

## TEMA 3

# DISSENY DE B.D: EL MODEL RELACIONAL

### 1. Introducció

### 2. Conceptes bàsics del Model Relacional

#### 2.1. Models i Esquemes

#### 2.2. Elements del Model Relacional

#### 2.3. Altres consideracions

### 3. Model Relacional: transformació del E-R a taules

#### 3.1. Entitats i atributs

#### 3.2. Relacions binàries M:M, 1:M, 1:1

#### 3.3. Relacions unàries M:M, 1:M, 1:1

#### 3.4. Relacions ternàries

#### 3.5. Atributs multivalent

#### 3.6. Atributs multivalents compostos

#### 3.7. Restriccions d'Identificació (entitats dèbils)

#### 3.8. Restriccions d'Existència

#### 3.9. Agregacions

#### 3.10. Especialitzacions / Generalitzacions

### 4. Model Relacional: normalització

#### 4.1. Dependències funcionals

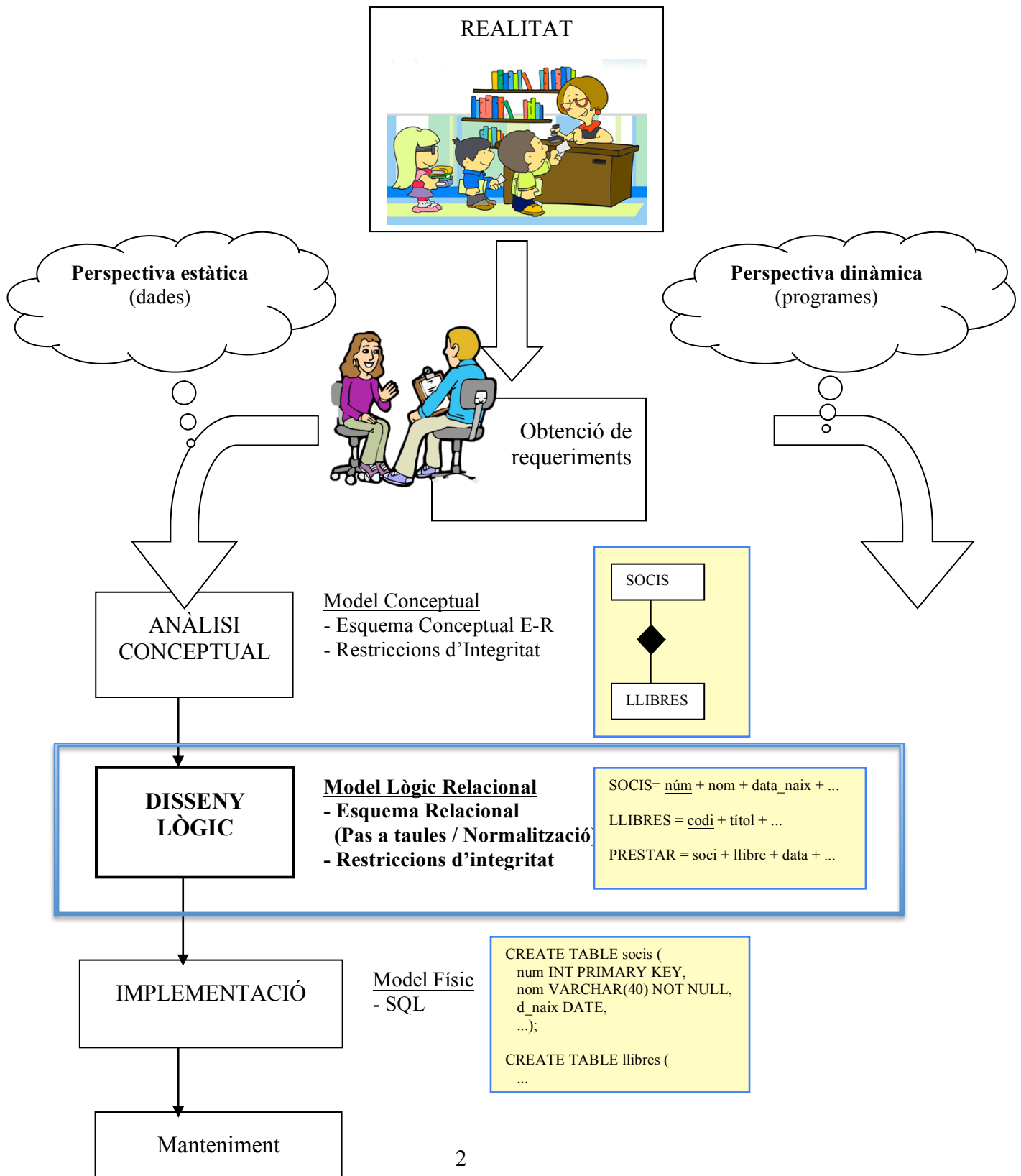
#### 4.2. Formes normals

#### 4.3. Esquema-resum de com normalitzar

#### 4.4. Resolució d'exercicis de normalització

## 1. Introducció

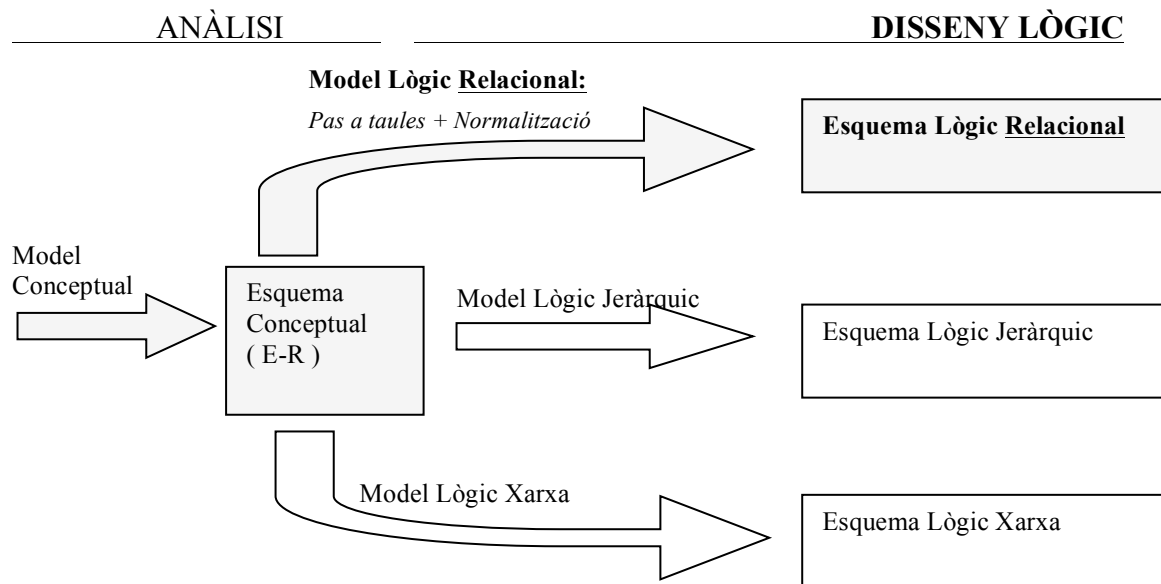
L'objectiu del Model Relacional és obtenir un conjunt de relacions ("taules") sense redundàncies i que satisfagen els requeriments de l'organització. Això ho aconseguirem transformant l'esquema E/R a taules. O bé, si no està fet l'E-R sinó que partim d'unes taules mal definides, les "corregirem" basant-nos en la teoria de la Normalització.



## 2. Conceptes bàsics del Model Relacional

### 2.1. MODELS I ESQUEMES

Durant la fase de *disseny lògic* transformarem l'Esquema Conceptual (Diagrama Entitat-Relació) en un Esquema Lògic. Com? Es pot fer de diferents maneres, anomenades *models lògics*.



És a dir, tradicionalment, tenim tres tipus de models lògics:

- Model Lògic Relacional (el que s'utilitza actualment)
- Model Lògic Jeràrquic
- Model Lògic Xarxa

Nosaltres utilitzarem el **Model Lògic Relacional**, en el qual, a partir d'un **esquema E-R**, obtindrem un **Esquema Lògic Relacional**.

Cal diferenciar entre "model" i "esquema":

- Model de dades: és l'eina intel·lectual (conjunt de conceptes i regles) que permet representar els aspectes estàtics i dinàmics de la part del món real que volem estudiar.
- Esquema: és la **representació** concreta de la part del món real utilitzant un model de dades.

## 2.2. ELEMENTS DEL MODEL RELACIONAL

Els distints tipus d'elements que veïrem en el model E-R tenen noms diferents en les diferents fases del cicle de vida:

Fase:	ANÀLISI	DISSENY	IMPLEMENTACIÓ
Model:	E-R	Relacional	Físic
Objectes:	Entitats i Relacions	Relacions	Taules
Característiques:	Atributs	Camps	Columnnes
Dades:	Ocurrències	Registres o tuples	Files

El **model relacional** (Edgard Codd 1970) representa una BD com un conjunt de relacions. Per exemple, la relació EMPLEATS podria ser esta:

<b>REGISTRES</b> (ocurrències, tuples o files)	<b>CAMPS</b> (atributs o columnnes)			
	<b>DNI</b>	<b>Nom</b>	<b>Cognom</b>	<b>Adreça</b>
	27456363Z	Pep	Garcia	c/ Sequial, 3
	32456783X	Marta	Pineda	c/ Nou, 17, 3a
	12345678A	Manel	Meseguer	c/ Niu, 33
	98765432Y	Joan	Roda	

Els camps són els noms de les propietats d'una relació.

Cada registre representa un conjunt de valors de dades relacionades entre sí.

És a dir, una **relació** està formada per **camps** i per **registres**, complint les següents condicions:

- 1) No hi ha tuples duplicades en una relació
- 2) No hi ha ordre entre les tuples

No és el mateix una taula i una relació. Les taules són la implementació de les relacions, on les files i columnnes tindran un ordre en concret. Ara bé, abusant del llenguatge, per a no confondre les "relacions" del disseny (les taules) amb les "relacions" de l'anàlisi (els rombes), utilitzarem indistintament els termes de "taula" i "relació".

El model de dades dependrà del SGBD amb el qual anem a treballar. Suposarem que el nostre SGBD sí que permetrà que una clau alternativa accepti valors nuls.

## CLAUS DE LES TAULES

En el model relacional, les tuples no poden estar duplicades. Per a això estan les claus.

Hi ha distints tipus de claus:

- Clau candidata (primàries / alternatives)
- Clau aliena

Clau: És un atribut (o més d'un) de la taula que complix 2 condicions:

- a) UNICITAT (identificació única). No poden haver dos tuples de R amb el mateix valor de cadascun dels atributs de la clau.
- b) MINIMALITAT (no redundància). No sobra cap atribut de la clau. És a dir: cap dels atributs de la clau pot ser eliminat sense que el conjunt resultant deixi de complir la propietat anterior. Exemple:

PROFESSOR = dni + nif + ... + dpt

La clau NO pot ser dni + nif perquè puc eliminar el nif de la clau, i la nova clau resultant (dni) segueix complint la primera propietat (identifica unívocament cada registre).

- Clau candidata: Cadascuna de les claus d'una taula. 2 tipus de claus candidates:

Clau primària: De totes les claus d'una taula, s'elegix una com a clau primària. Aquesta ha de complir la condició que no podran haver valors nuls en eixa clau.

Clau alternativa: Totes les altres claus. Sempre hi haurà una clau primària, però potser no hi haja una clau alternativa. Nosaltres suposarem que el SGBD on implementarem la nostra BD accepta valors nuls en les claus alternatives. Si no fóra així, el "pas a taules" podria ser diferent.

- Clau aliena (o externa): atribut (o conjunt d'atributs) d'una taula, el valor del qual ha de coincidir amb el valor de la clau primària d'una altra taula, a la qual referencia.

Les claus alienes servixen per a representar les relacions entre entitats de l'E-R.

PROFESSORS = dni + nom + ... + dpt

C. Ali : dpt → DEPARTAMENTS (dept)

Estem dient que *dpt* és una clau aliena i identifica el DEPARTAMENT: els valors de *dpt* que apareixen en PROFESSORS han d'estar també com a valors de la clau primària de DEPARTAMENTS (és a dir, en *dept*).

DEPARTAMENTS = dept + nom dept + ...

### RESTRICCIÓ DE VALOR NO NUL (VNN)

Si a una relació se li posa esta restricció en un atribut, cap valor d'este atribut podrà ser nul.

En el model relacional, per a expressar que un atribut té una restricció de valor no nul, ho indicarem amb VNN. Per exemple:

VNN: nom

### RESTRICCIONS D'INTEGRITAT IMPLÍCITES

Hi ha algunes restriccions que no cal indicar-les a l'esquema lògic perquè ja se sobreentenen. Estan implícites en el model. Són regles d'integritat general relacionades amb els tipus de claus:

- Integritat de claus (clau primària): Cap atribut component de la clau primària d'una relació pot tindre un valor nul.
- Integritat referencial (clau aliena): Si algun valor d'una clau aliena no és nul, haurà de ser igual a algun valor del camp de la taula a la qual fa referència.

Exemple de clau aliena:

ASSIGNATURA = cod + nom + ... + dpt

C.Ali : dpt → DEPARTAMENT (dept)

DEPARTAMENT = dept + nom\_dept + ...

La integritat referencial indica en este exemple que els valors de la clau aliena (*dpt*) o són nuls o coincideixen amb el valor de la clau primària de DEPARTAMENT (*dept*) en alguna tupla.

Per a indicar que un camp és clau primària, subratllarem eixe camp.

I per a indicar que un camp és una clau aliena, ho indicarem amb "C. Ali", com en l'exemple anterior.

## 2.3. ALTRES CONSIDERACIONS

En este apartat repassarem alguns conceptes i ho veurem gràficament.

