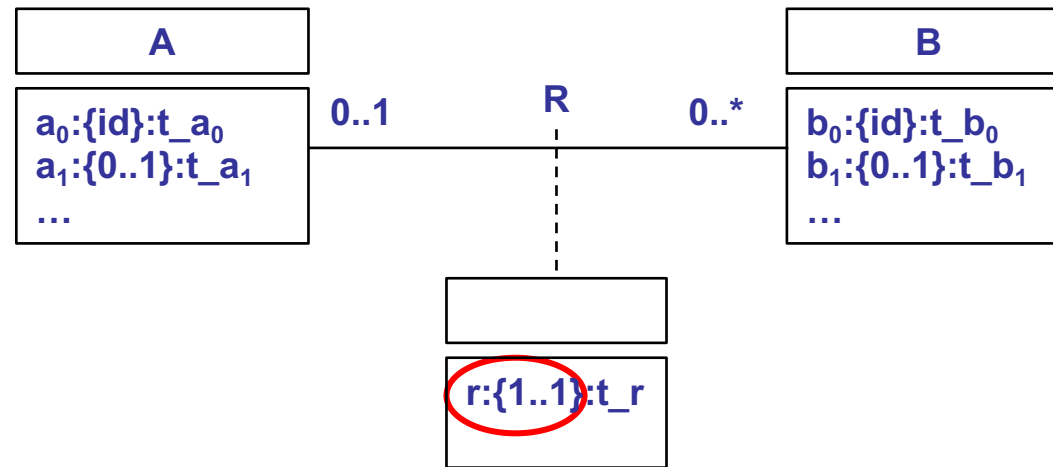


$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$
 $CP: \{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots)$
 $CP: \{b_0\}$

$R(a_0:t_{a_0}, b_0:t_{b_0})$
 $CP: \{a_0, b_0\}$
 $CAj: \{a_0\} \rightarrow A(a_0)$
 $CAj: \{b_0\} \rightarrow B(b_0)$

3 Transf. d'associacions amb atributs



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$

CP: $\{a_0\}$

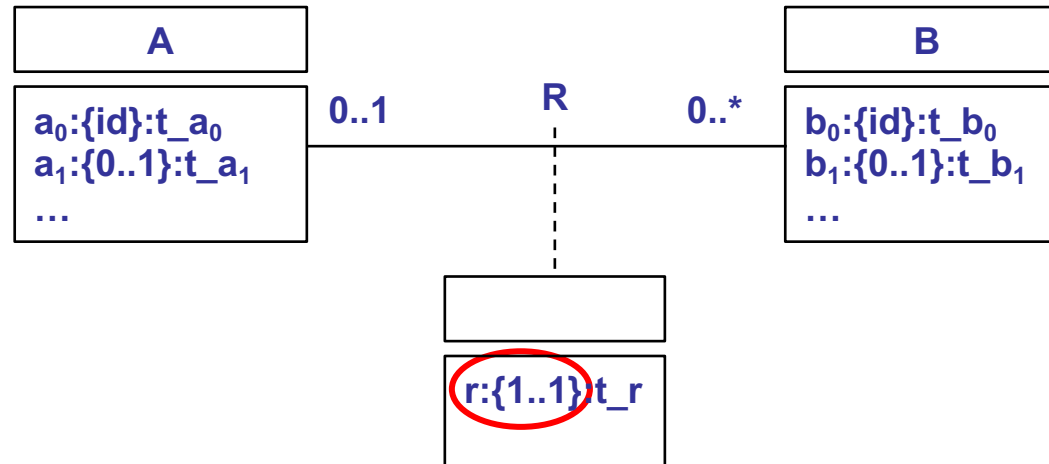
$B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m}, a_0: t_{a_0}, r: t_r)$

CP: $\{b_0\}$

CAj: $\{a_0\} \rightarrow A(a_0)$

RI1: En toda tupla de B s'ha de complir que si l'atribut a_0 és no nul, l'atribut r també ha de ser no nul, i a més, no pot haver-hi cap tupla de B en la qual a_0 siga nul i r distint de nul.

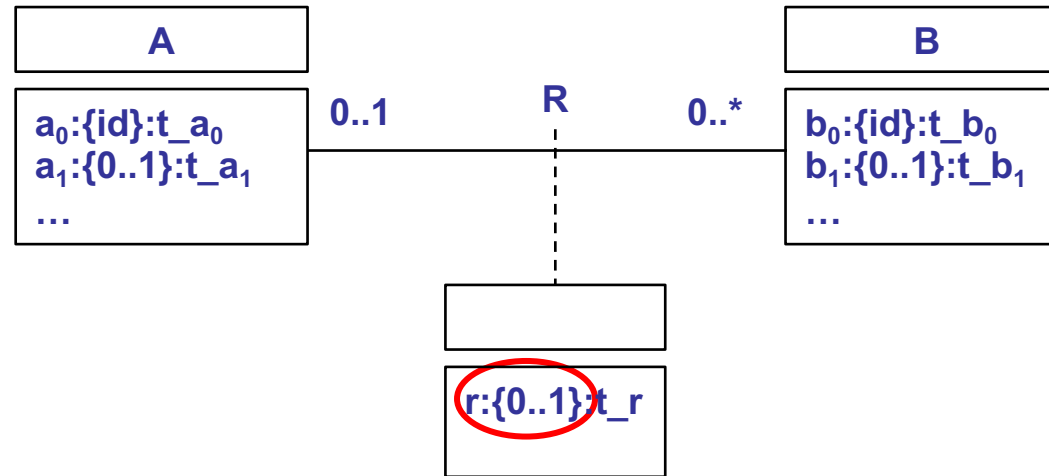
3 Transf. d'associacions amb atributs



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$
 CP: $\{a_0\}$
 $B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m})$
 CP: $\{b_0\}$
 $R(b_0: t_{b_0}, a_0: t_{a_0}, r: t_r)$
 CP: $\{b_0\}$
 VNN: $\{a_0\}$
 VNN: $\{r\}$
 $CA_j: \{a_0\} \rightarrow A(a_0)$
 $CA_j: \{b_0\} \rightarrow B(b_0)$

Solució MILLOR que l'anterior, no inclou RI

3 Transf. d'associacions amb atributs



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$

CP: $\{a_0\}$

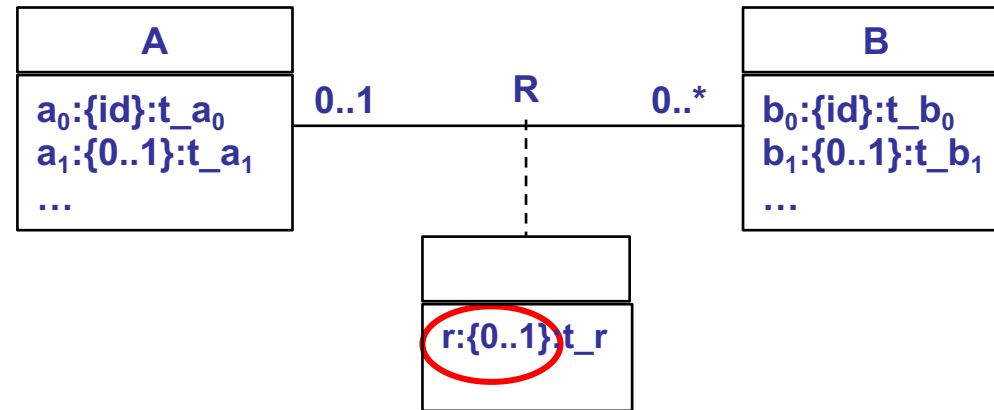
$B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m}, a_0: t_{a_0}, r: t_r)$

CP: $\{b_0\}$

CAj: $\{a_0\} \rightarrow A(a_0)$

RI1: No pot existir una tupla en B que tinga l'atribut a_0 nul i l'atribut r distint de nul

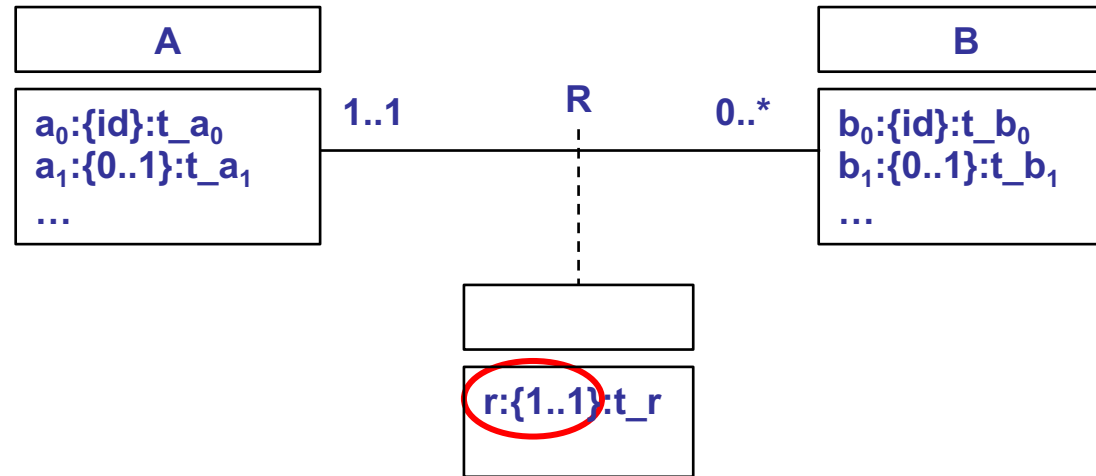
3 Transf. d'associacions amb atributs



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$
CP: $\{a_0\}$
 $B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m})$
CP: $\{b_0\}$
 $R(b_0: t_{b_0}, a_0: t_{a_0}, r: t_r)$
CP: $\{b_0\}$
VNN: $\{a_0\}$
CAj: $\{a_0\} \rightarrow A(a_0)$
CAj: $\{b_0\} \rightarrow B(b_0)$

Solució MILLOR que l'anterior, no inclou RI

3 Transf. d'associacions amb atributs



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$

CP: $\{a_0\}$

$B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m}, a_0: t_{a_0}, r: t_r)$