L'esquema d'una taula  $T$ , denotat  $T = \underline{A_1} + A_2 + \dots + A_n$ , es compon del nom de la taula ( $T$ ) i de la llista dels atributs de la taula ( $A_1, \dots, A_n$ ). A més, tot esquema de taula ha de tindre una clau primària, que s'acostuma a subratllar.

Exemple d'esquema de taula:

$EMPLEATS = \underline{DNI} + \text{Nom} + \text{Cognoms} + \text{Adreça}$

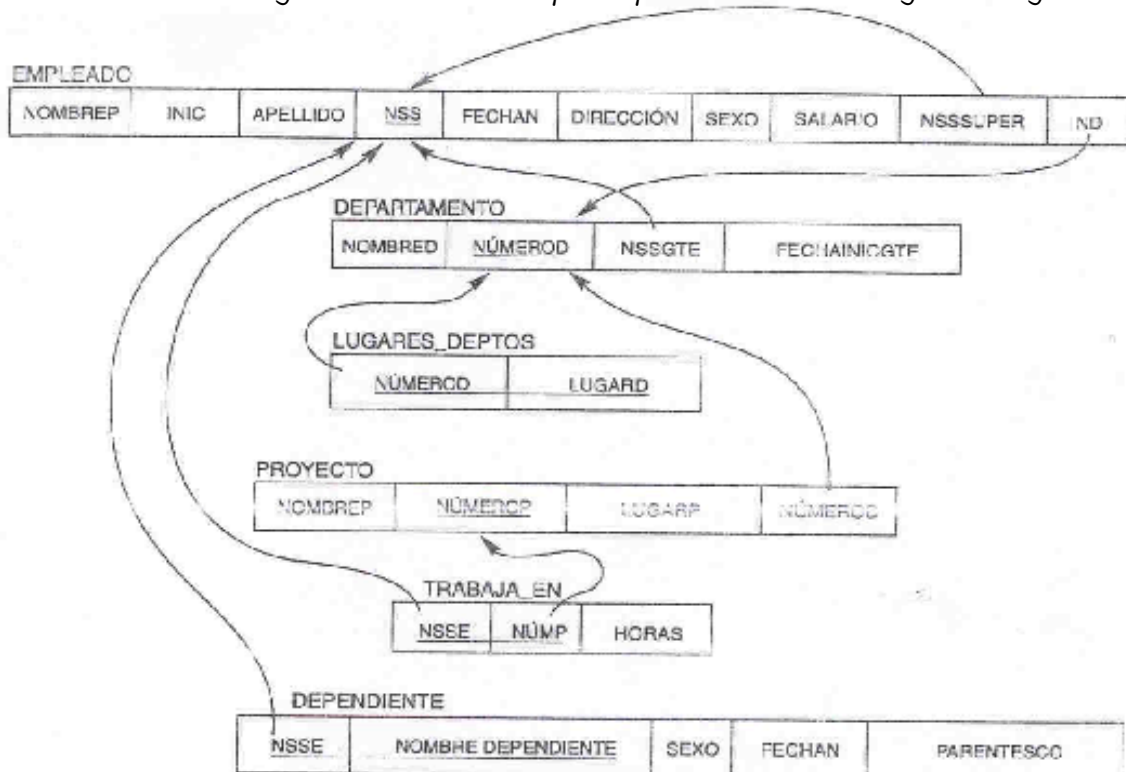
Exemples d'ocurrències (registres) de la taula EMPLEATS:

20934594S	Pep	Garcia Martínez	c/ Sequial, 5, 2a
24562354A	Maria	Albors Marqués	c/ Nou 19
78378243N	Joan	Compta	

Una BD relacional sol contindre moltes taules. Per exemple, la BD Institut podria estar formada per les taules: PROFESSORS, ASSIGNATURES, ALUMNES i GRUPS.

Altra cosa: una clau aliena pot fer referència a la seua pròpia taula.

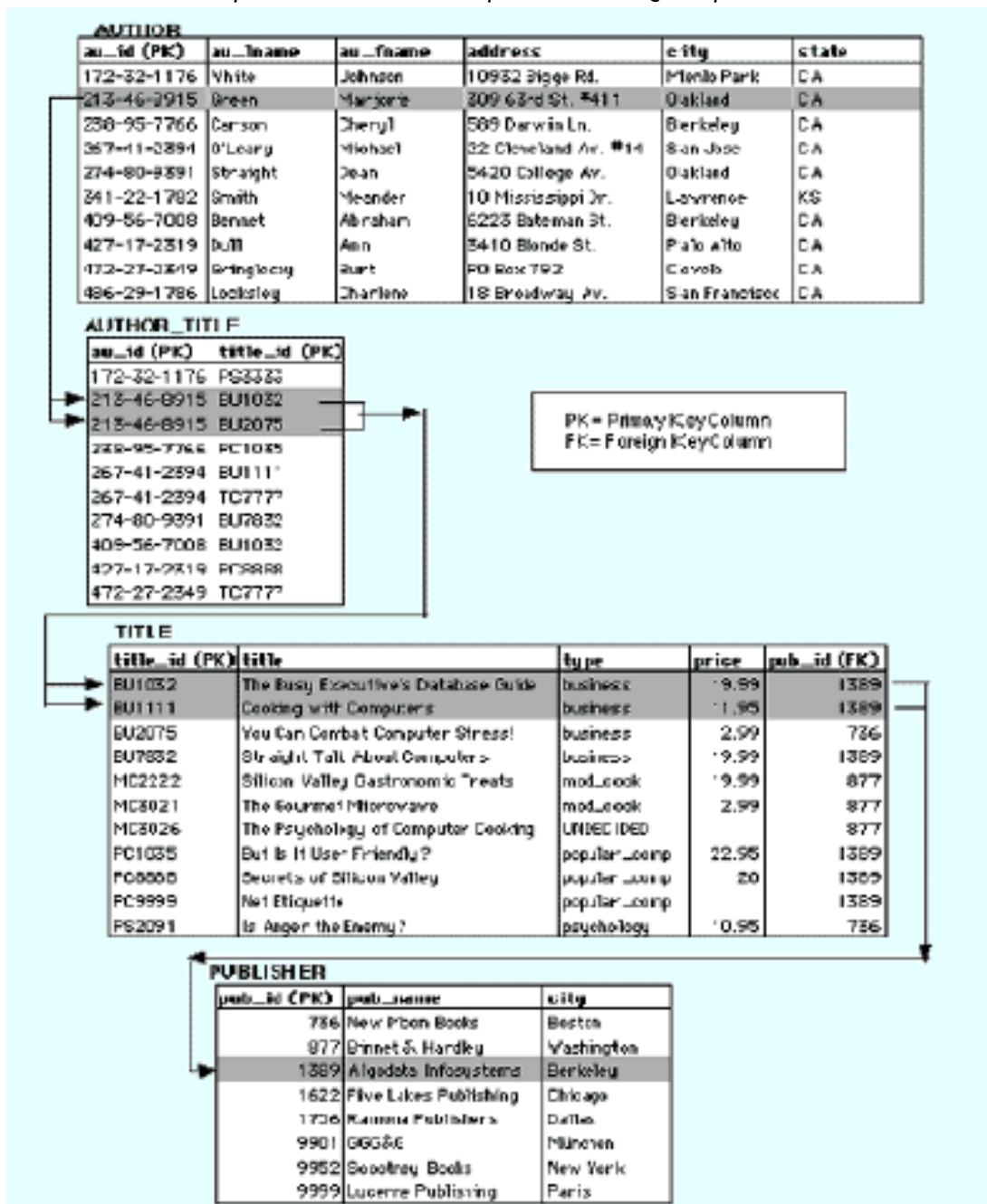
La restricció d'integritat referencial es pot representar amb el següent diagrama:



Algunes observacions que cal fer (ja ho vorem àmpliament) són:

- Un mateix concepte del món real, al representar-se com un atribut, pot tindre noms diferents a les taules on apareix (“dpt”, “dept”, “depart”, ...)
- Les relacions del E-R (és a dir, els “rombes” i “triangles”) es representaran en l'esquema relacional amb les claus primàries i alienes de les taules.
- En un esquema relacional caldrà crear:
  - Una taula per cada entitat
  - Una taula per cada atribut multivalent.
  - Una taula per cada relació ternària
  - Una taula per cada relació binària M:M

Tenint en compte les claus alienes, podem “navegar” per les taules:



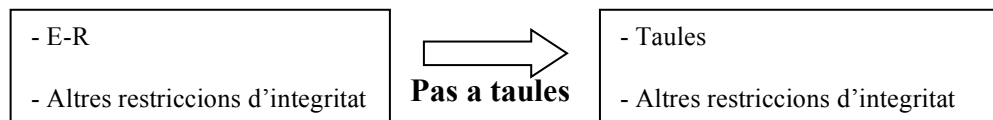


### 3. Model Relacional: transformació del E-R a taules

Hi ha dues tècniques per a dissenyar BD relacionals, depenent de si hem fet prèviament el E-R o no. És a dir, per a obtenir el conjunt de taules d'un esquema relacional, ho podem fer de 2 formes:

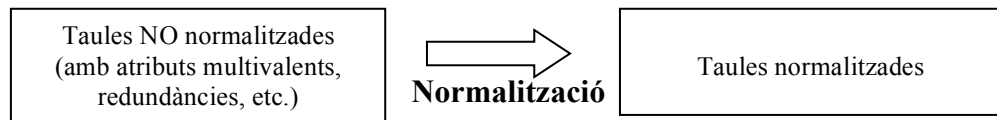
#### ✓ Transformació del E-R a taules

Esta tècnica s'utilitza quan partim d'un esquema E-R. Aplicant-li unes regles obtindrem el conjunt de taules corresponent. També s'ha d'intentar passar a taules les altres restriccions que no es podien captar en l'E-R. Les que no es puguin passar a taules, quedaran encara com a "altres restriccions d'integritat".



#### ✓ Normalització

Esta tècnica s'utilitza quan partim d'un esquema de taules "fet de qualsevol manera". Aplicant unes regles a eixes taules, obtindrem un bon esquema de taules. És a dir: obtindrem un conjunt de taules normalitzades.



En ambdues tècniques, es pretén aconseguir, entre altres, els següents objectius:

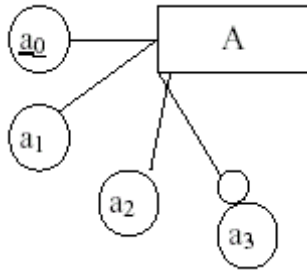
- ✓ Conservar les restriccions de clau, connectivitat i cardinalitat.
- ✓ Evitar duplicacions de dades.
- ✓ Evitar valors nuls on no toque posar-los.

D'eixes dos tècniques del model relacional en este punt vorem la transformació del E-R a taules: vorem com obtenir el conjunt de relacions (taules) equivalents a cadascuna de les entitats i relacions del model Entitat/Relació.

Comencem ja explicant com passar a taules cada element del model E-R.

### 3.1. ENTITATS I ATRIBUTS

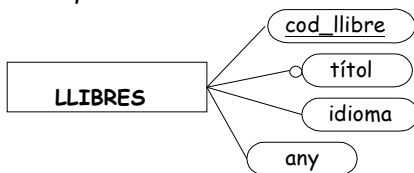
#### 3.1.1. Entitats amb atributs simples



$$A = \underline{a0} + a1 + a2 + a3$$

VNN: a3

*Exemple:*

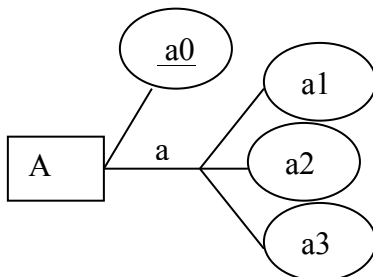


$$\text{LLIBRES} = \underline{\text{cod\_llibre}} + \text{títol} + \text{idioma} + \text{any}$$

VNN : títol

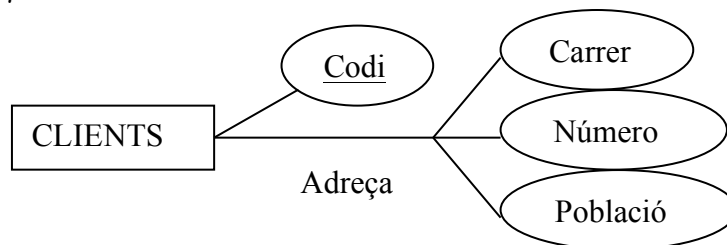
Més endavant vorem com passar a taules les entitats dèbils. És a dir, aquelles que tenen una restricció d'identificació (ID).

#### 3.1.2. Entitats amb atributs compostos



$$A = \underline{a0} + a1 + a2 + a3$$

*Exemple:*



$$\text{CLIENTS} = \underline{\text{Codi}} + \text{Carrer} + \text{Número} + \text{Població}$$

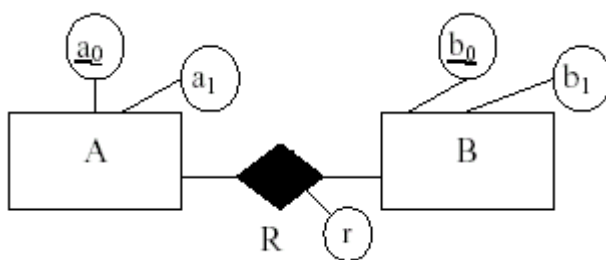
Més endavant vorem els atributs multivalents (després de vore les relacions 1:M).

## 3.2. RELACIONS BINÀRIES

### 3.2.1. Relació binària M:M

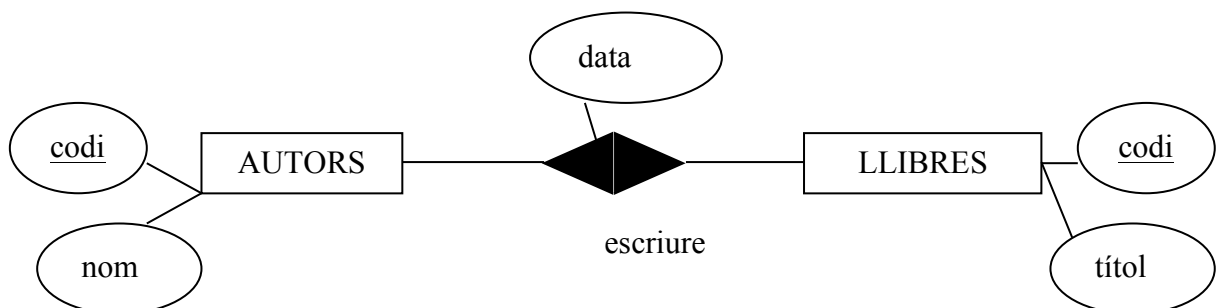
Quan una relació és M:M (tant unàries com binàries), cal crear una taula per a la relació. Els atributs de la nova taula seran:

- Les claus de les entitats de la relació (que tots junts formaran la clau i cadascun d'ells serà una clau aliena)
- Els propis atributs de la relació.

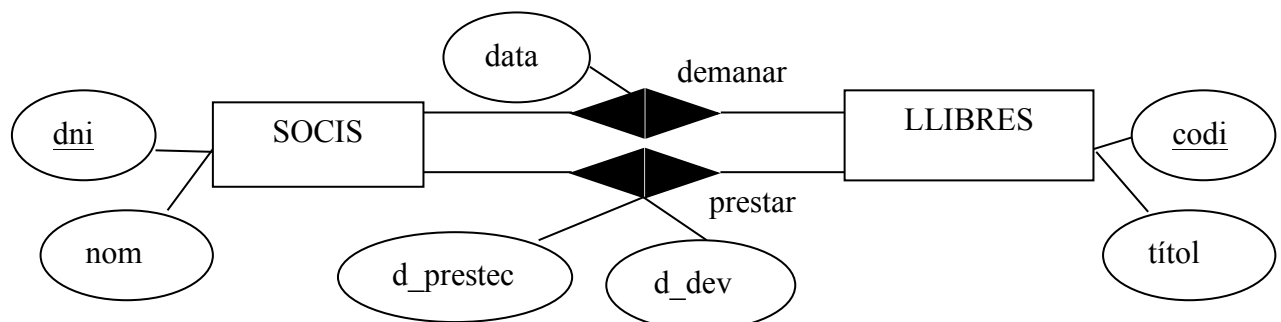


$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1$   
 $R = \underline{a0} + \underline{b0} + r$   
**C.Ali:**  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$

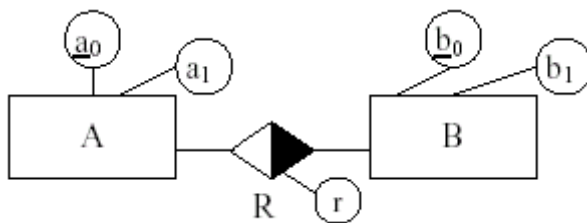
Exercici 1: Passa a taules el següent E-R.



Exercici 2: Passa a taules el següent E-R.



### 3.2.2. Relació binària 1:M

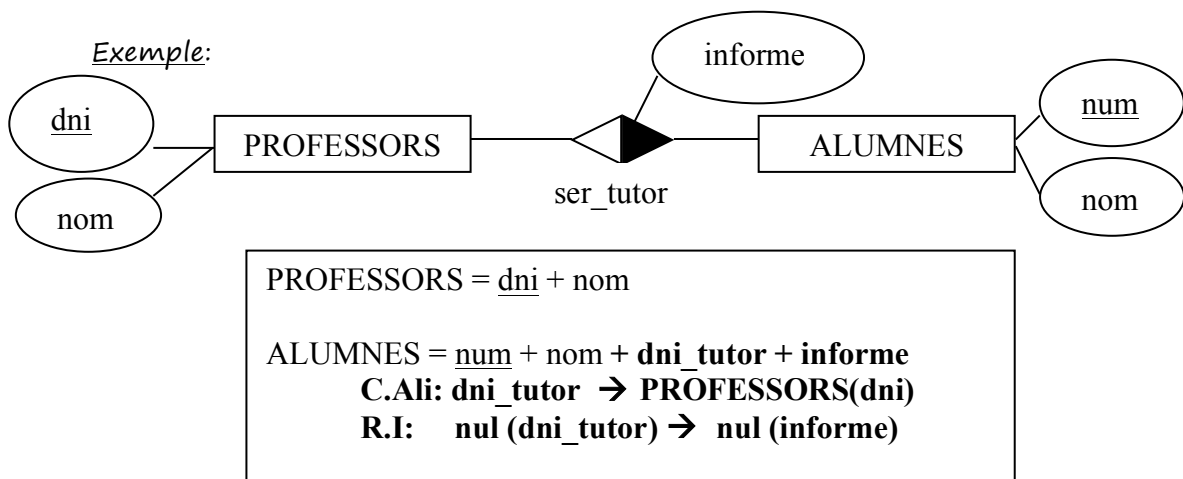


$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1 + a0 + r$$

**C.Ali:**  $a0 \rightarrow A(a0)$

**R.I:**  $\text{nul}(a0) \rightarrow \text{nul}(r)$



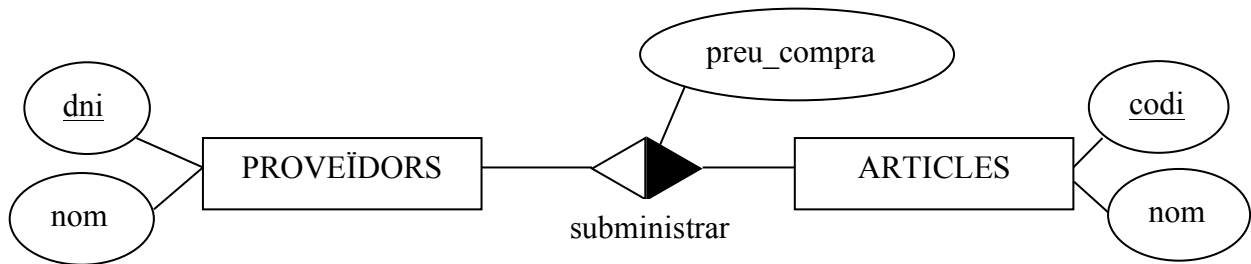
Eixa R.I. indica que si algun alumne no té un tutor assignat, no té sentit que eixe alumne tinga un informe.

És a dir: si en algun registre de l'entitat alumnes hi ha un valor de "dni\_pro" que està buit, també haurà d'estar buit el valor d' "informe" d'eixe registre.

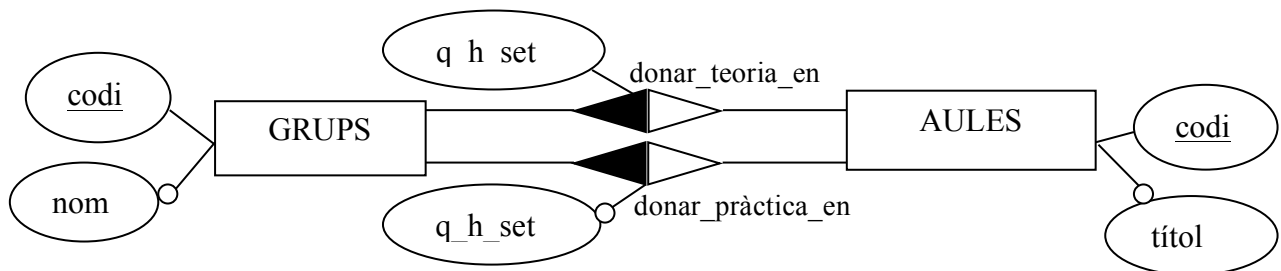
ALUMNES			
<u>num</u>	nom	dni_tutor	informe
1	Pep	33333333	Es porta mal
2	Pepa	33333333	No fa el deure
3	Pepet	55555555	Molesta en classe
4	Pepin		<b>Ha furtat un pen</b>
5	Pepeta		
6	Pepiqueta	55555555	

Amb la R.I. diem que no té sentit que Pepin tinga alguna cosa en el camp "informe" ja que no té cap tutor assignat (i l'informe l'elabora el tutor). Ara bé, potser que Peiqueta tinga un tutor assignat però que encara no ha fet cap informe d'ella.

Exercici 3: Passa a taules el següent E-R.

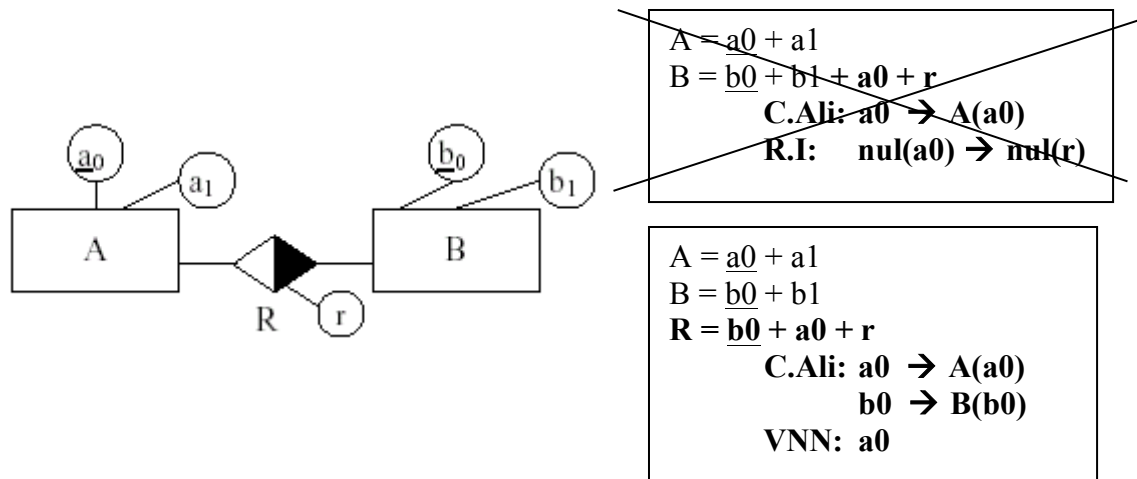


Exercici 4: Passa a taules el següent E-R.

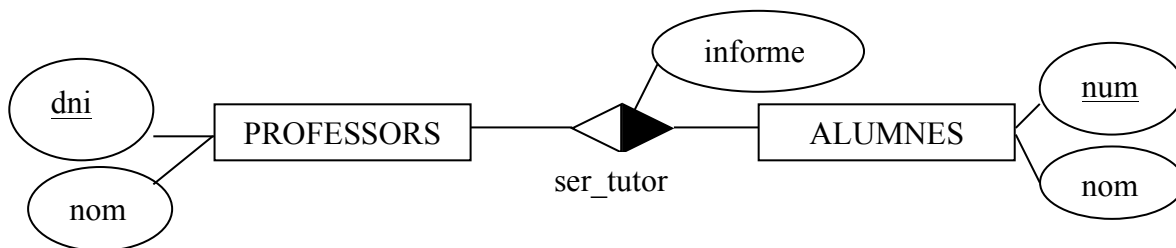


## Relació binària 1:M. Una altra solució

Si no volem tindre la R.I (de l'atribut en la relació 1:M), en compte d'expressar la relació mitjançant una de les entitats, podríem posar-la com a una taula apart, on estigueren només les parelles d'ocurrències de les dos entitats que estiguen relacionades:



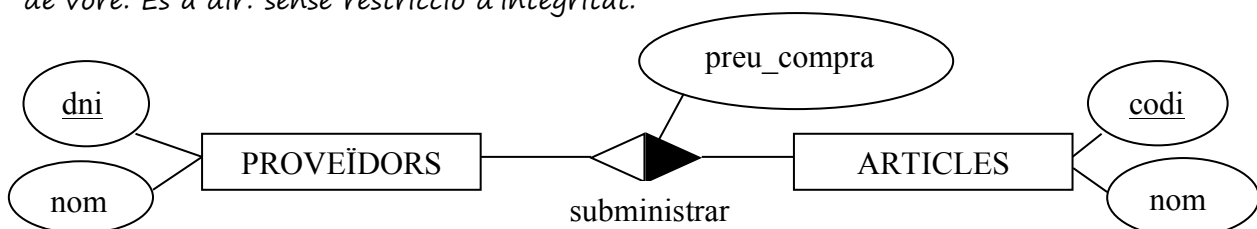
Exemple:



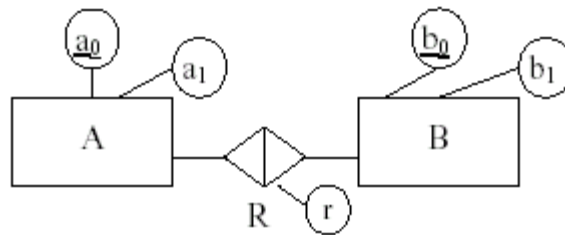
PROFESSORS = dni + nom  
ALUMNES = num + nom  
TUTORIES = num\_alumne + dni\_tutor + informe  
C.Ali:  $\text{num\_alumne} \rightarrow \text{ALUMNES}(\text{num})$   
 $\text{dni\_tutor} \rightarrow \text{PROFESSORS}(\text{dni})$   
VNN:  $\text{dni\_tutor}$

En la taula ALUMNES estaran tots els alumnes, però en la de TUTORIES només posarem aquells alumnes que tenen tutor. Per això li hem posat el VNN a "dni\_tutor". Així no podrà donar-se el cas que un alumne tinga informe sense tindre tutor.

Exercici 5: Fes el pas a taules d'esta relació 1:M seguint esta forma que acabem de veure. És a dir: sense restricció d'integritat.