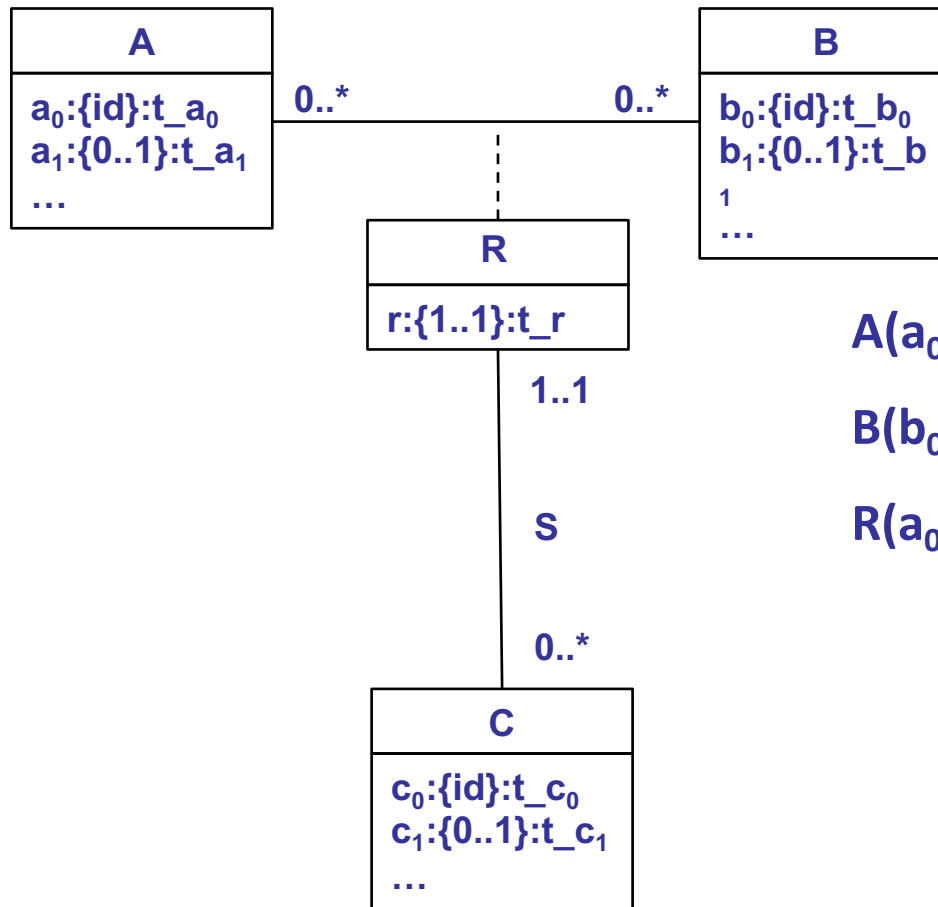
CP: $\{b_0\}$

VNN: $\{a_0\}$

VNN: $\{r\}$

CAj: $\{a_0\} \rightarrow A$

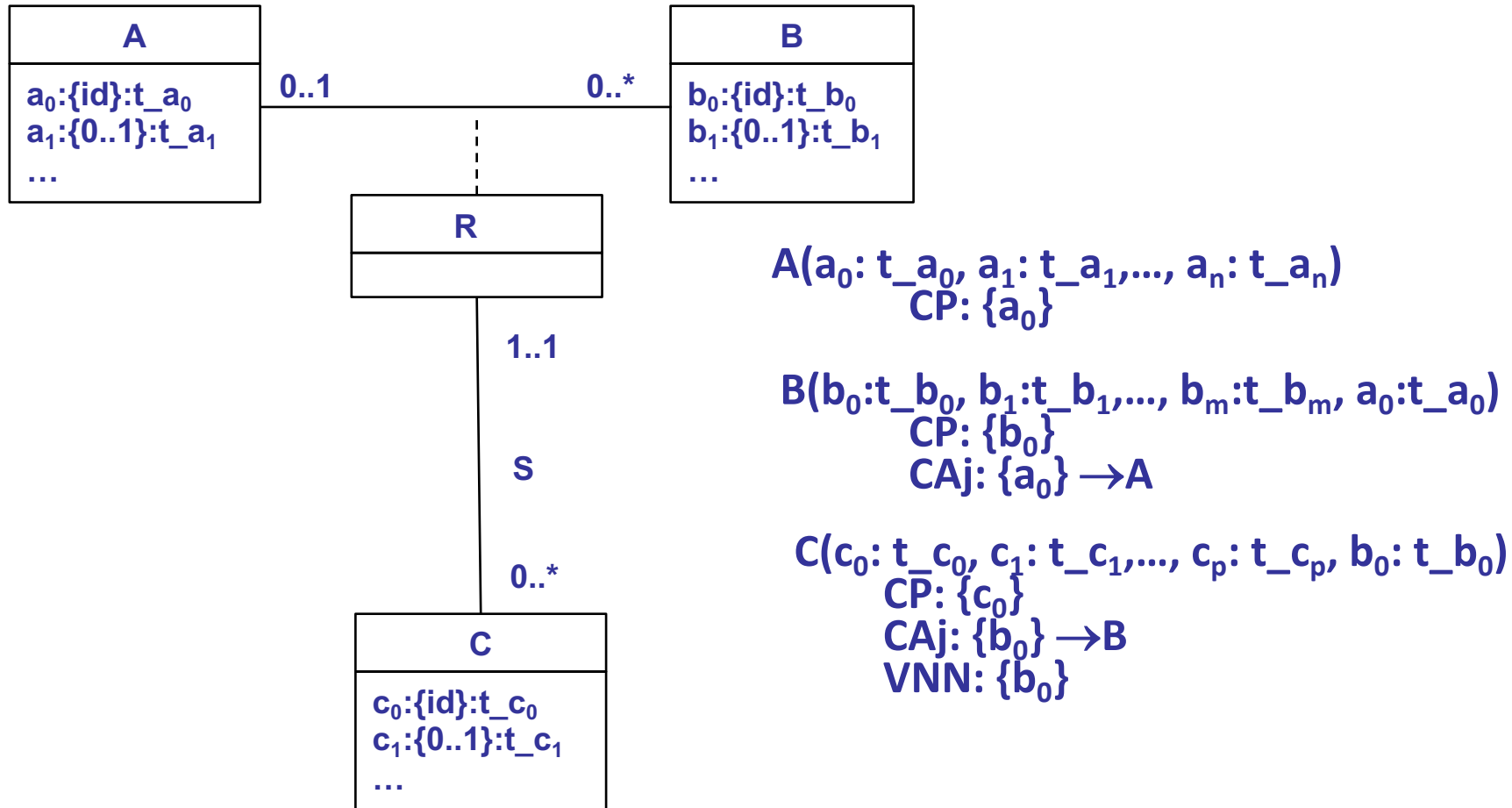
3 Transf. Associació (quan s'associa amb altres classes)



$A(a_0: t_{a_0}, a_1: t_{a_1}, \dots, a_n: t_{a_n})$
 $CP: \{a_0\}$
 $B(b_0: t_{b_0}, b_1: t_{b_1}, \dots, b_m: t_{b_m})$
 $CP: \{b_0\}$
 $R(a_0: t_{a_0}, b_0: t_{b_0}, r: t_r)$
 $CP: \{a_0, b_0\}$
 $CAj: \{a_0\} \rightarrow A(a_0)$
 $CAj: \{b_0\} \rightarrow B(b_0)$

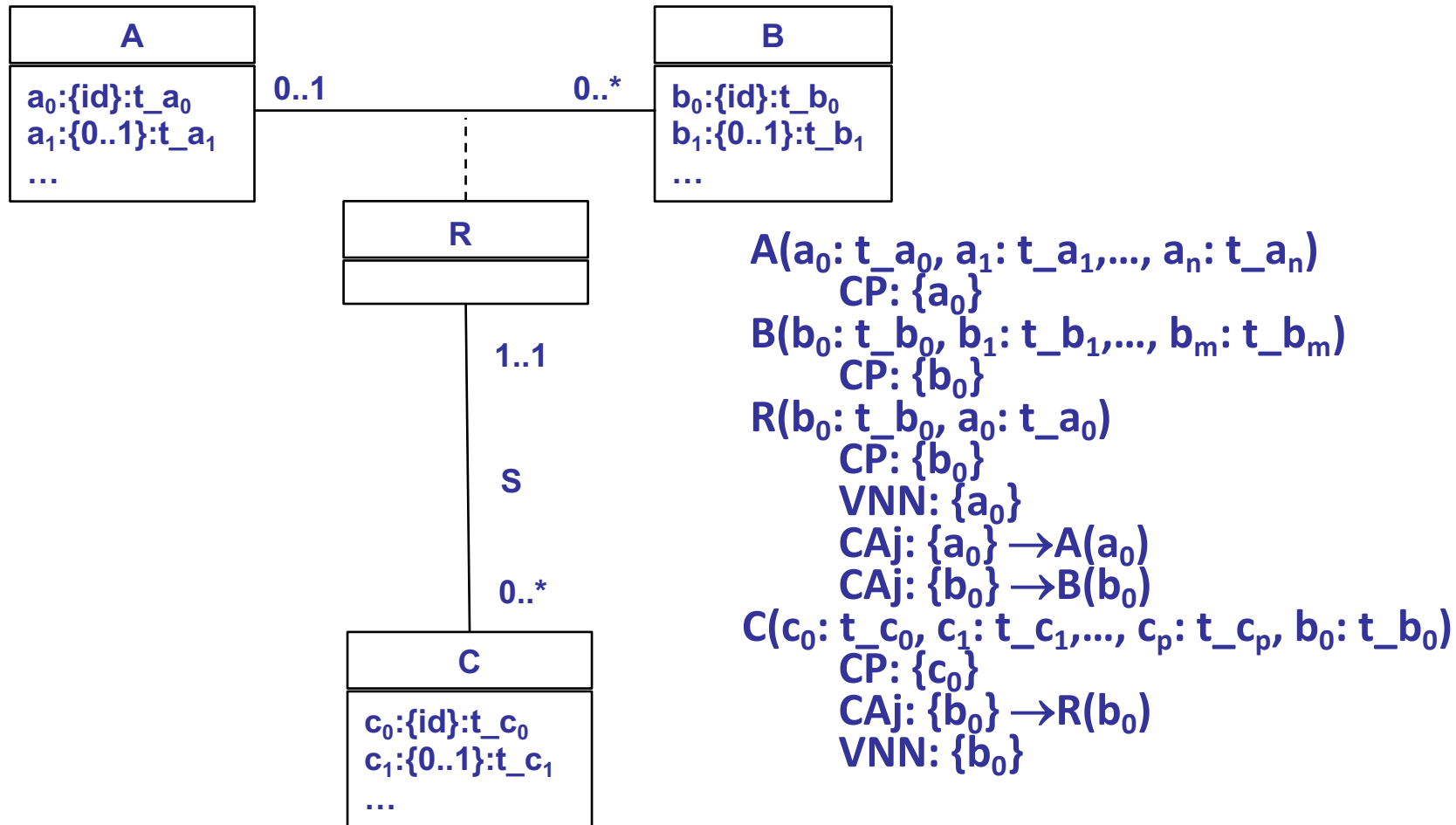
$C(c_0: t_{c_0}, c_1: t_{c_1}, \dots, c_p: t_{c_p}, a_0: t_{a_0}, b_0: t_{b_0})$
 $CP: \{c_0\}$
 $CAj: \{a_0, b_0\} \rightarrow R(a_0, b_0)$
 $VNN: \{a_0, b_0\}$

3 Transf. Associació (quan s'associa amb altres classes)



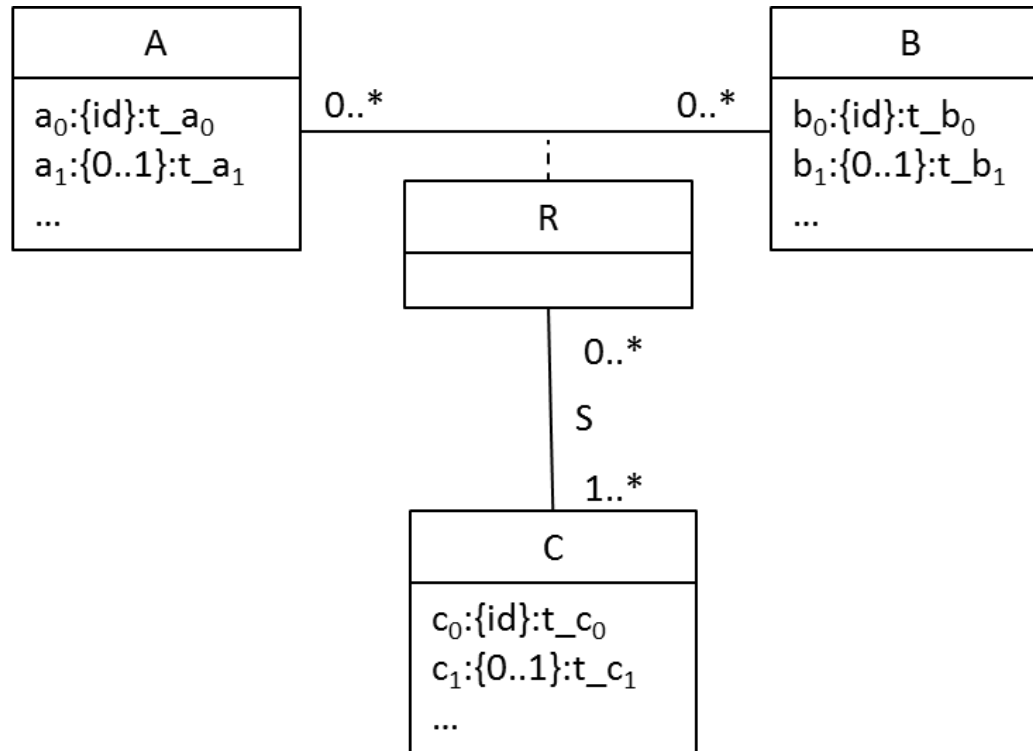
RI1: No pot existir una tupla en C que es relacione amb una tupla de B que no estiga relacionada amb A.