### 3.2.3. Relació binària 1:1



$A = \underline{a0} + a1 + b0 + r$   
 C.Ali:  $b0 \rightarrow B(b0)$   
**C.Alt:  $b0$**   
 R.I:  $\text{nul}(b0) \rightarrow \text{nul}(r)$   
 $B = \underline{b0} + b1$

O bé:

$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1 + a0 + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
**C.Alt:  $a0$**   
 R.I:  $\text{nul}(a0) \rightarrow \text{nul}(r)$

Les relacions 1:1 són iguals que les de 1:M però posant la corresponent **clau alternativa**.

Recordem que, com havíem dit que **les claus alternatives poden tindre valors nuls**, cal indicar també que si és nul, que també ho siga l'atribut de la relació ( R.I:  $\text{nul}(a0) \rightarrow \text{nul}(r)$  ). És a dir, una 1:1 és igual que una 1:M però amb la clau alternativa.

**IMPORTANT:** Si l'entitat A tinguera una restricció d'existència sobre la relació R, usaríem la primera solució que hem vist ja que desapareixeria la R.I (ja ho vorem quan fem el pas a taules de les restriccions d'existència).

### Relació binària 1:1. Una altra solució

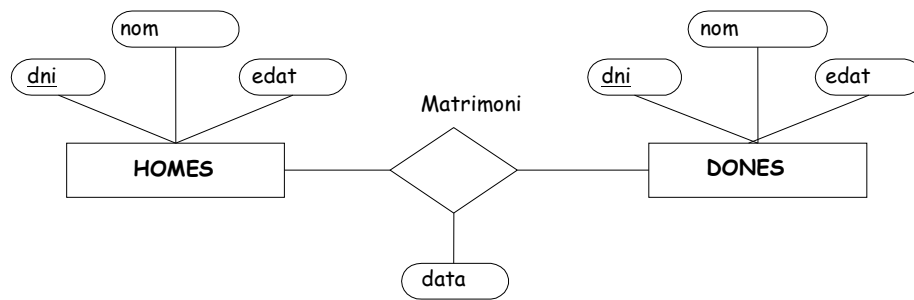
Si no volem tindre eixa R.I. (de l'atribut en la relació 1:1), en compte d'expressar R per mig de B, podríem posar-la com a una taula apart, on estigueren **només** les parelles de A i B relacionades:

$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1$   
 $R = \underline{b0} + a0 + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
            $b0 \rightarrow B(b0)$   
**C.Alt:  $a0$**   
**VNN:  $a0$**

o bé:

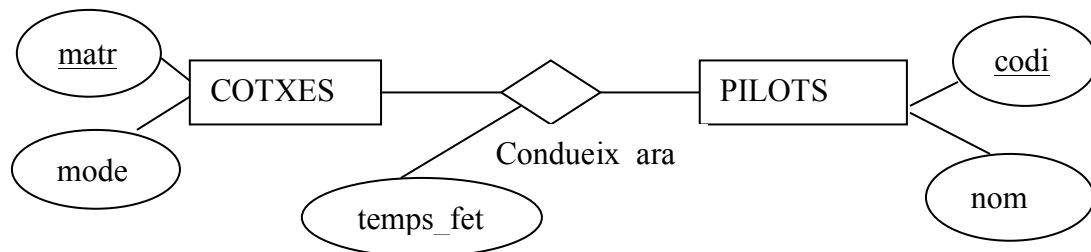
$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1$   
 $R = \underline{a0} + b0 + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
            $b0 \rightarrow B(b0)$   
**C.Alt:  $b0$**   
**VNN:  $b0$**

*Exemple, utilitzant altra taula per a la relació:*



HOMES = dni + nom + edat  
 DONES = dni + nom + edat  
 MATRIMONI = dni\_h + dni\_d + data  
 CAI: dni\_h → HOMES(dni)  
      dni\_d → DONES(dni)  
 CAI: dni\_d  
 VNN: dni\_d

*Exercici 6: Passa a taules este E-R, amb les 4 formes vistes.*



	<i>Cada cotxe tindrà el seu pilot</i>	<i>Cada pilot tindrà el seu cotxe</i>
<i>Sense altra taula per a la relació</i>		
<i>Amb altra taula per a la relació</i>		



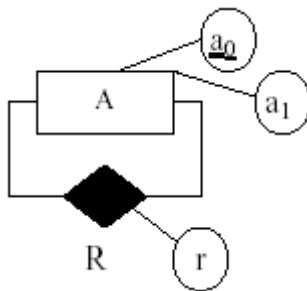
### 3.3. RELACIONS UNÀRIES

Cal imaginar que una unària és com una binària però on les dos entitats són realment la mateixa. En una mateixa taula tindrem 2 voltes la clau de l'entitat. Per tant, caldrà triar un nom diferent.

El nom dels camps en les relacions unàries:

- Si les 2 ocurrences que es relacionen tenen el mateix significat en la relació, els noms poden ser qualsevol. Per exemple, en la relació "casar-se" els codis de les persones podrien dir-se "cònjuge1" i "cònjuge2", posem per cas.
- Però si tenen una participació diferent en la relació, els noms hauran de ser significatius. Per exemple, en la relació "estimar" els atributs haurien de dir-se "estimador" i "estimat", posem per cas.

#### 3.3.1. Relació unària M:M

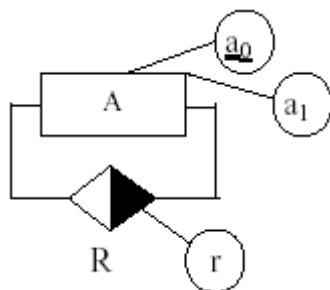


$$A = \underline{a0} + a1$$

$$R = \underline{a0} + \underline{a0'} + r$$

C.Ali:  $a0 \rightarrow A(a0)$   
 $a0' \rightarrow A(a0)$

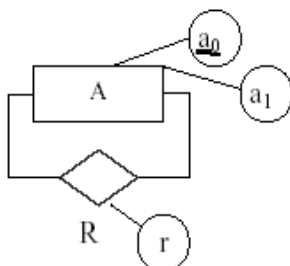
#### 3.3.2. Relació unària 1:M



$$A = \underline{a0} + a1 + a0' + r$$

C.Ali:  $a0' \rightarrow A(a0)$   
 R.I:  $\text{nul}(a0') \rightarrow \text{nul}(r)$

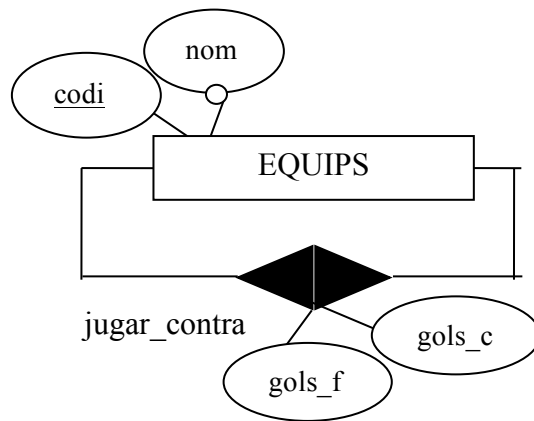
#### 3.3.3. Relació unària 1:1



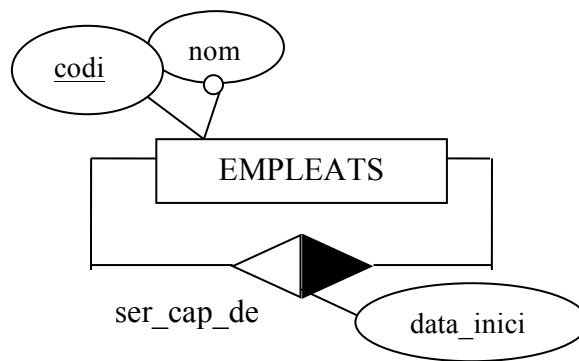
$$A = \underline{a0} + a1 + a0' + r$$

C.Ali:  $a0' \rightarrow A(a0)$   
 C.Alt:  $a0'$   
 R.I:  $\text{nul}(a0') \rightarrow \text{nul}(r)$

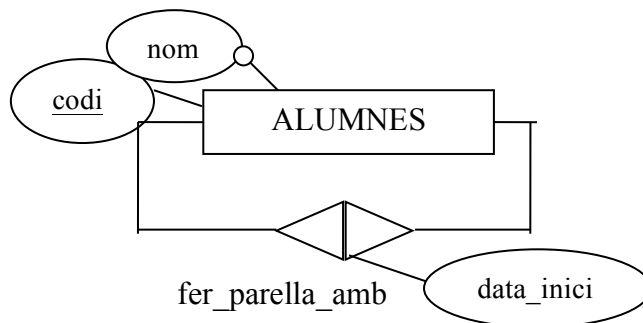
Exercici 7: Passa a taules este E-R.



Exercici 8: Passa a taules este E-R.



Exercici 9: Passa a taules este E-R.



### 3.4. RELACIONS TERNÀRIES

#### Taula nova

Totes les ternàries generen taula per a la relació R.

#### Atributs

Els atributs de la nova taula de R (igual que passava en les relacions M:M) seran:

- les claus de les corresponents entitats
- els atributs propis de la relació

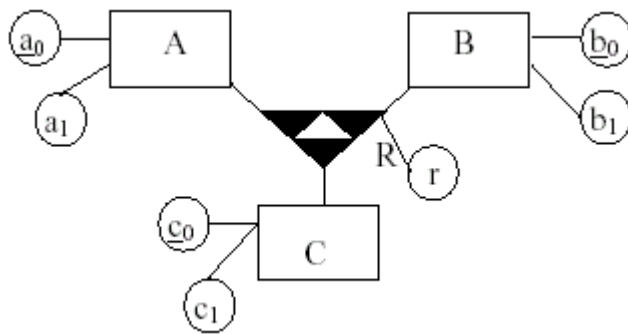
#### Clau

Si és M:M:M, la clau estarà formada per les claus de les respectives taules.

Si no, la clau sempre estarà formada per les claus de 2 de les 3 taules. N'hi haurà tantes claus com connectivitats a "1", sent la clau respectiva, la parella dels altres dos. Una de les claus serà la primària; si n'hi ha més, les altres seran les alternatives. Ho veurem més clar amb els esquemes del punt següent.

Veiem cada cas particular:

### 3.4.1. Relació ternària M:M:M (zero triangles blancs)



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$C = \underline{c0} + c1$$

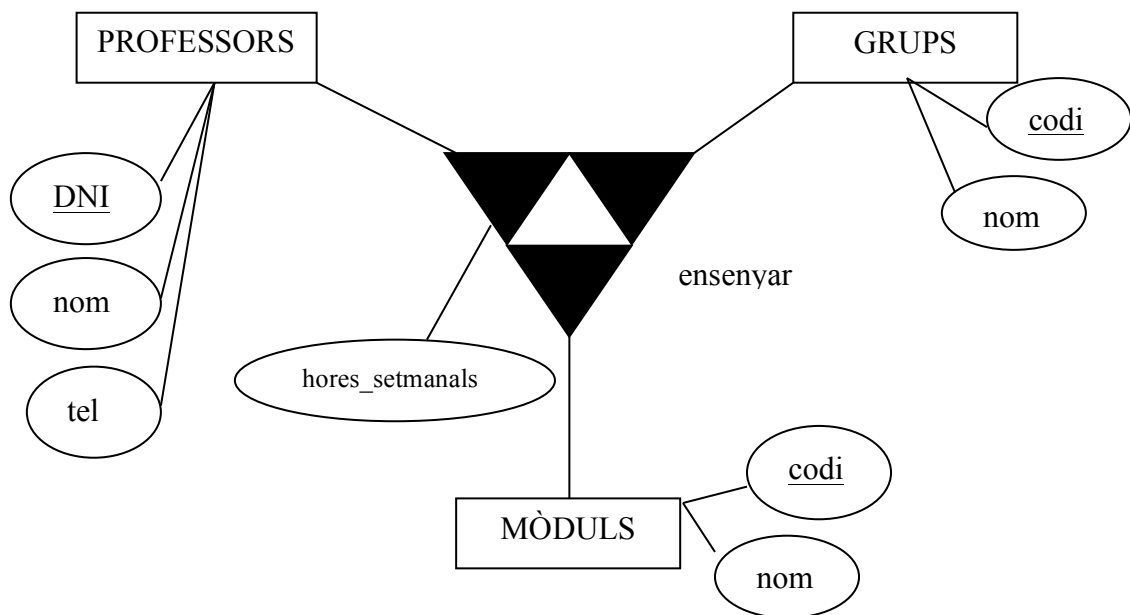
$$R = \underline{a0} + \underline{b0} + \underline{c0} + r$$

$$\text{C.Ali: } a0 \rightarrow A(a0)$$

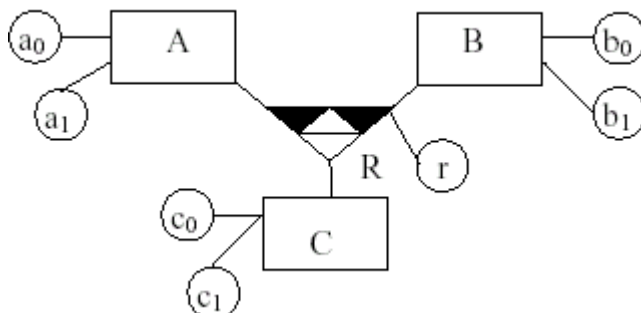
$$b0 \rightarrow B(b0)$$

$$c0 \rightarrow C(c0)$$

*Exercici 10: Passa a taules el següent E-R*



### 3.4.2. Relació ternària 1:M:M (1 triangle blanc)



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$C = \underline{c0} + c1$$

$$R = \underline{a0} + \underline{b0} + c0 + r$$

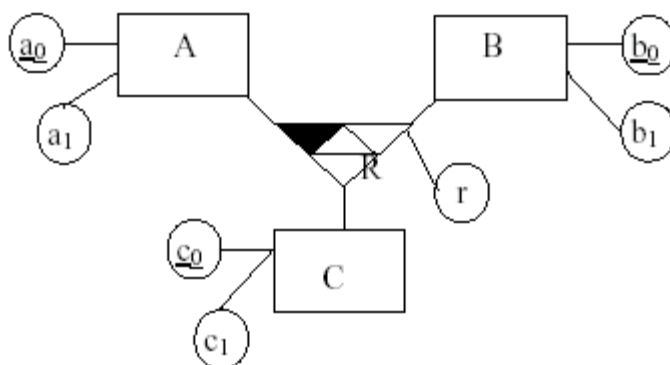
$$\text{C.Ali: } a0 \rightarrow A(a0)$$

$$b0 \rightarrow B(b0)$$

$$c0 \rightarrow C(c0)$$

$$\text{VNN: } c0$$

### 3.4.3. Relació ternària 1:1:M (2 triangles blancs)



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$C = \underline{c0} + c1$$

$$R = \underline{a0} + \underline{b0} + c0 + r$$

$$C.Ali: a0 \rightarrow A(a0)$$

$$b0 \rightarrow B(b0)$$

$$c0 \rightarrow C(c0)$$

$$VNN: c0$$

$$C.Alt : a0 + c0$$

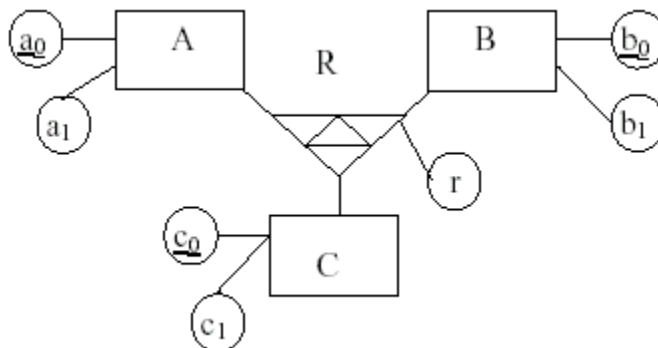
Altra forma seria:

$$R = \underline{a0} + \underline{c0} + b0 + r$$

$$VNN: b0$$

$$C.Alt : a0 + b0$$

### 3.4.4. Relació ternària 1:1:1 (3 triangles blancs)



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$C = \underline{c0} + c1$$

$$R = \underline{a0} + \underline{b0} + c0 + r$$

$$C.Ali: a0 \rightarrow A(a0)$$

$$b0 \rightarrow B(b0)$$

$$c0 \rightarrow C(c0)$$

$$VNN: c0$$

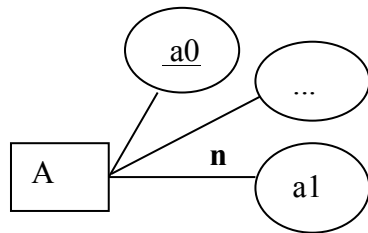
$$C.Alt : a0 + c0$$

$$b0 + c0$$

Igual que en el cas anterior, també tenim solucions alternatives: que la clau primària de R siga a0 + c0 o bé b0 + c0 (amb les respectives C.Alt i VNN, clar).

Exercici 11: Com quedaria el pas a taules de l'exercici anterior si les connectivitats de GRUPS i de PROFESSORS respecte a la relació "pertànyer" estigueren a 1 i no a M.

### 3.5. ATRIBUTS MULTIVALENTS



$$A = \underline{a0} + \dots + \{a1\}$$

A vegades es representa així però no està normalitzat i, per tant, no ho deixarem així, sinó que caldrà crear una taula nova.



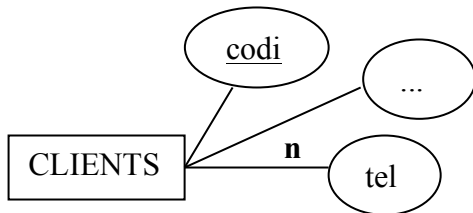
$$A = \underline{a0} + \dots$$

$$B = \underline{a0} + a1$$

$$C. Ali: a0 \rightarrow A(a0)$$

Per a fer el pas a taules d'atributs multivalents crearem una taula nova.

Per exemple:



$$CLIENTS = \underline{codi} + \dots + \{tel\}$$

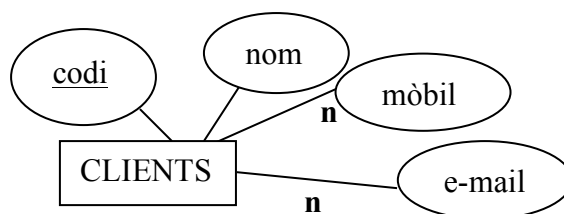

$$CLIENTS = \underline{codi} + \dots$$

$$CLI\_TEL = \underline{codi} + tel$$

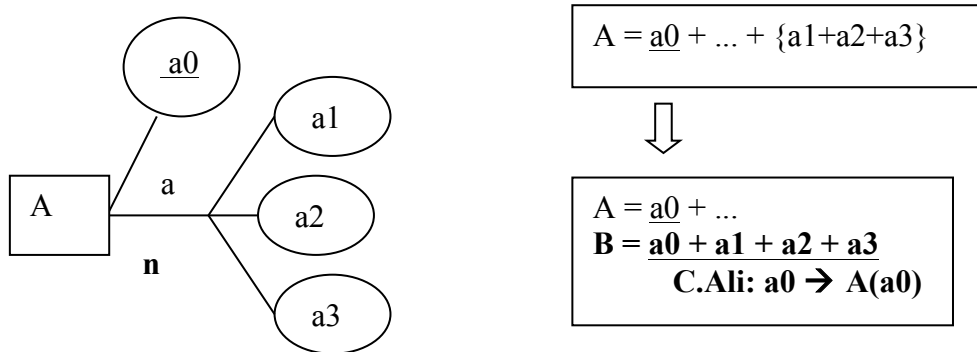
$$C. Ali: codi \rightarrow CLIENTS(codi)$$

Això suposant que la relació entre clients i telèfons és de M:M. Si fóra 1:M (és a dir, un telèfon només és d'un client), la clau de CLI\_TEL seria només el codi del telèfon.

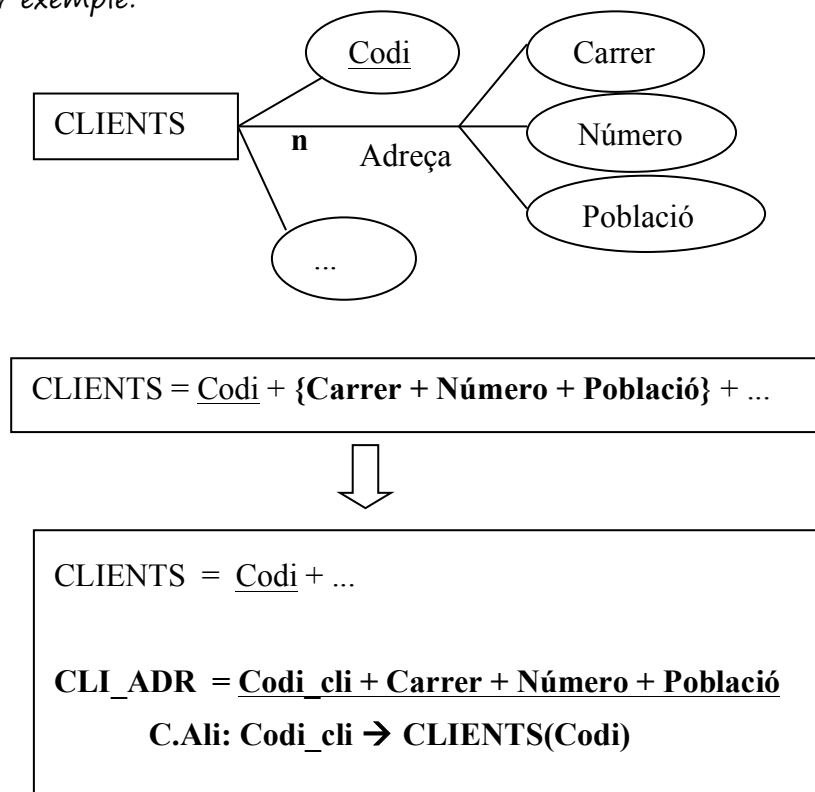
Exercici 12: Passa a taules este E-R tenint en compte que un mateix telèfon port pertànyer a més d'un client però un e-mail només és d'un i prou.



### 3.6. ATRIBUTS MULTIVALENTS COMPOSTOS

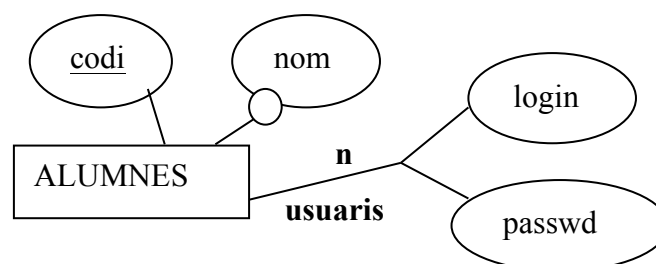


Per exemple:

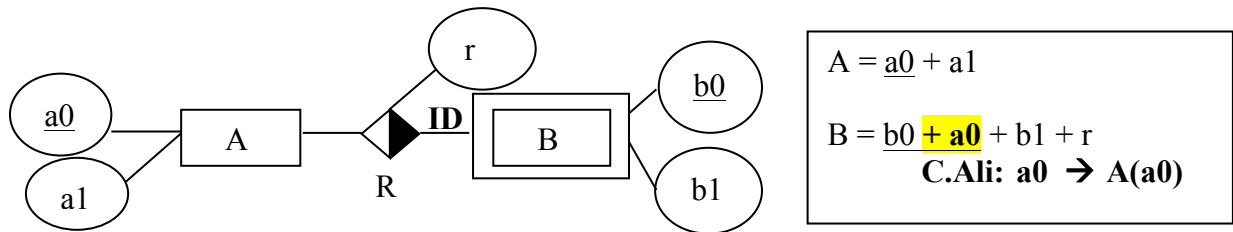


Això suposant que la relació entre clients i adreces és de M:M. Si fóra 1:M (és a dir, una adreça només correspon a un client), la clau de CLI\_ADR seria només el codi de l'adreça (carrer + número + població).

Exercici 13: Passa a taules este E-R.



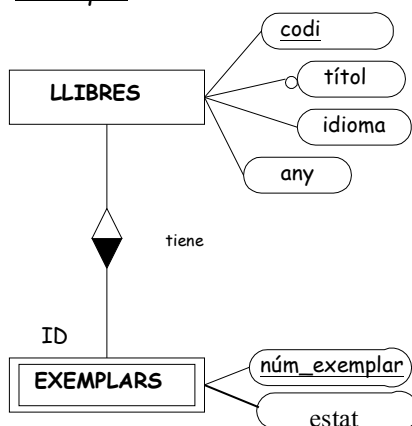
### 3.7 RESTRICCIONS D'IDENTIFICACIÓ (ENTITATS DÈBILS)



Les restriccions tipus ID sempre es donen en relacions 1:M, amb la iD a la part del “molts”). Per tant, el pas a taules serà fer estos 2 passos:

- 1) Fer el pas a taules d'una relació 1:M
- 2) Subratllar també la clau aliena. I com esta ja no podrà tindre valors nuls, ja no té sentit posar la R.I.  $nul(a0) \rightarrow nul(r)$

Exemple:



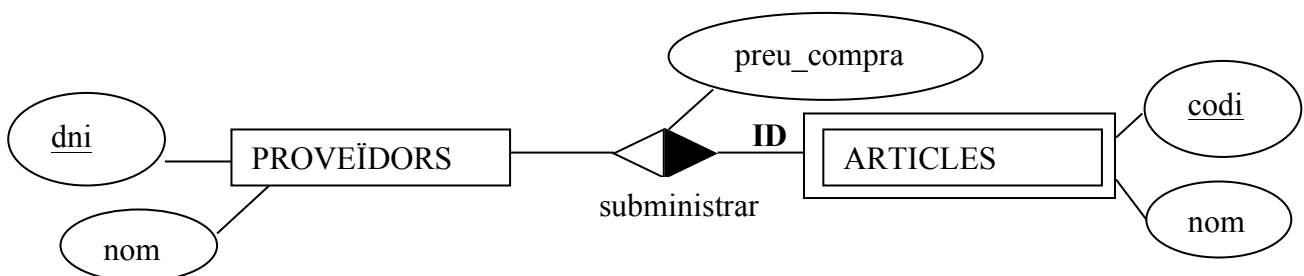
LLIBRES = codi + títol + idioma + any

VNN : títol

EXEMPLARS = núm\_exemplar + cod\_llibre + estat

C.Ali : cod\_llibre → LLIBRES (codi)

Exercici 14: Fes el pas a taules del següent E-R.



Exercici 15: Pensa com seria el pas a taules de la ID anterior si haguérem resolt la relació “subministrar” de la segona forma que hem vist en les relacions 1:M (és a dir: sense la R.I.  $nul(...) \rightarrow nul(...)$ ).



### 3.8. RESTRICCIONS D'EXISTÈNCIA

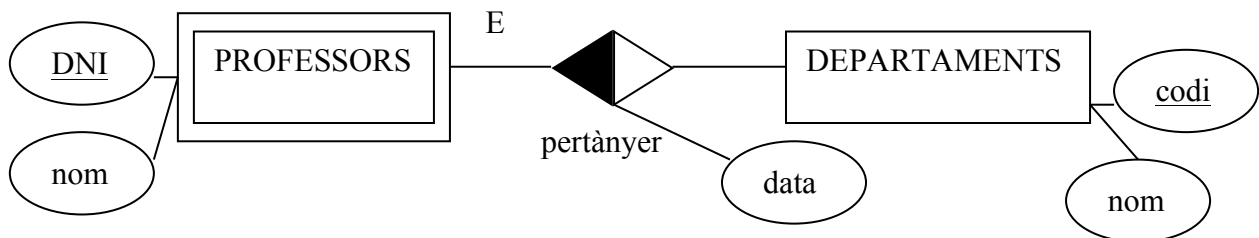
Hi ha dues formes de tractar les RE. N'aplicarem una o altra, depenent de si les parelles d'ocurrències relacionades estan en la mateixa taula que l'entitat que té la restricció o no:

a) Si estan en la mateixa taula, ho indicarem amb VNN

b) Si no, ho indicarem amb una R.I. semblant a una clau aliena ( $\subset$ )

#### 3.8.a) La R.E. està expressada en la mateixa taula: **VNN**

Veiem-ho amb un exemple.