3 Transf. Associació (quan s'associa amb altres classes)



Solució MILLOR que l'anterior, no inclou RI

3 Transf. Associació (quan s'associa amb altres classes)



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$

CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$

CP: $\{b_0\}$

$C(c_0:t_{c_0}, c_1:t_{c_1}, \dots)$

CP: $\{c_0\}$

$R(a_0:t_{a_0}, b_0:t_{b_0})$

CP: $\{a_0, b_0\}$

CAj: $\{a_0\} \rightarrow A(a_0)$

CAj: $\{b_0\} \rightarrow B(b_0)$

$S(a_0:t_{a_0}, b_0:t_{b_0}, c_0:t_{c_0})$

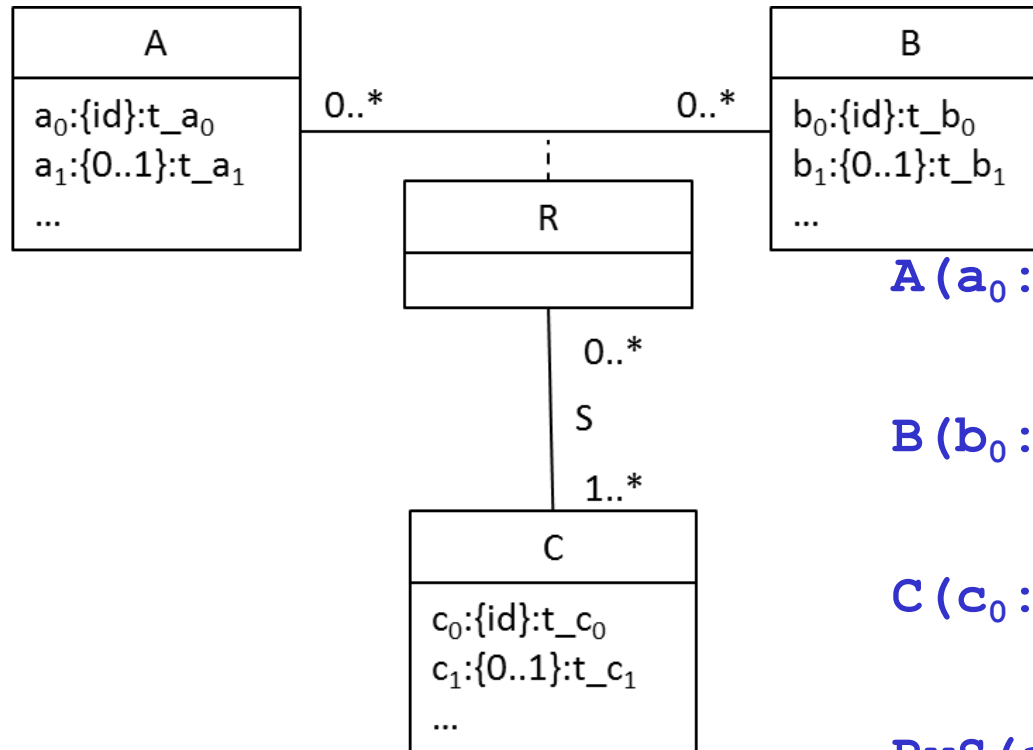
CP: $\{a_0, b_0, c_0\}$

CAj: $\{a_0, b_0\} \rightarrow R(a_0, b_0)$

CAj: $\{c_0\} \rightarrow C(c_0)$

RI1: No pot existir una tupla en R tal que el valor de (a_0, b_0) , no aparega en els atributs (a_0, b_0) de S.

3 Transf. Associació (quan s'associa amb altres classes)



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$

CP: $\{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$

CP: $\{b_0\}$

$C(c_0:t_{c_0}, c_1:t_{c_1}, \dots)$

CP: $\{c_0\}$

$RyS(a_0:t_{a_0}, b_0:t_{b_0}, c_0:t_{c_0})$

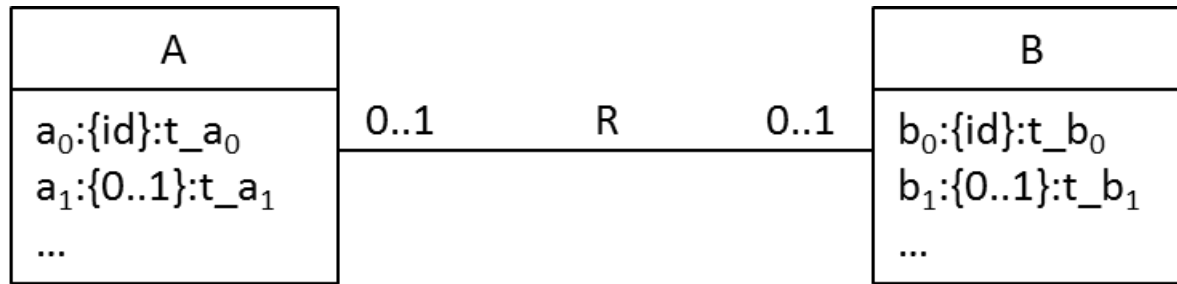
CP: $\{a_0, b_0, c_0\}$

$CAj: \{a_0\} \rightarrow A(a_0)$

$CAj: \{b_0\} \rightarrow B(b_0)$

$CAj: \{c_0\} \rightarrow C(c_0)$

3 Transf. d'associacions reflexives 0..1:0..1



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots)$

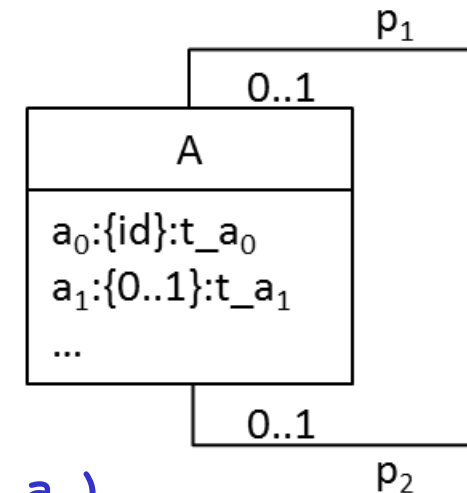
$CP: \{a_0\}$

$B(b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0})$

$CP: \{b_0\}$

$\acute{U}nico: \{a_0\}$

$CAj: \{a_0\} \rightarrow A(a_0)$



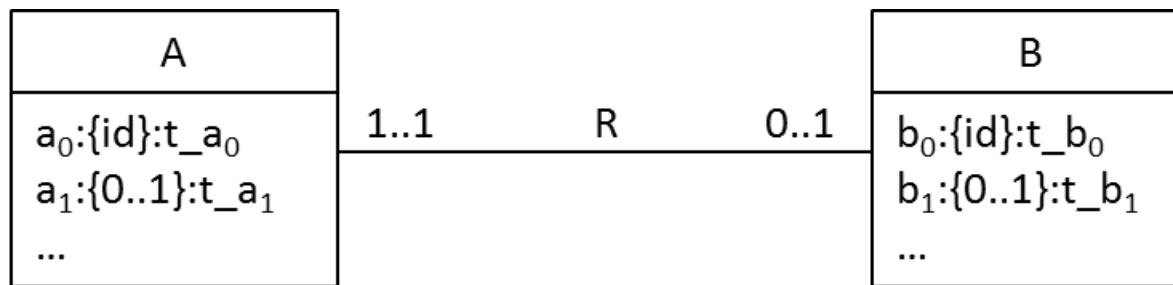
$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, a_{0_p2}:t_{a_0})$

$CP: \{a_0\}$

$\acute{U}nico: \{a_{0_p2}\}$

$CAj: \{a_{0_p2}\} \rightarrow A(a_0)$

3 Transf. d'associacions reflexives 0..1:1..1



A ($a_0:t_{a_0}, a_1:t_{a_1}, \dots$)
CP: { a_0 }

B ($b_0:t_{b_0}, b_1:t_{b_1}, \dots, a_0:t_{a_0}$)
CP: { b_0 }

Único: { a_0 }

VNN: { a_0 }

CAj: { a_0 } \rightarrow **A** (a_0)

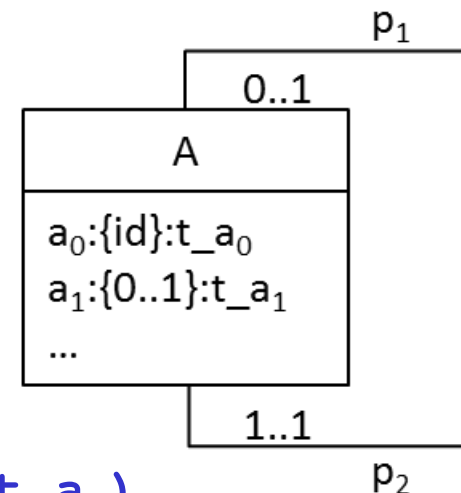
A ($a_0:t_{a_0}, a_1:t_{a_1}, \dots, a_{0_p_2}:t_{a_0}$)

CP: { a_0 }

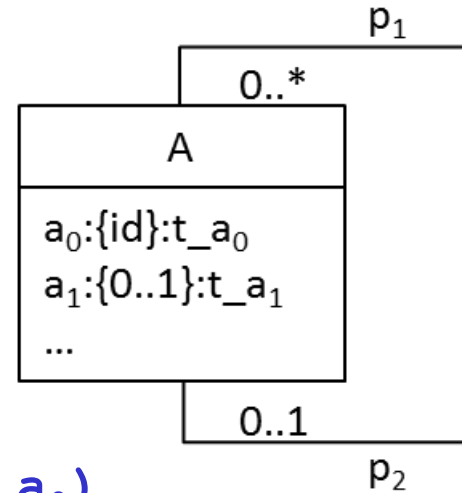
Único: { $a_{0_p_2}$ }

VNN: { $a_{0_p_2}$ }

CAj: { $a_{0_p_2}$ } \rightarrow **A** (a_0)



3 Transf. d'associacions reflexives $0..*:0..1$

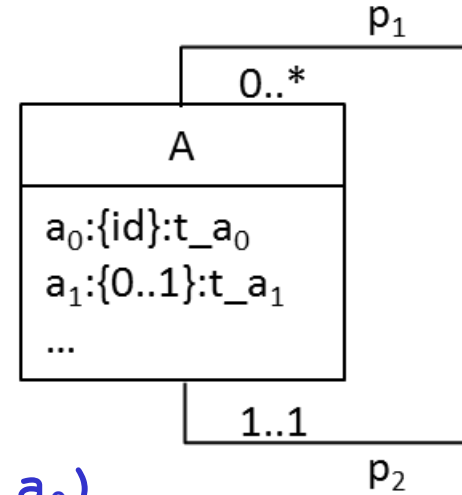


$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, a_{0_p_2}:t_{a_0})$

$CP:\{a_0\}$

$CAj:\{a_{0_p_2}\} \rightarrow A(a_0)$

3 Transf. d'associacions reflexives 0..*:1..1



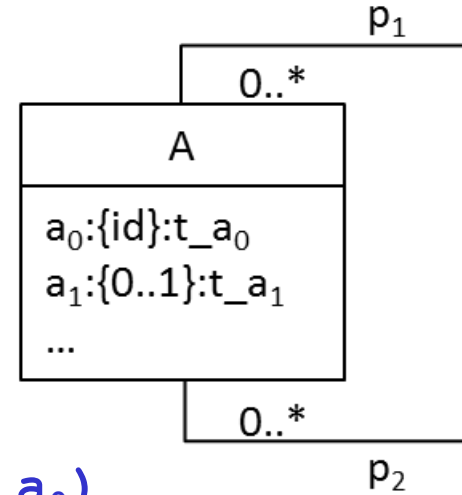
$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, a_{0_p_2}:t_{a_0})$

$CP:\{a_0\}$

$CAj:\{a_{0_p_2}\} \rightarrow A(a_0)$

$VNN:\{a_{0_p_2}\}$

3 Transf. d'associacions reflexives $0..*:0..*$



$A(a_0:t_{a_0}, a_1:t_{a_1}, \dots, a_{0_p_2}:t_{a_0})$

$CP:\{a_0\}$

$R(a_{0_p_1}:t_{a_0}, a_{0_p_2}:t_{a_0})$

$CP:\{a_{0_p_1}, a_{0_p_2}\}$

$CAj:\{a_{0_p_1}\} \rightarrow A(a_0)$

$CAj:\{a_{0_p_2}\} \rightarrow A(a_0)$

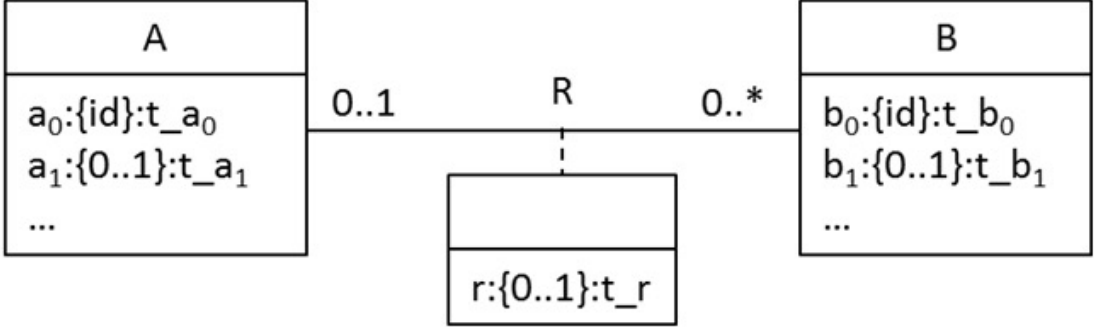
3 Transf. d'associacions directrius CAj

- La definició d'una clau alinea implica l'especificació de les directrius d'esborrament i d'actualització perquè el SGBD pugui, si es vol, restaurar la integritat referencial si aquesta es viola.
- Donat que les claus alienes permeten representar vincles entre els objectes del sistema, aquestes directrius permetran també la representació d'algunes restriccions d'integritat incloses en l'esquema conceptual o fins i tot algunes restriccions detectades en l'etapa d'anàlisi del sistema.
- De vegades hi ha diverses opcions raonables.
- En el cas de la directriu de modificació, en general, s'opta per la directriu en cascada per la qual cosa no s'especificarà.

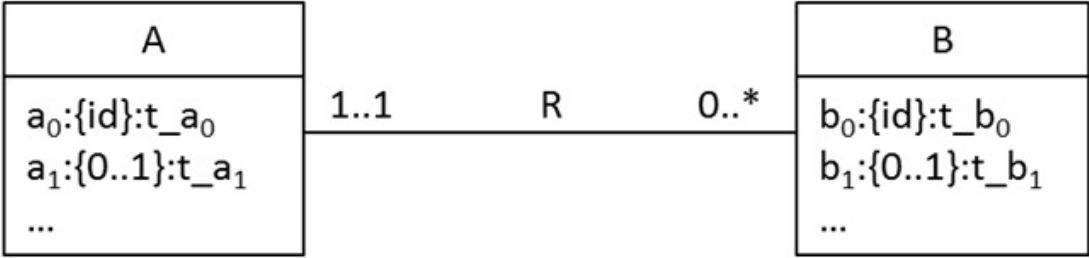
3 Transf. d'associacions directrius CAj

| | |
|--------------------|--|
| Esquema Conceptual | <pre> graph LR A["A a0:{id}:t_a0 a1:{0..1}:t_a1 ..."] B["B b0:{id}:t_b0 b1:{0..1}:t_b1 ..."] A --- 0..1 R 0..* B </pre> |
| Esquema Relacional | <pre> A(a0: t_a0, a1: t_a1, ...) CP: {a0} B(b0: t_b0, b1: t_b1, ..., a0: t_a0) CP: {b0} CAj: {a0} → A BORRADO {NULOS RESTRICTIVO CASCADA} </pre> |
| Comentarios | <p>En este caso las tres directrices son aceptables y su elección dependerá del comportamiento que se desee que tenga el sistema ante borrados de tuplas de la relación A.</p> |

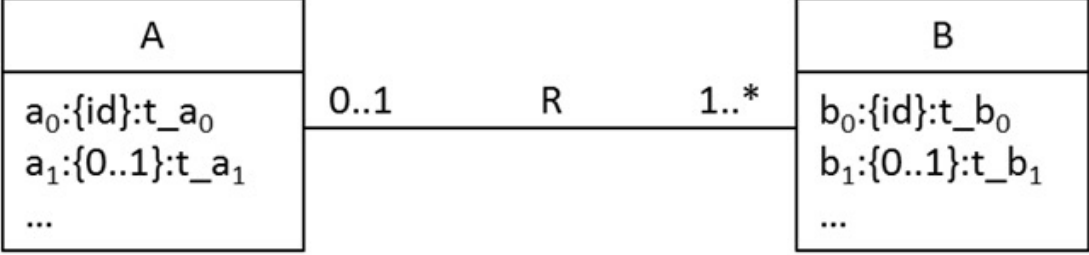
3 Transf. d'associacions directrius CAj

| | |
|---------------------------|--|
| Esquema Conceptual |  <pre> graph LR A["A a0:{id}:t_a0 a1:{0..1}:t_a1 ..."] B["B b0:{id}:t_b0 b1:{0..1}:t_b1 ..."] R["R r:{0..1}:t_r"] A --- 0..1 R R --- 0..* B </pre> |
| Esquema Relacional | <pre> A(a0: t_a0, a1: t_a1, ...) CP: {a0} B(b0: t_b0, b1: t_b1, ...) CP: {b0} R(b0: t_b0, a0: t_a0, r: t_r) CP: {b0} VNN: {a0} CAj: {a0} → A BORRADO {RESTRICTIVO CASCADA} CAj: {b0} → B BORRADO {RESTRICTIVO CASCADA} </pre> |
| Comentarios | <p>En este caso dado que relación <i>R</i> representa una relación entre <i>A</i> y <i>B</i> el borrado a nulos no es posible.</p> |

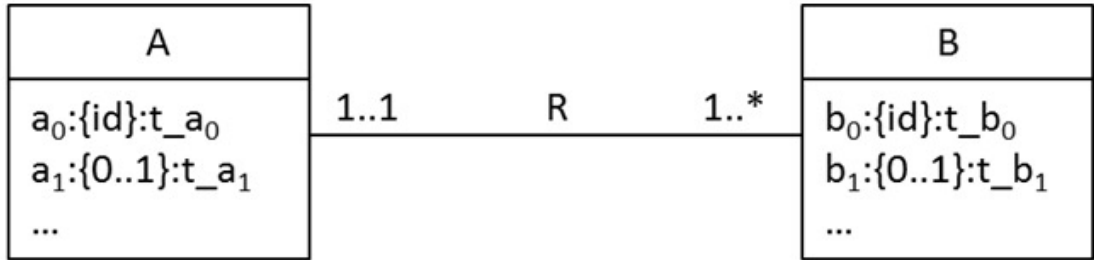
3 Transf. d'associacions directius CAj

| | |
|---------------------------|---|
| Esquema Conceptual |  <pre> graph LR A[A] --- 1..1 R[R] R --- 0..* B[B] subgraph A_attrs [A] a0[a0:{id}:t_a0] a1[a1:{0..1}:t_a1] end subgraph B_attrs [B] b0[b0:{id}:t_b0] b1[b1:{0..1}:t_b1] end </pre> |
| Esquema Relacional | <pre> A(a0: t_a0, a1: t_a1, ...) CP: {a0} B(b0: t_b0, b1: t_b1, ..., a0: t_a0) CP: {b0} CAj: {a0} → A BORRADO{RESTRICTIVO CASCADA} VNN: {a0} </pre> |
| Comentarios | <p>En este caso el borrado a nulos no es posible por la restricción de existencia <i>B</i> en <i>R</i> que a nivel lógico implica que toda tupla de <i>B</i> debe tener un valor no nulo en el atributo <i>a</i>₀.</p> |

3 Transf. d'associacions directius CAj

| | |
|---------------------------|---|
| Esquema Conceptual |  <pre> graph LR A[A] --- 0..1 R[R] R --- 1..* B[B] style A fill:#fff,stroke:#000,stroke-width:1px style B fill:#fff,stroke:#000,stroke-width:1px style R fill:none,stroke:none </pre> |
| Esquema Relacional | <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"> <p><u>A</u>(a₀: t_a₀, a₁: t_a₁, ...)</p> <p>CP: {a₀}</p> <p>B(b₀: t_b₀, b₁: t_b₁, ..., a₀: t_a₀)</p> <p>CP: {b₀}</p> <p><u>CAj</u>: {a₀} → A</p> <p>BORRADO {NULOS CASCADA}</p> </div> <div style="width: 35%;"> <p>RI:</p> <p>Todo valor de a₀ de la relación A aparece al menos una vez en el atributo a₀ de la relación B</p> </div> </div> |
| Comentarios | <p>En este caso el borrado restrictivo no es posible por la restricción de existencia de la entidad A en R que a nivel lógico implica que toda tupla de A aparece referenciada en B.</p> |

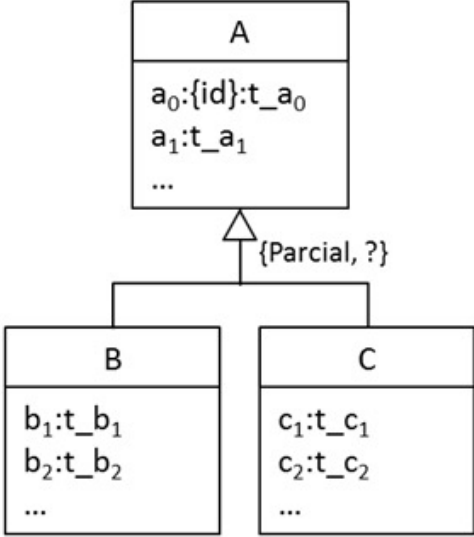
3 Transf. d'associacions directius CAj

| | |
|---------------------------|--|
| Esquema Conceptual |  <pre> graph LR A["A a0:{id}:t_a0 a1:{0..1}:t_a1 ..."] B["B b0:{id}:t_b0 b1:{0..1}:t_b1 ..."] A --- 1..1 R 1..* B </pre> |
| Esquema Relacional | <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"> <p><u>A</u>(a₀: t_a₀, a₁: t_a₁, ...)</p> <p>CP: {a₀}</p> <p>B(b₀: t_b₀, b₁: t_b₁, ..., a₀: t_a₀)</p> <p>CP: {b₀}</p> <p><u>CAj</u>: {a₀} → A</p> <p>BORRADO CASCADA</p> <p>VNN: {a₀}</p> </div> <div style="width: 35%;"> <p>RI:</p> <p>Todo valor de a₀ de la relación A aparece al menos una vez en el atributo a₀ de la relación B.</p> </div> </div> |
| Comentarios | <p>En este caso el borrado restrictivo no es posible por la restricción de existencia de la entidad A en R que a nivel lógico implica que toda tupla de A debe aparecer referenciada en B. El borrado a nulos tampoco es posible por la restricción de existencia B en R que a nivel lógico implica que toda tupla de B debe tener un valor no nulo en el atributo a₀.</p> |

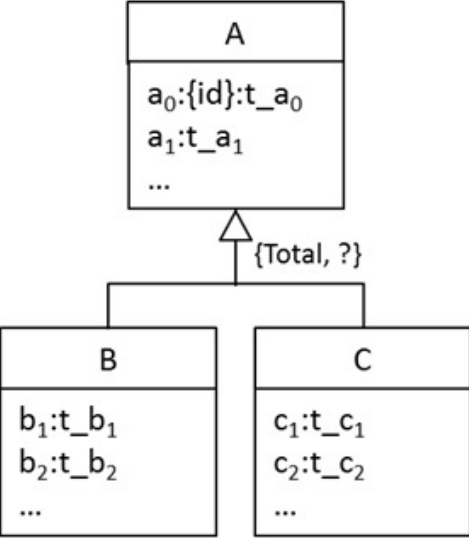
3 Transf. d'associacions directrius CAj

| | | |
|--------------------|--|---|
| Esquema Conceptual | <div><div><div>A</div><div>$a_0:\{id\}:t_{a_0}$ $a_1:\{0..1\}:t_{a_1}$...</div></div><div>$0..*$ R $1..*$</div><div><div>B</div><div>$b_0:\{id\}:t_{b_0}$ $b_1:\{0..1\}:t_{b_1}$...</div></div></div> | |
| Esquema Relacional | <div><div>$A(a_0: t_{a_0}, a_1: t_{a_1}, ...)$ CP: {a_0}</div><div>$B(b_0: t_{b_0}, b_1: t_{b_1}, ...)$ CP: {b_0}</div><div>$R(a_0: t_{a_0}, b_0: t_{b_0})$ CP: {a_0, b_0}</div><div>$CAj: \{a_0\} \rightarrow A$ BORRADO CASCADA</div><div>$CAj: \{b_0\} \rightarrow B$ BORRADO {CASCADA RESTRICTIVO}</div></div> | <div><div>RI:</div><div>Todo valor de a_0 de la relación A aparece al menos una vez en el atributo a_0 de la relación R</div></div> |
| Comentarios | En este caso el borrado restrictivo de A no es posible por la restricción de existencia que tiene en R que a nivel lógico implica que toda tupla de A debe aparecer referenciada en R . El borrado a nulos tampoco es posible porque la clave ajena forma parte de la clave primaria de R . | |