Anem a deduir detalladament, com faríem el pas a taules.

Primer fem el pas a taules sense tindre en compte la E:

PROFESSORS = dni + nom + dep + data

C. Ali: dep  $\rightarrow$  DEPARTAMENTS(codi)

R. I: nul(dep)  $\rightarrow$  nul(data)

DEPARTAMENTS = codi + nom

Ara anem a per la E. Si pensem en possibles ocurrències, quin significat té eixa E?

PROFESSORS = dni + nom + dep + data

1    Pep    INF    4-3-12

2    Pepa    INF

3    Pepet    MAT

4    Pepi    —

No volem que passe això: que hi hagen professors sense departament (com és el cas del 4, Pepi).

Per tant, obligarem a que en el camp dep no puguem haver valors nuls:

VNN: dep

És a dir, eixa E significa que:

“Tots els professors han de pertànyer obligatòriament a un departament”.

- En quina taula estan “tots els professors”?
  - o En la de PROFESSORS
- En quina taula estan les parelles “professor-departament”?
  - o En la de PROFESSORS també

Veiem que per a expressar la E ho tenim tot en la mateixa taula. En eixe cas expressarem la E com un VNN:

PROFESSORS = dni + nom + dep + data  
C. Ali: dep → DEPARTAMENTS(codi)  
R. I: nul(dep) → nul(data)  
VNN: dep

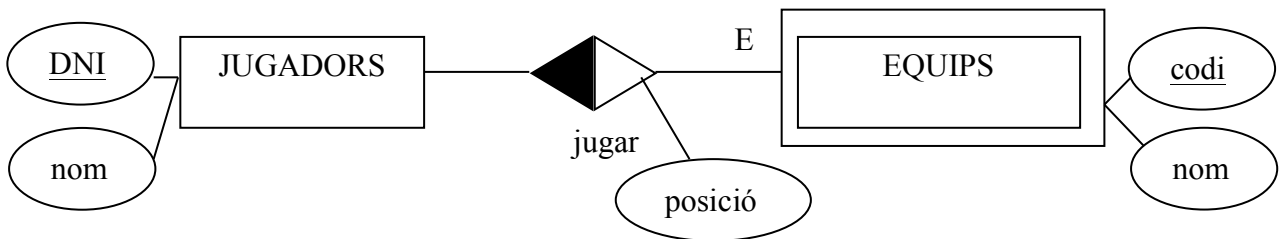
Però com ara no anem a deixar que el dep siga nul, no té sentit eixa R. I. i, per tant, la llevarem.

En resum, el pas a taules de l'exemple quedaria així:

PROFESSORS = dni + nom + dep + data  
C. Ali: dep → DEPARTAMENTS(codi)  
**VNN: dep**  
  
DEPARTAMENTS = codi + nom

### 3.8.b) La R.E. està expressada en taules diferents: R.I. tipus $\subset$

Veiem-ho amb un exemple:



Anem a deduir detalladament, com faríem el pas a taules.

Primer fem el pas a taules sense tindre en compte la E:

JUGADORS = dni + nom + equip + posició

C. Ali: equip → EQUIPS(codi)

R. I: nul(equip) → nul(posició)

EQUIPS = codi + nom

Ara anem a per la E. Si pensem en possibles ocurrències, quin significat té eixa E?

JUGADORS = dni + nom + equip + posició

1 Pep VCF porter

2 Pepa VCF

3 Pepet BAR defensa

4 Pepi

EQUIPS = codi + nom

VCF València C.F.

BAR F.C. Barcelona

**LLE** Llevant

No volem que passe això: que hi hagen equips que no tenen jugadors (com és el cas del LLE).

Per tant, obligarem a que totes les ocurrències del camp *codi* de la taula d'*EQUIPS* estiguen en alguna ocurrència del camp *equip* de la taula de *JUGADORS*.

R.I: EQUIPS(codi)  $\subset$  JUGADORS(equip)

És a dir, eixa E significa que:

“Tots els equips han de tindre obligatòriament algun jugador que jugue en ell”.

- En quina taula estan “tots els equips”?
  - o En la d'EQUIPS
- En quina taula estan les parelles “equip-jugador”?
  - o En l'altra taula: la de JUGADORS

Veiem que per a expressar la E ho tenim tot en taules diferents. En eixe cas expressarem la E com una RI:

JUGADORS = dni + nom + equip + posició

C. Ali: equip → EQUIPS(codi)

R. I: nul(equip) → nul(posició)

EQUIPS = codi + nom

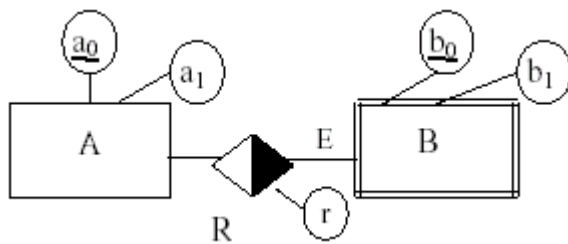
R. I: codi ⊂ JUGADORS(equip)

És a dir: igual que diem en la clau aliena que les ocurrències de l'atribut equip de JUGADORS han d'estar com a ocurrències en el camp codi d'EQUIPS, amb eixa R. I. estem dient que al revés també s'ha de complir: que totes les ocurrències del camp codi de la taula d'EQUIPS també han d'aparèixer com a ocurrències del camp equip de la taula JUGADORS.

Nota: eixa R.I. l'hem posada com a una restricció de la taula d'equips però a vegades se sol representar com a una restricció que ha de complir la base de dades (ja que afecta a més d'una taula). En eixe cas es posaria “fora” de les 2 taules, així: R. I: EQUIPS(codi) ⊂ JUGADORS(equip)

Amb el que hem vist ja sabem com passar a taules les restriccions d'existència per a qualsevol tipus de relació. No obstant, ara vorem com seria per a les distintes possibilitats de graus i connectivitats.

### Relació binària 1:M amb R.E a la part del "Molts"



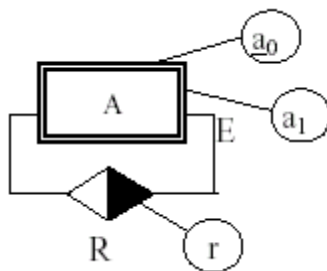
$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1 + a0 + r$$

$$\text{C.Ali: } a0 \rightarrow A(a0)$$

$$\text{VNN: } a0$$

### Relació unària 1:M amb R.E. a la part de "Molts".

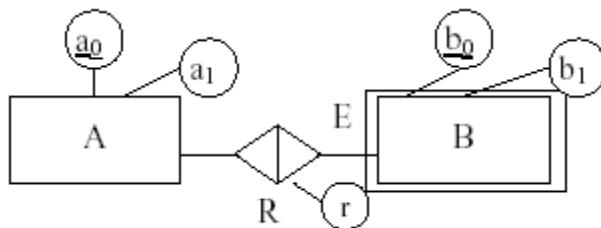


$$A = \underline{a0} + a1 + a0' + r$$

$$\text{C.Ali: } a0' \rightarrow A(a0)$$

$$\text{VNN: } a0'$$

### Relació binària 1:1 amb una R.E.



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1 + a0 + r$$

$$\text{C.Ali: } a0 \rightarrow A(a0)$$

$$\text{C.Alt: } a0$$

$$\text{VNN: } a0$$

#### Notes:

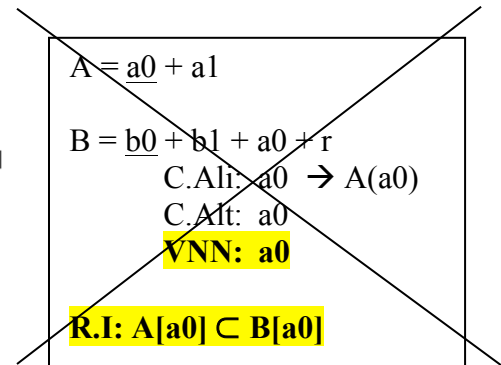
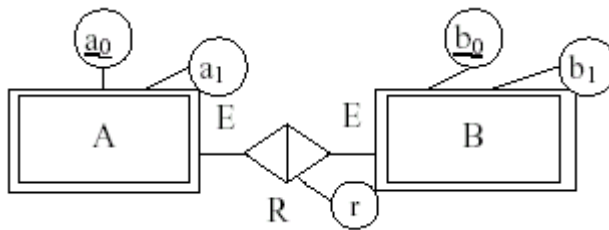
✓ Ara no cal posar R.I:  $\text{nul}(a0) \rightarrow \text{nul}(r)$  perquè no estem deixant que  $a0$  siga nul.

✓ Si en compte de posar  $a0$  com a clau aliena en B, ho haguérem fet al revés (posar  $b0$  com a clau aliena en A), a més de la R.I. de l'atribut de la relació, tindríem la de la restricció d'existència. Per tant, la primera solució és la bona, ja que t'estalvies les R.I:

$$\begin{aligned} A &= \underline{a0} + a1 + b0 + r \\ \text{C.Ali: } b0 &\rightarrow B(b0) \\ \text{C.Alt: } b0 \\ \text{R.I: } \text{nul}(b0) &\rightarrow \text{nul}(r) \end{aligned}$$

$$B = \underline{b0} + b1$$

$$\text{R.I: } A[a0] \subset B[a0]$$

**Relació binària 1:1 amb doble R.E.**

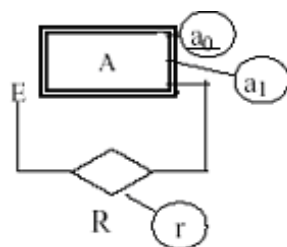
Aquest cas és especial. L'entitat A està tan íntimament relacionada amb l'entitat B, que es pot considerar la mateixa entitat. Per tant, no tindrem una taula per a A i altra per a B, sinó que només tindrem una única taula (li posarem el nom més adequat), on posarem tots els atributs:

**1a solució:** Posem els atributs de A en B:

**$R = B = A = \underline{b0} + b1 + a0 + a1 + r$**   
**C.Alt: a0**  
**VNN: a0**

**2a solució:** Posem els atributs de B en A:

**$R = A = B = \underline{a0} + a1 + b0 + b1 + r$**   
**C.Alt: b0**  
**VNN: b0**

**Relació unària 1:1 amb doble R.E.**

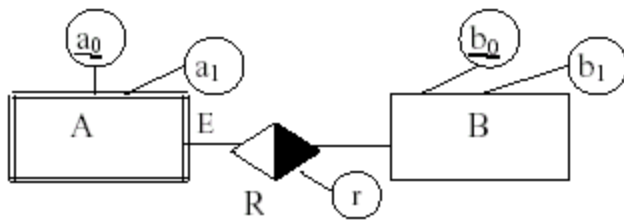
**$A = \underline{a0} + a1 + a0' + r$**   
**C. Ali: a0' → a0**  
**C. Alt: a0'**  
**VNN: a0'**

Nota: les següents relacions unàries...

- 1:1 amb una única E
- 1:M amb E a la part de 1
- 1:M amb doble E

... són realment unàries 1:1 amb doble E. Per tant, es resoldran igual.

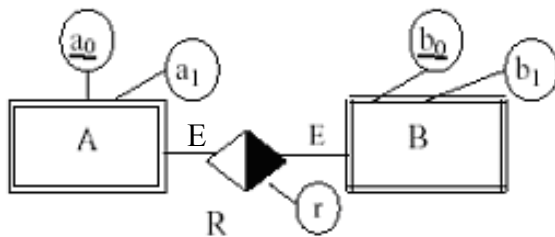
### Relació binària 1:M amb R.E a la part de l' "1"



$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1 + a0 + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
 R.I:  $\text{nul}(a0) \rightarrow \text{nul}(r)$

**R.I:  $A[a0] \subset B[a0]$**

### Relació binària 1:M amb doble R.E.

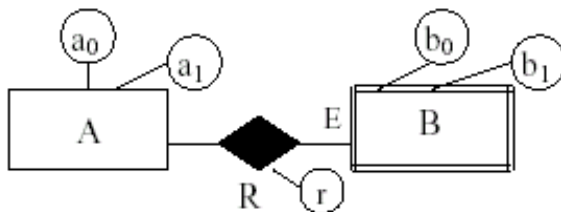


$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1 + a0 + r$   
 C.Ali:  $a0 \rightarrow A(a0)$

**VNN: a0**

**R.I:  $A[a0] \subset B[a0]$**

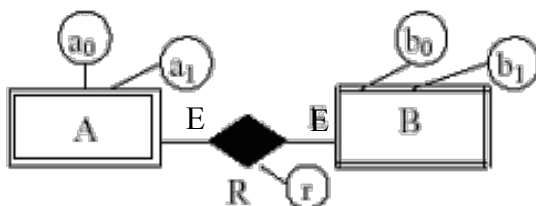
### Relació binària M:M amb una R.E.