3 Transf. d'associacions directrius CAj

| | |
|---------------------------|--|
| <p>Esquema Conceptual</p> |  |
| <p>Esquema Relacional</p> | <pre> A(a0: t_a0, a1: t_a1, ...) CP: {a0} B(a0: t_a0, b1: t_b1, b2: t_b2, ...) CP: {a0} CAj: {a0} → A BORRADO {RESTRICTIVO CASCADA} C(a0: t_a0, c1: t_c1, c2: t_c2, ...) CP: {a0} CAj: {a0} → A BORRADO {RESTRICTIVO CASCADA} </pre> |
| <p>Comentarios</p> | <p>En este caso el borrado a nulos no es posible ya que el atributo a_0 es la clave primaria de la relación B. El borrado en cascada representa el borrado de ocurrencias de las entidades especializadas cuando se borra en la entidad general. El borrado restrictivo supone que sólo se puede borrar una ocurrencia de la entidad general si no está especializada.</p> |

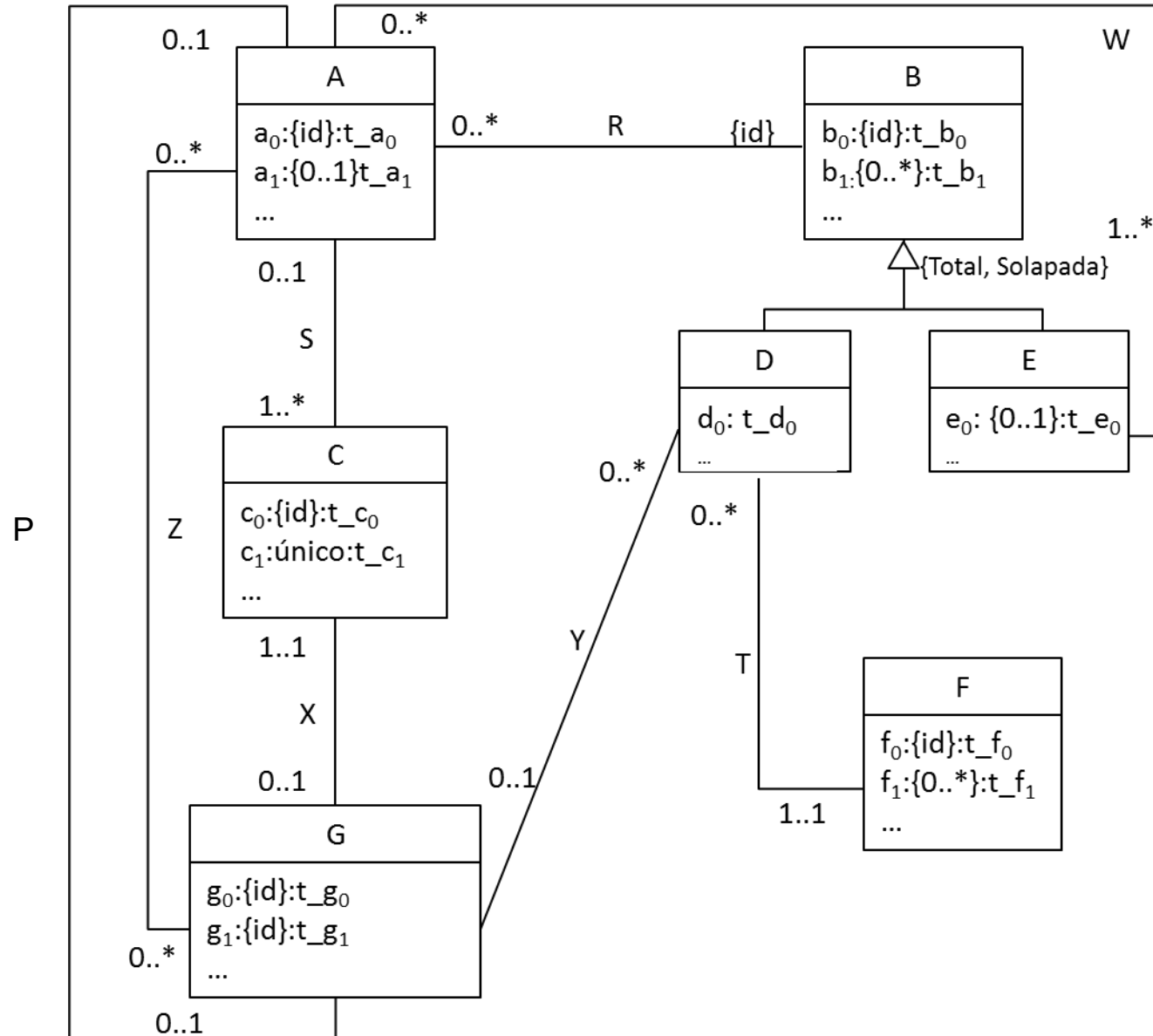
3 Transf. d'associacions directrius CAj

| | | |
|---------------------------|---|--|
| <p>Esquema Conceptual</p> |  | |
| <p>Esquema Relacional</p> | <pre> A(a0: t_a1, a1: t_a1,...) CP: {a0} B(a0: t_a0, b1: t_b1, b2: t_b2,...) CP: {a0} CAj: {a0} → A BORRADO CASCADA C(a0: t_a0, c1: t_c1, c2: t_c2,...) CP: {a0} CAj: {a0} → A BORRADO CASCADA </pre> | <p>RI:</p> <p>Todo valor de a_0 de la relación A aparece al menos una vez en el atributo a_0 de la relación B o en el atributo a_0 de la relación C.</p> |
| <p>Comentarios</p> | <p>En este caso el borrado a nulos no es posible ya que el atributo a_0 es la clave primaria de la relación B. El borrado restrictivo no es posible ya que toda ocurrencia de la entidad general debe estar en al menos una de las especializaciones.</p> | |

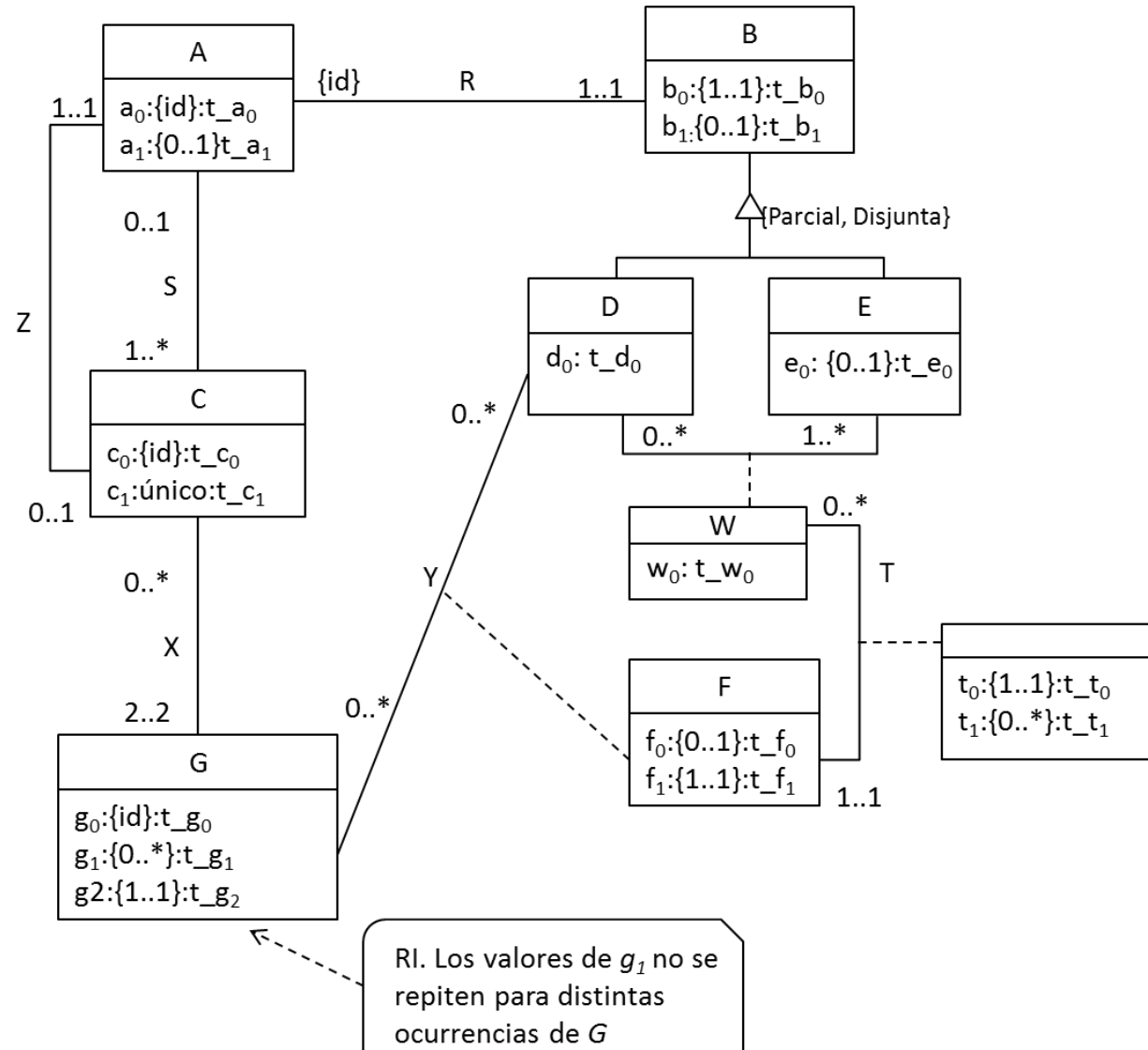
3 Transf. d'associacions directrius CAj

| Esquema Conceptual | <table><tr><th>A</th></tr><tr><td>$a_0: \{id\}: t_{a_0}$ $a_1: \{0..1\}: t_{a_1}$ $a_2: \{0..*\}: t_{a_2}$</td></tr></table> | A | $a_0: \{id\}: t_{a_0}$ $a_1: \{0..1\}: t_{a_1}$ $a_2: \{0..*\}: t_{a_2}$ |
|--|---|---|--|
| A | | | |
| $a_0: \{id\}: t_{a_0}$ $a_1: \{0..1\}: t_{a_1}$ $a_2: \{0..*\}: t_{a_2}$ | | | |
| Esquema Relacional | $A(a_0: t_{a_1}, a_1: t_{a_1})$ CP: {a0} $A2(a_0: t_{a_0}, a_2: t_{a_2})$ CP: {a0} <u>CAj</u> : {a0} → A BORRADO CASCADA | | |
| Comentarios | En este caso la única directriz razonable es el borrado en cascada ya que las tuplas de A2 almacenan el valor de una propiedad de A y por lo tanto deben desaparecer cuando desaparezca la tupla a la que hacen referencia. | | |

Exemple (diferent al dels apunts)



4 Exemple (diferent al dels apunts)



5 Teoria de la Normalització

- Té com a objectiu garantir que les relacions obtingudes a partir del procés de disseny lògic acomplisquen les propietats del model relacional.
- Nosaltres utilitzarem les formes normal com un mecanisme de validació/correcció de la transformació obtinguda en el disseny lògic.

5 Teoria de la Normalització

- **Conceptes Previs:**
 - Dependència Funcional (completa o no)
 - Diagrama de Dependències Funcionals
 - Clau d'una relació
 - Atribut Primer

5 Teoria de la Normalització

- **Conceptes Previs:**

- Dependència Funcional

- $A = \{A_1, \dots, A_n\}$ conjunt d'atributs de l'esquema R

Una dependència funcional entre X i Y ($X \subseteq A$, $Y \subseteq A$, $X \neq Y$)

$X \rightarrow Y \Rightarrow$ per a qualsevol parell de tuples possibles de R, t_1 i t_2 , es compleix que si $t_1[X]$ és igual a $t_2[X]$, aleshores $t_1[Y]$ és igual a $t_2[Y]$.



Per a cada valor de X, Y sols pot tindre un valor possible

5 Teoria de la Normalització

- **Conceptes Previs:**
 - Dependència Funcional **completa**

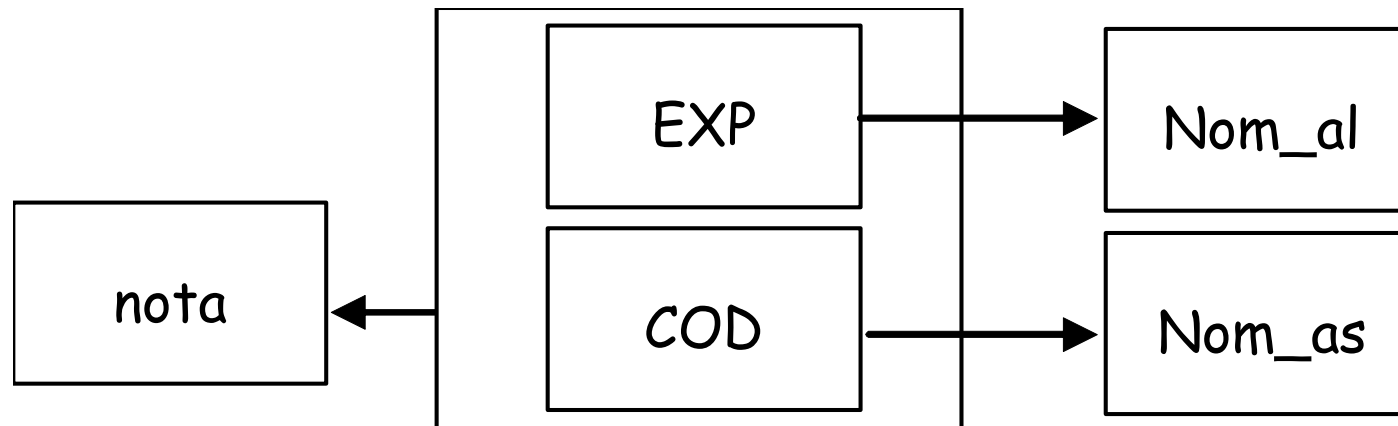
Una dependència funcional entre dos conjunts d'atributs $X \rightarrow Y$ és completa si l'eliminació de qualsevol atribut A_i de X fa que la dependència deixi d'existir, és a dir, si $\forall A_i / A_i \in X$ s'acompleix que Y no depèn funcionalment de $(X - \{A_i\})$.

5 Teoria de la Normalització

- **Conceptes Previs:**

- **Diagrama de Dependències Funcionals**

Representació gràfica de les dependències. Utilitzen caixes per emmarcar els atributs o conjunts d'atributs i fletxes per denotar la dependència funcional. Normalment sols es representen les dependències funcionals completes.



5 Teoria de la Normalització

- **Conceptes Previs:**

- **Clau d'una relació**

Siga R un esquema de relació amb el conjunt d'atributs $A = \{A_1, A_2, \dots, A_n\}$, i siga C un subconjunt d'atributs d'eixe esquema ($C \subseteq A$); es diu que C és una clau de R si C és la clau primària de R o bé si C té una restricció d'unicitat.



Tots els atributs de la relació que no formen part de la clau depenen funcionalment de cada clau de la relació

5 Teoria de la Normalització

- **Conceptes Previs:**

- **Atribut Primer**

Siga R un esquema de relació amb el conjunt d'atributs $A = \{A_1, A_2, \dots, A_n\}$, direm que un atribut A és primer si forma part d'alguna clau de R

5 Teoria de la Normalització

- **1^a Forma Normal**

- Una Relació està en 1FN si els seus atributs solament tenen valors atòmics (simples e indivisibles).
 - Problemes quan s'utilitzen relacions que no estan en 1FN: Cal utilitzar operadors associats als tipus de dades complexos (llistes, conjunts, registres,...)

CP: {vcod}

| vcod | nombre | teléfonos | dir |
|------|--------|--|-------------------|
| V1 | Pepe | (96 3233258, 964 523844, 979 568987, 987 456123) | Paz 7, Valencia |
| V2 | Juan | (96 3852741, 910 147258) | Eolo 3, Castellón |
| V3 | Eva | (987 456 312) | F. Lorca 2, Utiel |

Conjunt

Registre

5 Teoria de la Normalització

- **1^a Forma Normal**

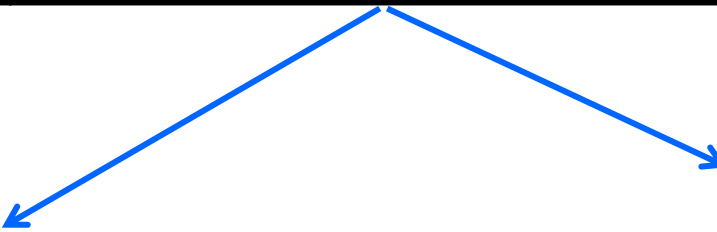
- Pas a 1FN:

- si R té un atribut que és un conjunt, l'eliminem de la relació i definirem una nova relació amb eixe atribut i buscarem la clau primària.

5 Teoria de la Normalització

- 1ª Forma Normal

| vcod | nombre | teléfonos | dir |
|------|--------|--|-------------------|
| V1 | Pepe | (96 3233258, 964 523844, 979 568987, 987 456123) | Paz 7, Valencia |
| V2 | Juan | (96 3852741, 910 147258) | Eolo 3, Castellón |
| V3 | Eva | (987 456 312) | F. Lorca 2, Utiel |



| vcod | nombre | dir |
|------|--------|-------------------|
| V1 | Pepe | Paz 7, Valencia |
| V2 | Juan | Eolo 3, Castellón |
| V3 | Eva | F. Lorca 2, Utiel |

| vcod | teléfonos |
|------|-------------|
| V1 | 96 3233258 |
| V2 | 96 3852741 |
| V3 | 987 456 312 |
| V1 | 964 523844 |
| V1 | 979 568987 |
| V1 | 987 456123 |
| V2 | 910 147258 |

5 Teoria de la Normalització

- **1ª Forma Normal**

Proveedor(vcod, nombre, teléfonos, dir)

CP: {vcod}

Proveedor(vcod, nombre, dir)

CP: {vcod}

Listín(vcod, teléfono)

CP: {teléfono}vcod}

CAj: {vcod} → Proveedor

VNN: {vcod}

Si els telèfons no es poden
compartir
Si els telèfons es poden
compartir

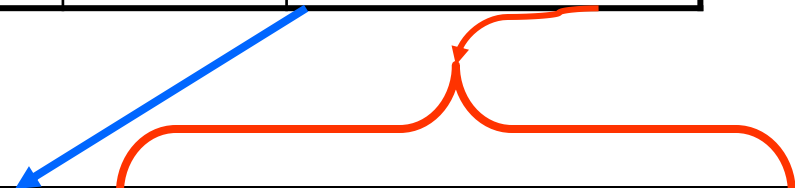
5 Teoria de la Normalització

- **1ª Forma Normal**

- Paso a 1FN:

- si R té un atribut que és un registre cal substituir-lo pels camps del registre.

| vcod | nombre | dir |
|-------------|---------------|-------------------|
| V1 | Pepe | Paz 7, Valencia |
| V2 | Juan | Eolo 3, Castellón |
| V3 | Eva | F. Lorca 2, Utiel |



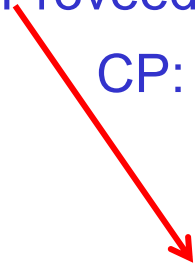
| vcod | nombre | calle | número | ciudad |
|-------------|---------------|--------------|---------------|---------------|
| V1 | Pepe | Paz | 7 | Valencia |
| V2 | Juan | Eolo | 3 | Castellón |
| V3 | Eva | F. Lorca | 2 | Utiel |

5 Teoria de la Normalització

- **1ª Forma Normal**

Proveedor(vcod, nombre, dir)

CP: {vcod}



Proveedor(vcod, nombre, calle, número, ciudad)

CP: {vcod}

5 Teoria de la Normalització

- **2ª Forma Normal**

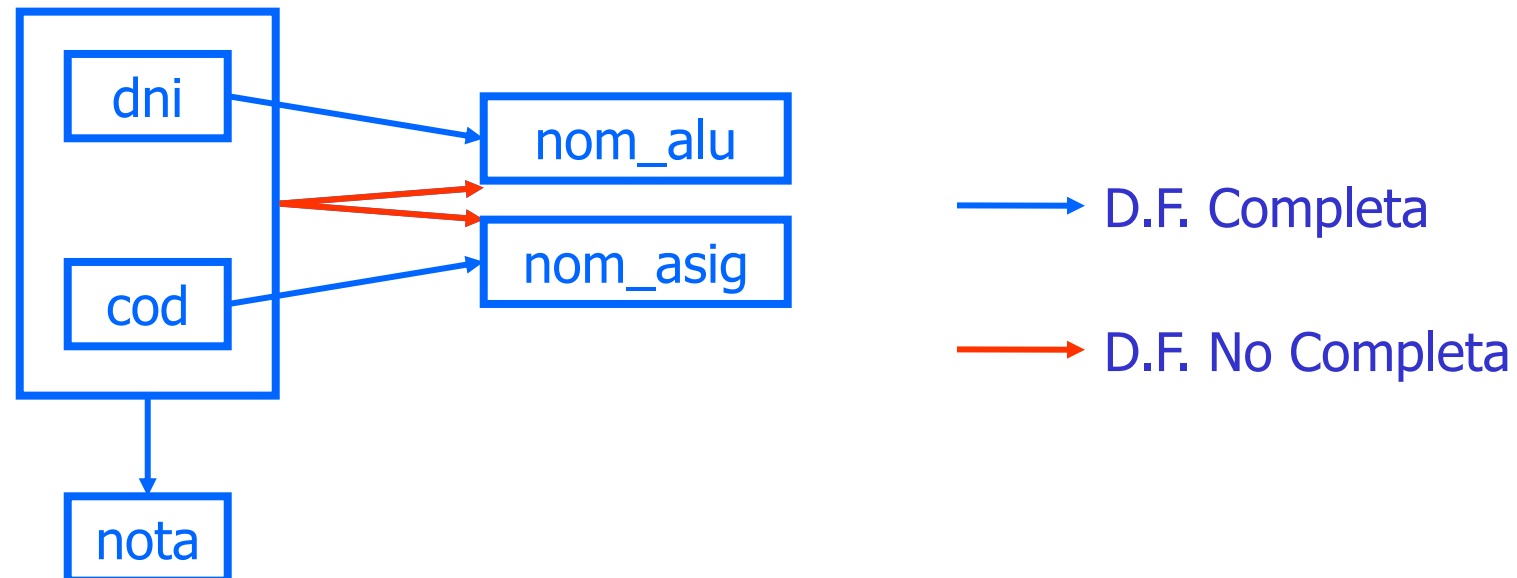
- Una Relació està en 2FN si està en 1FN i tot atribut no-primer depèn funcionalment de forma completa **de tota clau de R.**
- Problemes quan utilitzem relacions que no estan en 2FN:
 - Existeixen redundàncies, amb la qual cosa es complica la manipulació de la informació.
 - No és fàcil inserir ni esborrar