$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1$   
 $R = \underline{a0} + \underline{b0} + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$

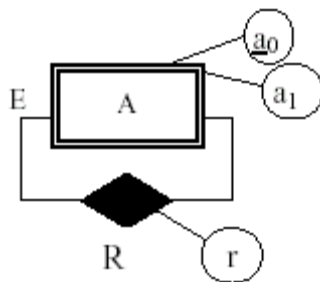
**R.I:  $B[b0] \subset R[b0]$**

### Relació binària M:M amb doble R.E.



$A = \underline{a0} + a1$   
 $B = \underline{b0} + b1$   
 $R = \underline{a0} + \underline{b0} + r$   
 C.Ali:  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$

**R.I:  $A[a0] \subset R[a0]$**   
 **$B[b0] \subset R[b0]$**

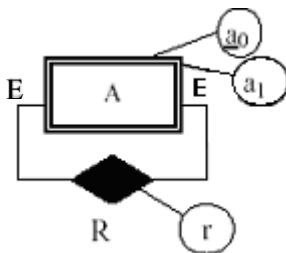
**Relació unària M:M amb una R.E.**

$$A = \underline{a0} + a1$$

$$R = \underline{a0} + a0' + r$$

C.Ali:  $a0 \rightarrow A(a0)$   
 $a0' \rightarrow A(a0)$

$$\mathbf{R.I: A[a0] \subset R[a0]}$$

**Relació unària M:M amb doble R.E.**

$$A = \underline{a0} + a1$$

$$R = \underline{a0} + a0' + r$$

C.Ali:  $a0 \rightarrow A(a0)$   
 $a0' \rightarrow A(a0)$

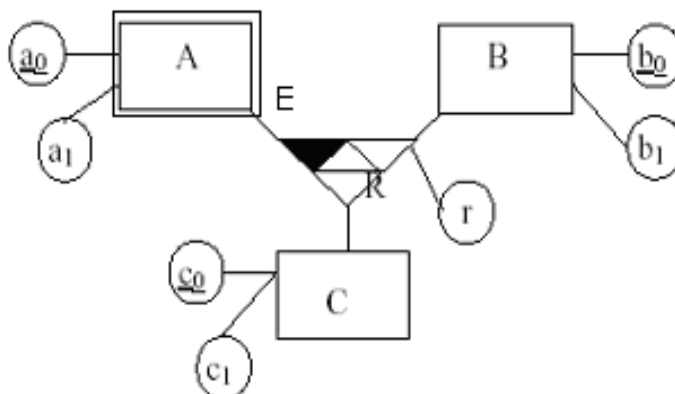
$$\mathbf{R.I: A[a0] \subset R[a0]}$$

$$\mathbf{A[a0] \subset R[a0']}$$

Exemple: entitat de carrers amb la relació “anar\_en\_direcció\_a”

**Relació ternària amb R.E.**

Per exemple, una 1:1:M, amb R.E. a la part del “Molts”:



$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$C = \underline{c0} + c1$$

$$R = \underline{a0} + \underline{b0} + \underline{c0} + r$$

C.Ali:  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$   
 $c0 \rightarrow C(c0)$

VNN:  $c0$   
 C.Alt :  $a0 + c0$

$$\mathbf{R.I: A[a0] \subset R[a0]}$$

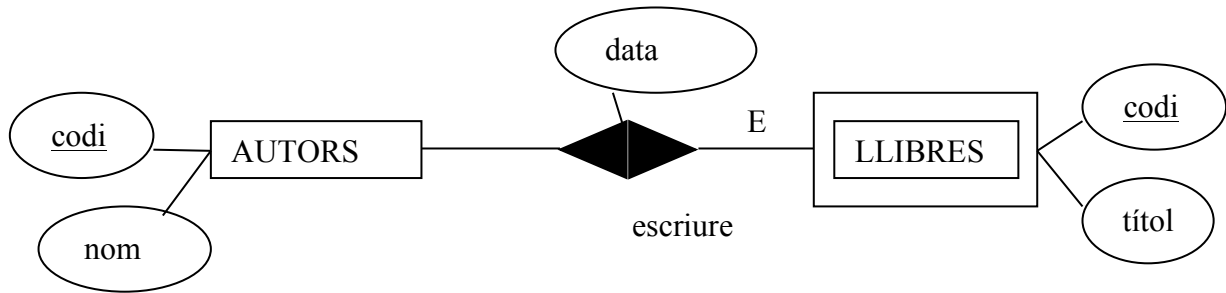
Donat que totes les ternàries generen taula per a la relació, qualsevol R.E. que posem en alguna de les entitats (independentment de la cardinalitat de la relació), provocarà una R.I. com l'anterior. És a dir:

R.I.

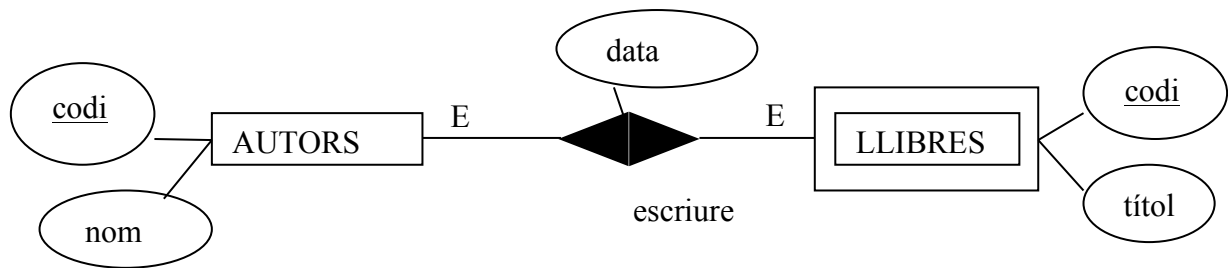
- Si A té una R.E. sobre R  $\rightarrow A[a0] \subset R[a0]$
- Si B té una R.E. sobre R  $\rightarrow B[b0] \subset R[b0]$
- Si C té una R.E. sobre R  $\rightarrow C[c0] \subset R[c0]$



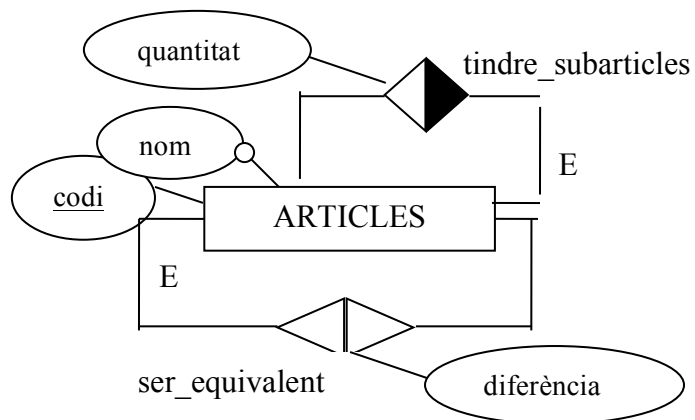
*Exercici 16: Passa a taules el següent E-R.*



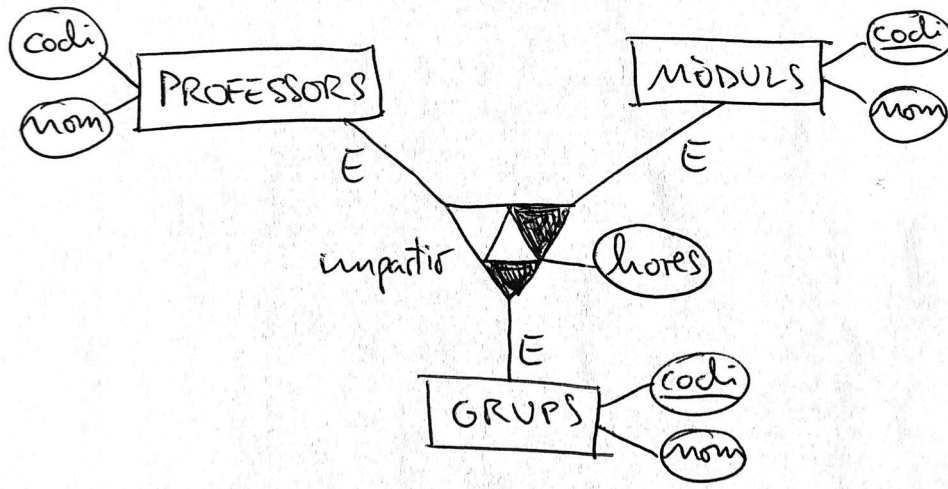
*Exercici 17: Passa a taules el següent E-R.*



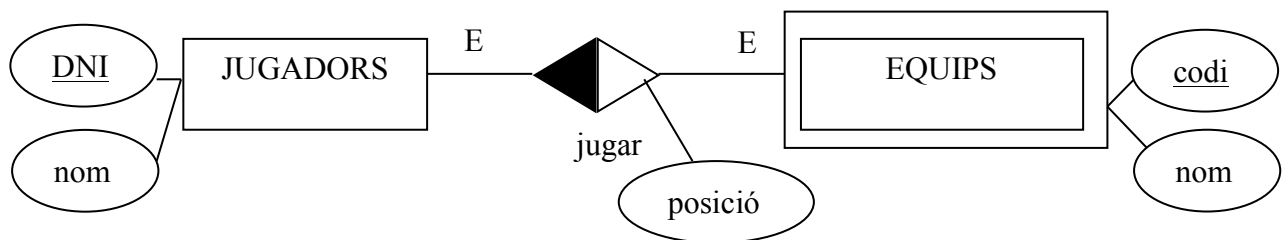
*Exercici 18: Passa a taules el següent E-R.*



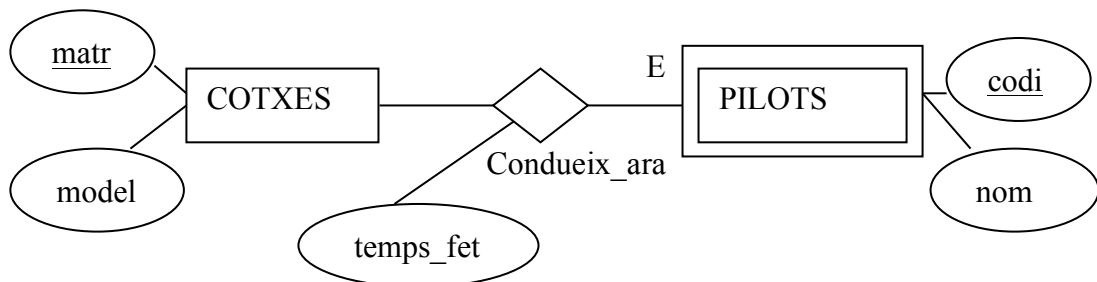
Exercici 19: Passa a taules el següent E-R.



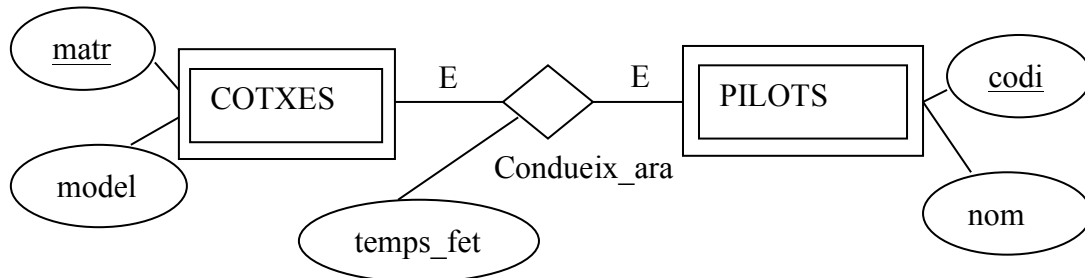
Exercici 20: Passa a taules el següent E-R.



Exercici 21: Passa a taules el següent E-R

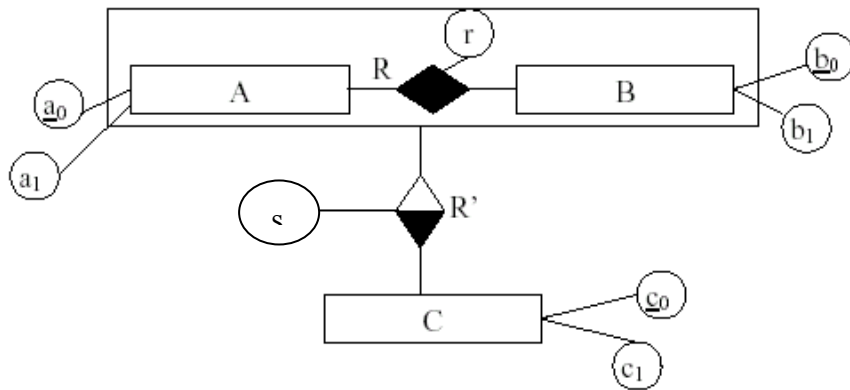


Exercici 22: Passa a taules el següent E-R



### 3.9. AGREGACIONS

Anem a veure com es resol a partir del següent exemple, on la relació de l'agregació és una binària M:M, però els passos a seguir serien els mateixos que si fóra una relació unària o ternària, tinga la connectivitat que tinga.