5 Teoria de la Normalització

- 2ª Forma Normal

CP: {dni,cod}

| dni | nom_alu | cod | nom_asig | nota |
|-----|---------|-----|--------------------------|------|
| 1 | Pepe | DBD | Diseño de Bases de Datos | 6 |
| 1 | Pepe | BDA | Bases de Datos | 7 |
| 2 | Juana | DBD | Diseño de Bases de Datos | 7 |
| 2 | Juana | BDA | Bases de Datos | 5 |



5 Teoria de la Normalització

- **2^a Forma Normal**

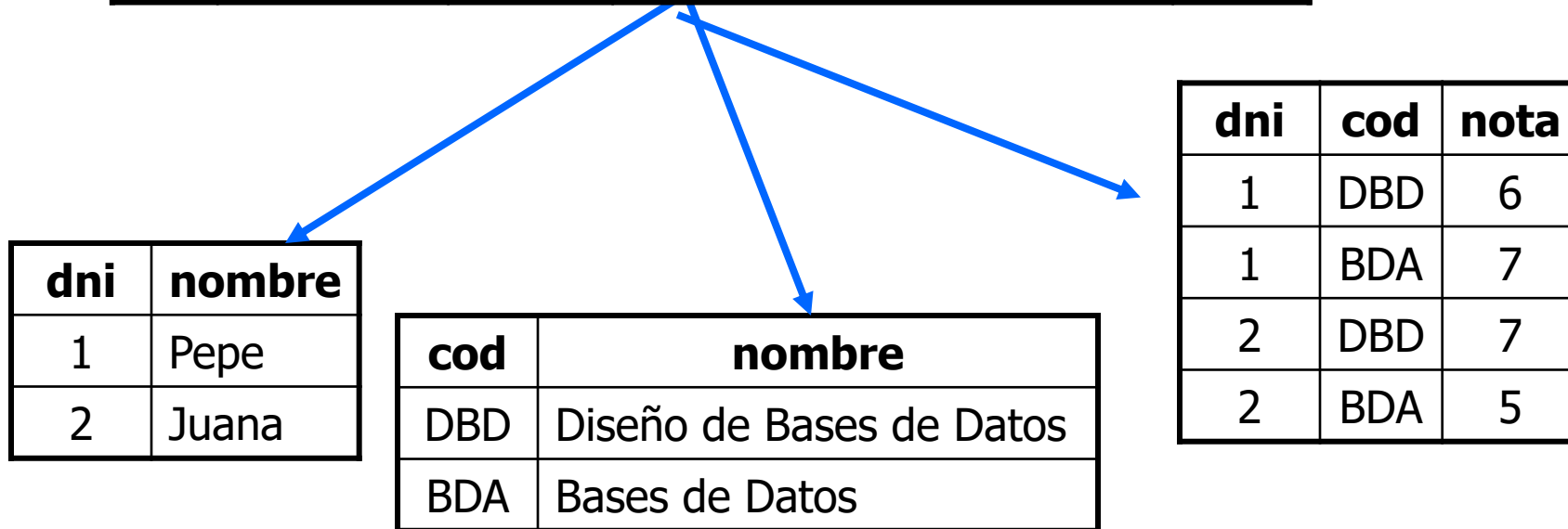
- Pas a 2FN:

- La clau consta de més d'un atribut i existex algun atribut no-primer que no depèn completament de la clau primària.
- Cal dividir la relació original en relacions per eliminar eixes dependències no completes.

5 Teoria de la Normalització

- 2ª Forma Normal

| dni | nom_alu | cod | nom_asig | nota |
|------------|----------------|------------|--------------------------|-------------|
| 1 | Pepe | DBD | Diseño de Bases de Datos | 6 |
| 1 | Pepe | BDA | Bases de Datos | 7 |
| 2 | Juana | DBD | Diseño de Bases de Datos | 7 |
| 2 | Juana | BDA | Bases de Datos | 5 |



| dni | nombre |
|------------|---------------|
| 1 | Pepe |
| 2 | Juana |

| cod | nombre |
|------------|--------------------------|
| DBD | Diseño de Bases de Datos |
| BDA | Bases de Datos |

| dni | cod | nota |
|------------|------------|-------------|
| 1 | DBD | 6 |
| 1 | BDA | 7 |
| 2 | DBD | 7 |
| 2 | BDA | 5 |

5 Teoria de la Normalització

- **2ª Forma Normal**

Matriculado(dni, cod, nom_alu, nom_asig)

CP: {dni,cod}

→ Alumno(dni, nom_alu)

CP: {dni}

→ Asignatura(cod, nom_asig)

CP: {cod}

→ Matriculado(dni, cod)

CP: {dni,cod}

CAj: {dni} → Alumno

CAj: {cod} → Asignatura

5 Teoria de la Normalització

- **3^a Forma Normal**

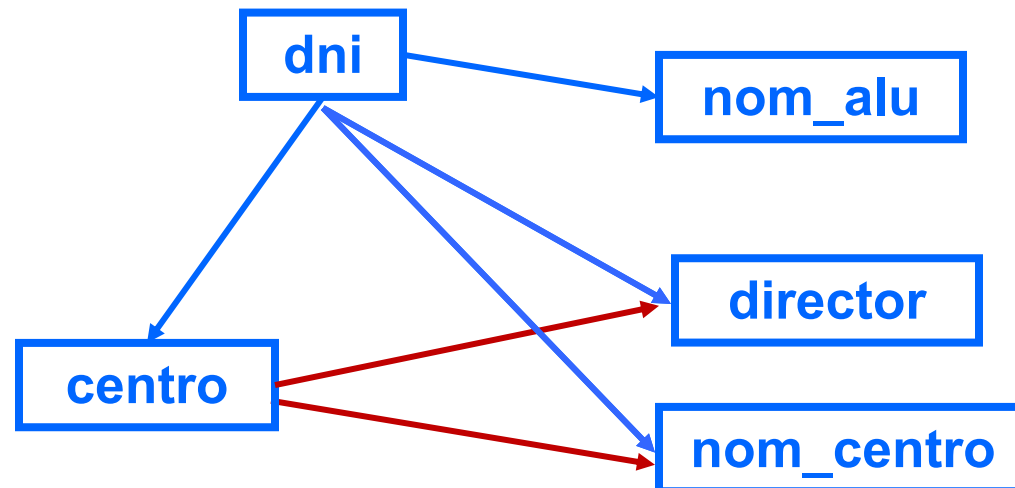
- Una Relació està en 3FN si està en 2FN i no existeixen dependències funcionals entre atributs no-primers.
- Problemes quan utilitzem relacions que no estan en 3FN
 - Existeixen redundàncies, amb la qual cosa es complica la manipulació de la informació.
 - No és fàcil inserir ni esborrar

5 Teoria de la Normalització

- 3^a Forma Normal

| dni | nom_alu | centro | nom_centro | director |
|------------|----------------|---------------|--------------------------------------|-----------------|
| 1 | Olga | EUI | Escuela Universitaria de Informática | Pepe |
| 2 | Juana | EUI | Escuela Universitaria de Informática | Pepe |
| 3 | Ana | FI | Facultad de Informática | Eva |
| 4 | Juan | FI | Facultad de Informática | Eva |

CP: {dni}



D.F. entre atributs

No primers

(Dep. Transitives)

5 Teoria de la Normalització

- **3^a Forma Normal**

- Pas a 3FN:

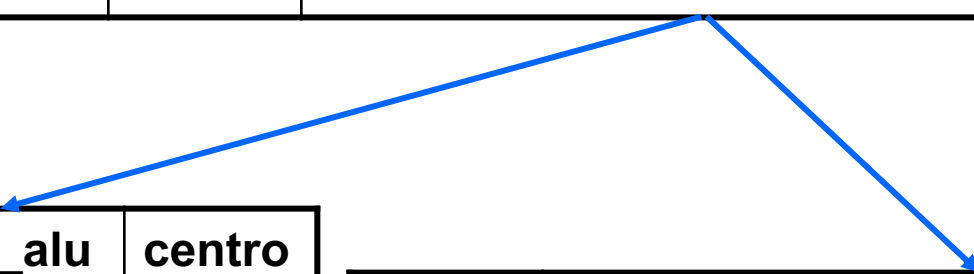
- Existeixen al menys un parell d'atributs no-primers que són dependents.
 - Traure l'atribut de la relació i crear-ne una nova que tinga com a clau primària l'atribut del qual depèn.

5 Teoria de la Normalització

- 3ª Forma Normal

| dni | nom_alu | centro | nom_centro | director |
|-----|---------|--------|--------------------------------------|----------|
| 1 | Olga | EUI | Escuela Universitaria de Informática | Pepe |
| 2 | Juana | EUI | Escuela Universitaria de Informática | Pepe |
| 3 | Ana | FI | Facultad de Informática | Eva |
| 4 | Juan | FI | Facultad de Informática | Eva |

CP: {dni}



| dni | nom_alu | centro |
|-----|---------|--------|
| 1 | Olga | EUI |
| 2 | Juana | EUI |
| 3 | Ana | FI |
| 4 | Juan | FI |

| centro | nom_centro | director |
|--------|--------------------------------------|----------|
| EUI | Escuela Universitaria de Informática | Pepe |
| FI | Facultad de Informática | Eva |

5 Teoria de la Normalització

- **3^a Forma Normal**

Alumno(dni, nom_alu, centro, nom_centro, director)

CP: {dni}

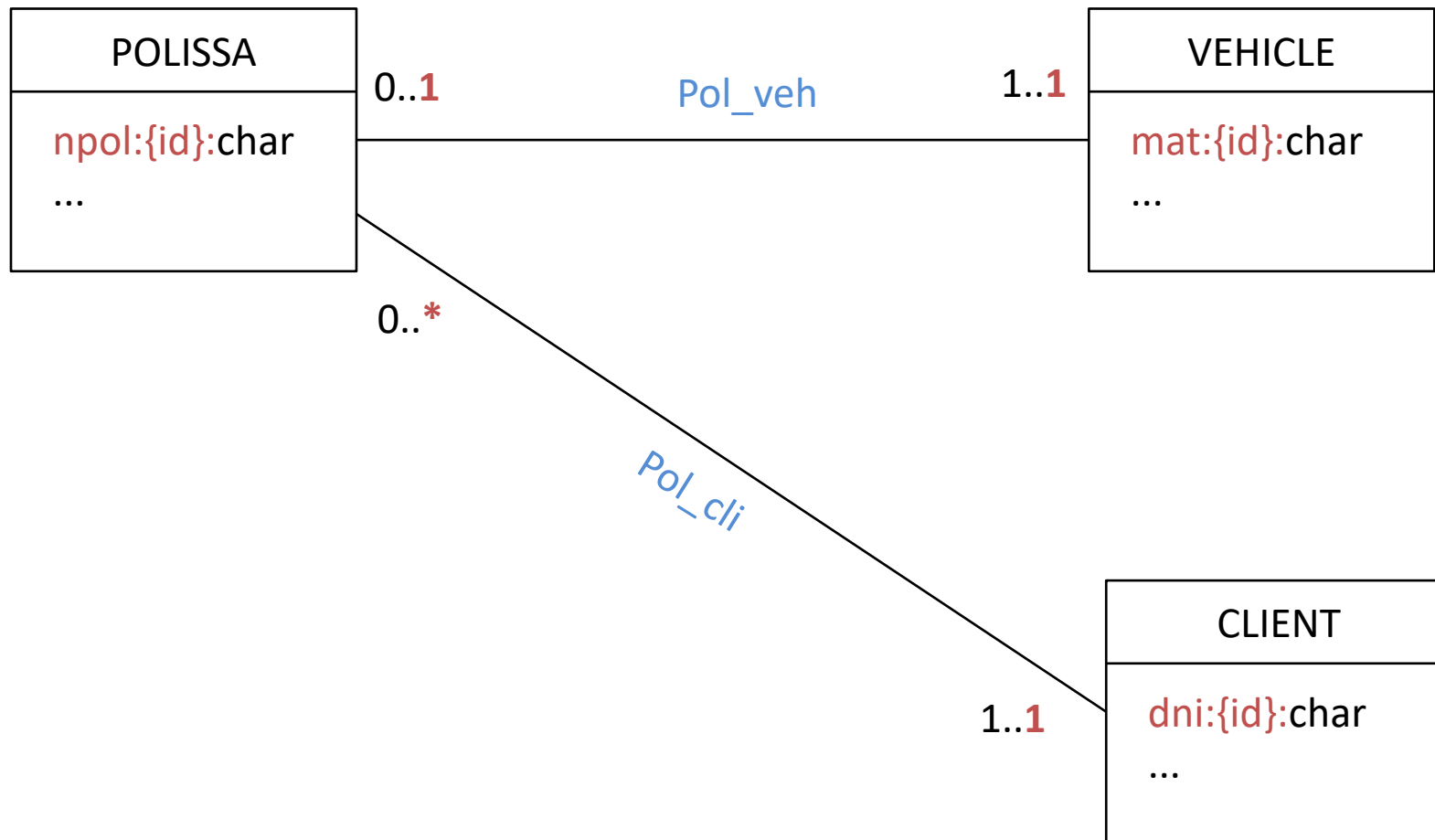
→ Alumno(dni, nom_alu, centro)

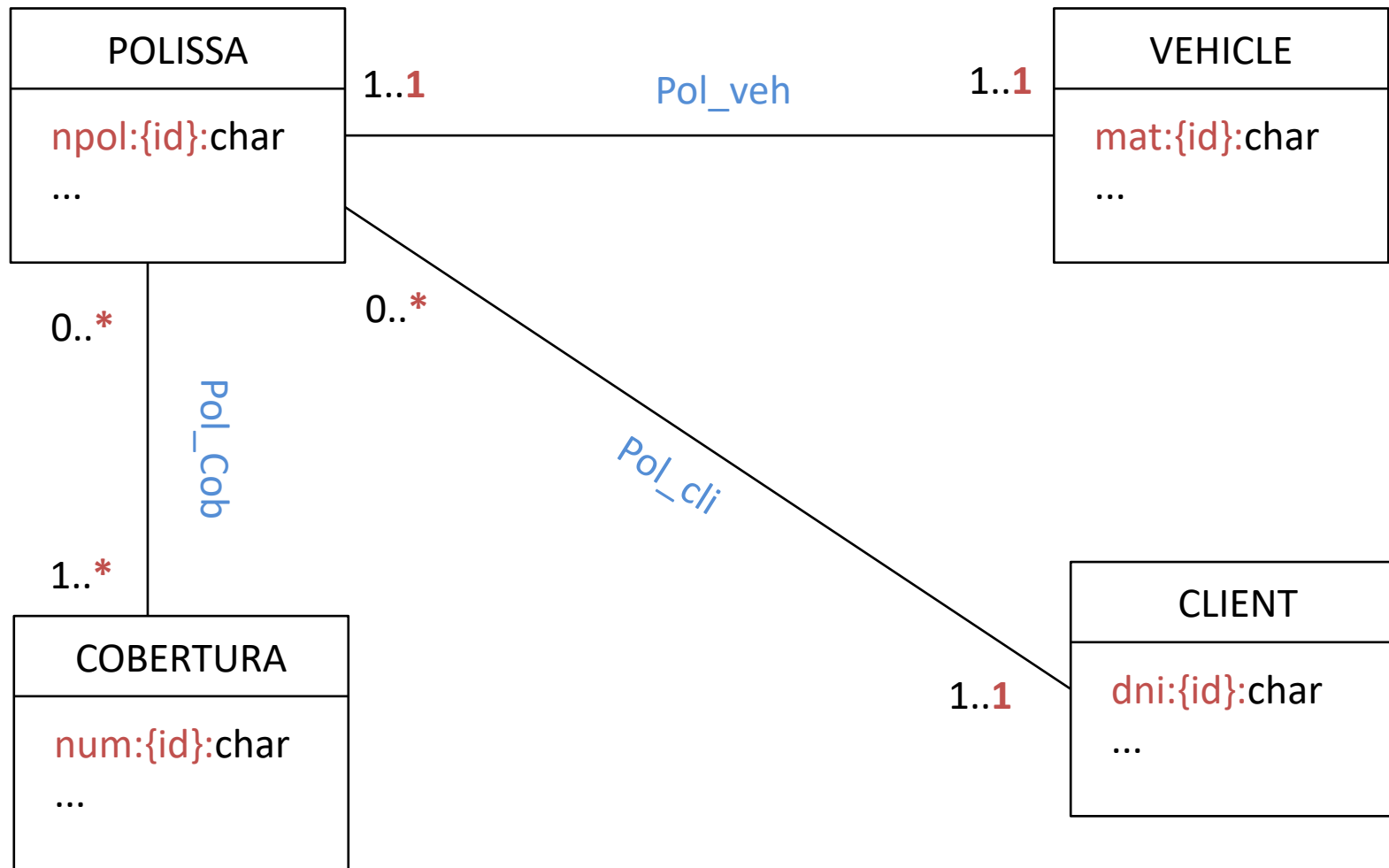
CP: {dni}

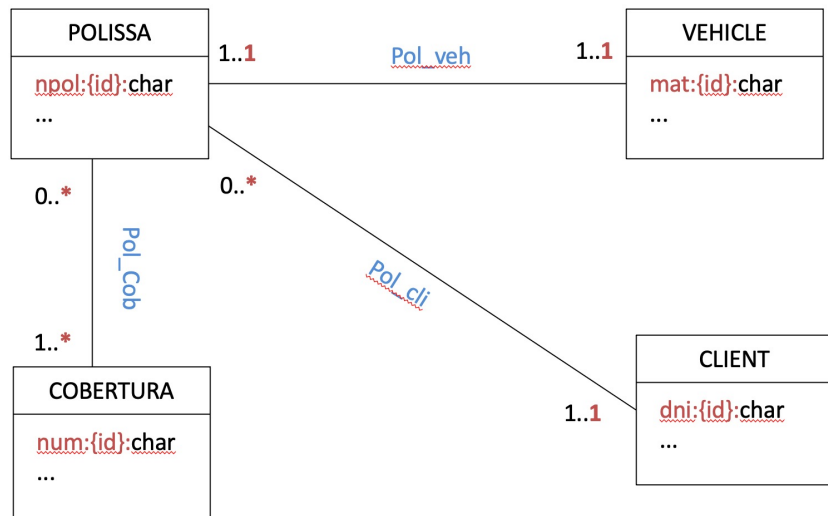
CAj: {centro} → Centro_universitario

→ Centro_universitario(centro, nom_centro, director)

CP: {centro}

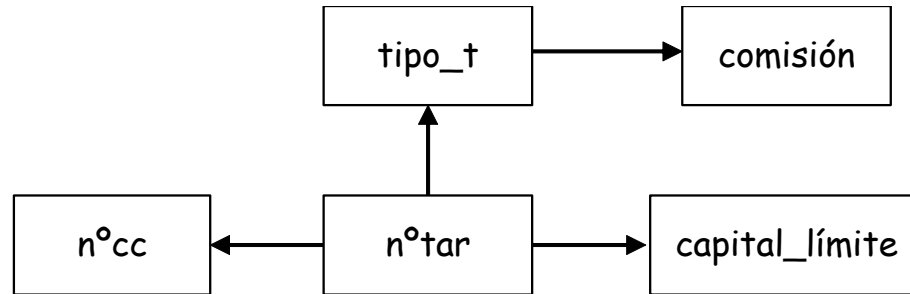






Ex-1:

Tarjeta(n°cc: cadena(15), n°tar: cadena(15), tipo_t: cadena(10), comisión: real, capital_límite: real)
CP: {n°tar}
CAj: {n°cc} → Cuenta
VNN: { n°cc, tipo_t, comisión, capital_límite }

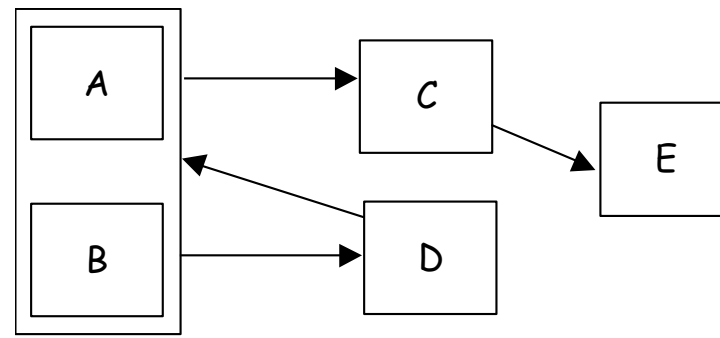


Tarjeta(n°cc: cadena(15), n°tar: cadena(15), tipo_t: cadena(10), capital_límite: real)
CP: {n°tar}
CAj: {n°cc} → Cuenta
VNN: {n°cc, tipo_t, capital_límite} Caj: {tipo_t} → Comisión
Comisión(tipo_t: cadena(10), comisión: real)
CP: {tipo_t}
VNN: {comisión}

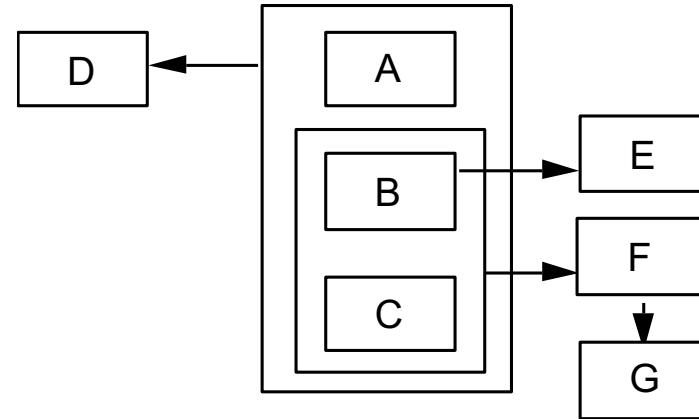
FORMES NORMALS

- **1FN:** si els seus atributs solament tenen valors atòmics (simples i indivisibles).
- **2FN:** si està en 1FN i tot atribut no-primer depèn funcionalment de forma completa **de tota clau de R.**
- **3FN:** si està en 2FN i no existeixen dependències funcionals entre atributs no-primers.

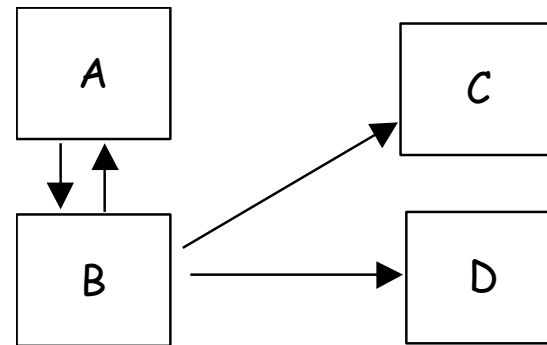
E1



E2



E3



Siga el següent esquema de relació:

R(A: sencer, B: text, C: sencer, D: text, E: text, F: text, G: text)

CP: {A, B}

VNN: {C, D, E, F, G}

A partir de les dependències que apareixen a continuació, transformeu la relació a un conjunt de relacions en tercera forma normal.

$\{G\} \rightarrow \{E\}$

$\{G\} \rightarrow \{F\}$

$\{A\} \rightarrow \{D\}$

$\{A\} \rightarrow \{G\}$

2. **(0'5 puntos)** Sea el siguiente esquema de relación:

R (A: char, B: char, C: conjunto de H:int, D: char, E: char, F: char, G: char)

CP: {A, B} VNN: {C, D, E, F, G}

A partir de las dependencias que aparecen a continuación, transforme la relación a un conjunto de relaciones en tercera forma normal.

$\{H\} \rightarrow \{A, B\}$

$\{B\} \rightarrow \{D\}$

$\{E\} \rightarrow \{F\}$

$\{F\} \rightarrow \{G\}$

| | |
|--|---|
| R(A: char, B: char, E: char) CP:{A,B} VNN:{E} CA:{E} → R2 CA:{B} → R3 Todo valor de (A,B) de R debe existir en R1 | R1 (A: char, B: char, H: int) CP:{H} CA: {A,B} → R VNN:{A,B} |
| R2(E: char, F:char) CP:{E} CA:{F} → R4 VNN:{F} | R3(B: char, D:char) CP:{B} VNN:{D} |
| R4(F: char, G: char) CP:{F} VNN:{G} | |