Passos a seguir:

1) Resoldre la relació de l'agregació sense tindre en compte la resta de l'esquema E-R, però **SEMPRE** ha d'eixir una taula nova per a la relació, encara que siga 1:1 o 1:M

$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$R = \underline{a0 + b0} + r$$

C.Ali:  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$

2) Continuar amb el disseny de la resta de l'esquema. Per a resoldre les relacions amb l'agregació, ho farem com si estigueren relacionades amb la taula que ha eixit abans en la relació de l'agregació (en este cas, R).

$$C = \underline{c0} + c1 + a0 + b0 + s$$

C.Ali:  $(a0 + b0) \rightarrow R(a0 + b0)$

R.I:  $\text{nul}(a0+b0) \rightarrow \text{nul}(s)$

Així doncs, quedaria:

$$A = \underline{a0} + a1$$

$$B = \underline{b0} + b1$$

$$R = \underline{a0 + b0} + r$$

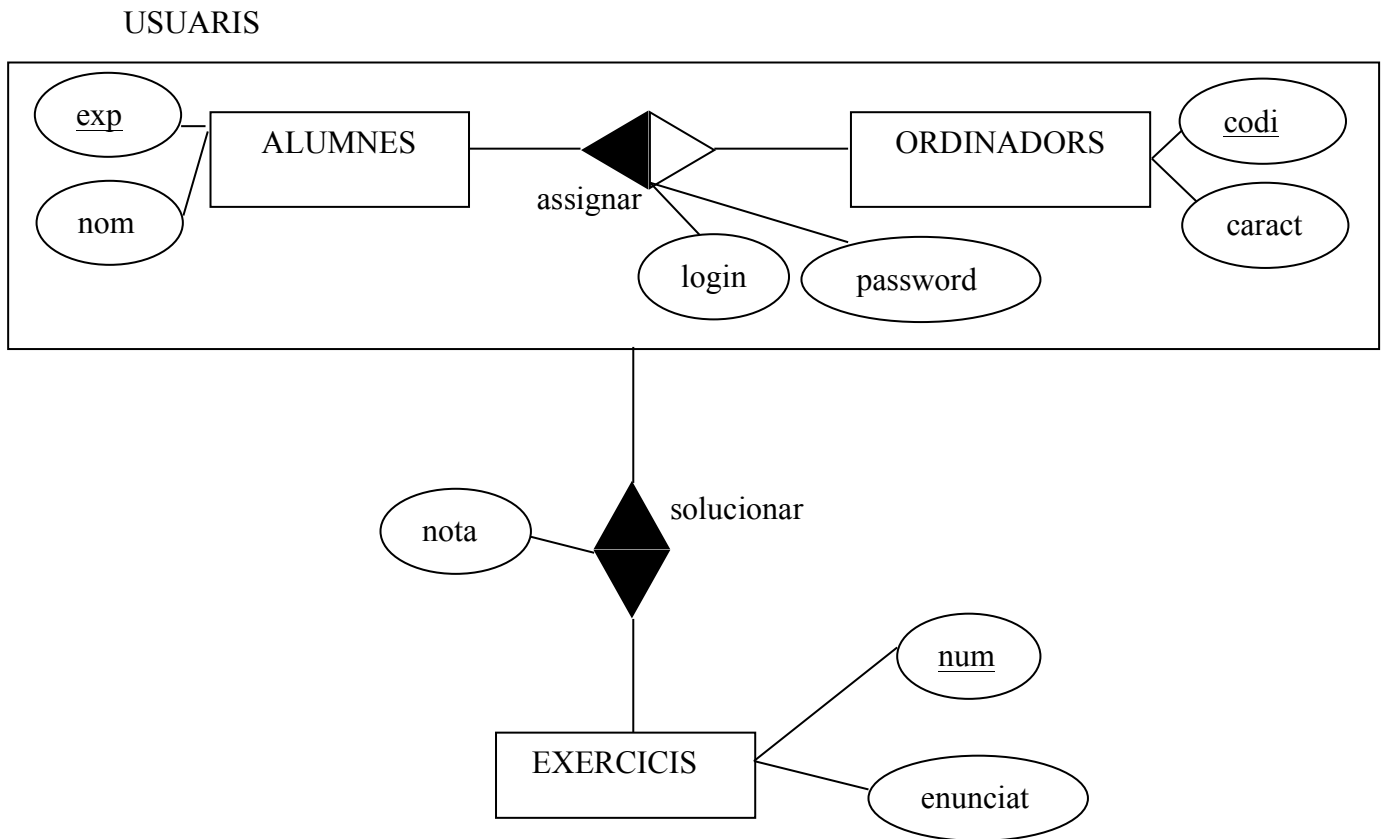
C.Ali:  $a0 \rightarrow A(a0)$   
 $b0 \rightarrow B(b0)$

$$C = \underline{c0} + c1 + a0 + b0 + s$$

C.Ali:  $(a0 + b0) \rightarrow R(a0 + b0)$

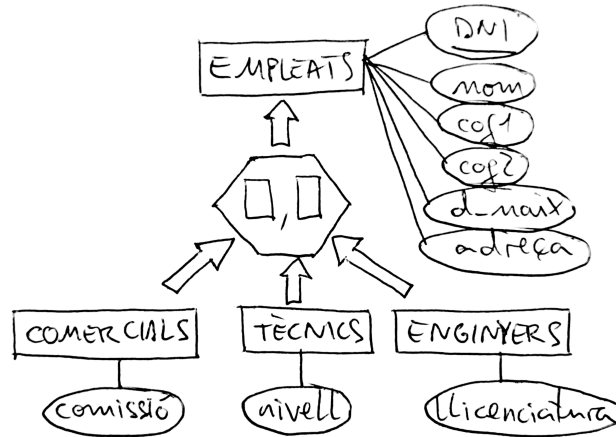
R.I:  $\text{nul}(a0+b0) \rightarrow \text{nul}(s)$

*Exercici 23: Passa a taules el següent E-R.*



### 3.10. ESPECIALITZACIONS / GENERALITZACIONS

Viem com seria el pas a taules a partir del següent exemple:



EMPLEATS = dni + nom + cog1 + cog2 + d\_naix + adreça

Una taula per a la superclasse (amb la seua clau i atributs)

COMERCIALS = dni + comissió

C. Ali: dni → EMPLEATS(dni)

TÈCNICS = dni + nivell

C. Ali: dni → EMPLEATS(dni)

ENGINYERS = dni + llicenciatura

C. Ali: dni → EMPLEATS(dni)

Una taula per a cada subclasse (amb la clau de la superclasse i els atributs propis de la subclasse). I claus alienes corresponents.

Si l'especialització és Total:

R. I: EMPLEATS[dni] = COMERCIALS[dni] ∪ TÈCNICS[dni] ∪ ENGINYERS[dni]

Si l'especialització és Disjunta:

R.I: COMERCIALS[dni] ∩ TÈCNICS[dni] = ∅

COMERCIALS [dni] ∩ ENGINYERS[dni] = ∅

TÈCNICS [dni] ∩ ENGINYERS [dni] = ∅

- Si és Total, una R.I.

- Si és Disjunta, les R.I. necessàries per a indicar que no estan superposades les ocurrències.

Per als altres tipus d'especialització (tant la parcial com la superposada) no posarem cap restricció d'integritat, ja que no suposen cap limitació a les dades.

### Altra forma, si l'especialització és Total i Disjunta

Cap taula per a la superclasse. I una taula per cada subclasse: amb la clau i els atributs de la superclasse més els atributs propis de la subclasse:

COMERCIALS = DNI + Nom + Cog1 + Cog2 + Data\_naix + Adreça + Comissió

TÈCNICS = DNI + Nom + Cog1 + Cog2 + Data\_naix + Adreça + Nivell

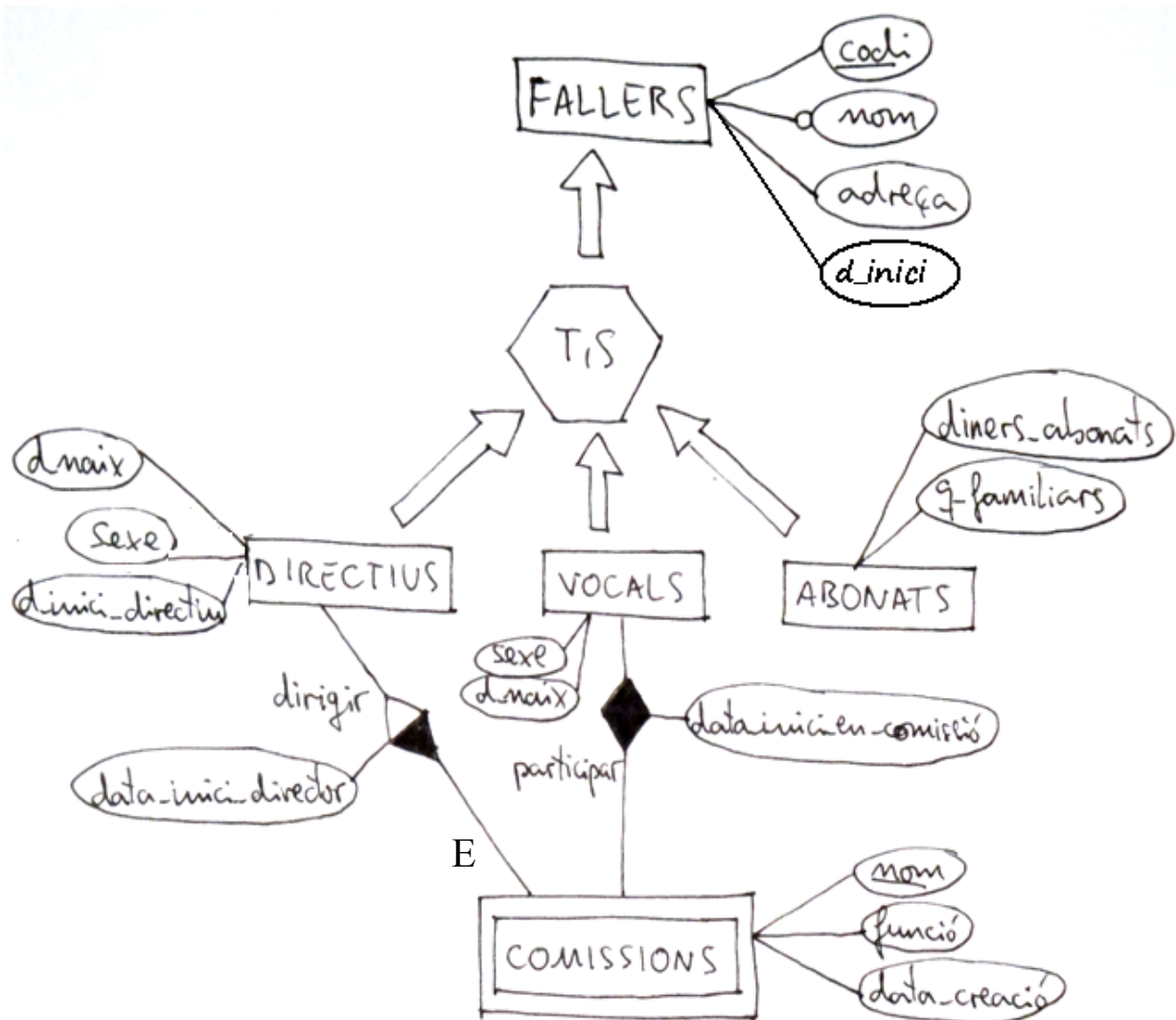
ENGINYERS = DNI + Nom + Cog1 + Cog2 + Data\_naix + Adreça + Llicenciatura

Però esta opció:

- no és vàlida ja que no podríem controlar que el dni no es puga repetir (per exemple, podria haver un comercial amb el mateix dni que u n tècnic)
- no és vàlida per a una especialització Parcial ja que no podríem inserir un empleat que no fóra de cap especialitat.
- no és vàlida per a una especialització Superposada, ja que si un empleat té moltes especialitats, les seues dades apareixerien de forma repetida en més d'una taula.

A més a més, té l'inconvenient que, quan busquem un empleat qualsevol, haurem de recórrer totes les taules de les subclasse per vore si està o no.

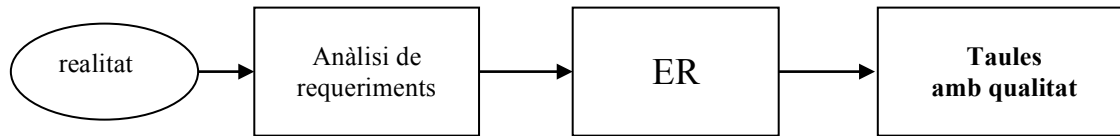
Exercici 24: Passa a taules el següent E-R.



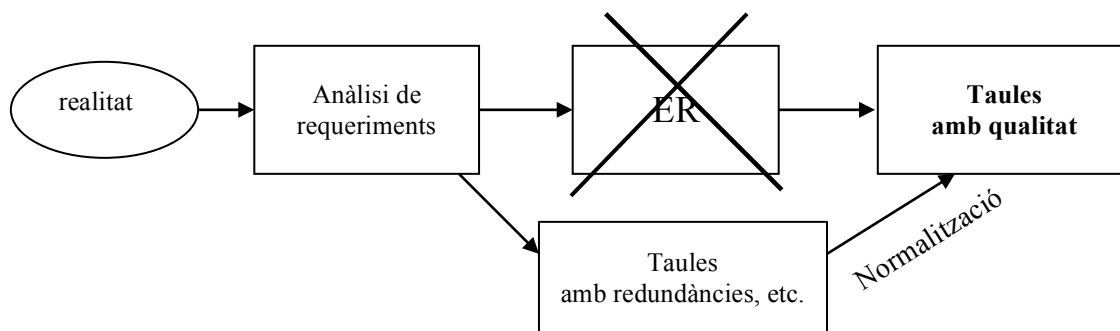
Si vols fer més exercicis de passar a taules esquemes E-R amb generalitzacions pots fer els que s'obtingueren en els exercicis del tema anterior.

## 4. Model Relacional: normalització

Hem vist que el model relacional consisteix en tindre un conjunt de taules amb qualitat (respectant els requeriments inicials i sense redundàncies). La forma ideal d'aconseguir-ho és la que hem vist: partint d'un esquema ER.



Però a vegades l'analista no fa el diagrama E-R sinó que fa uns esbossos inicials de com serien les taules. Eixes taules tindran problemes de redundàncies, etc. En eixe cas, cal aplicar la "normalització": tècniques per a transformar unes taules "mal fetes" en altres taules "amb qualitat".



La Normalització és la tècnica del model relacional que consistix en augmentar gradualment la qualitat de les taules, on tenim que:

**qualitat = simplicitat + no redundància.**

La Normalització ens dirà quines taules falten o sobren, les claus i atributs que tindran i les relacions entre les taules (claus alienes). Així obtindrem un esquema relacional que representarà correctament la informació i permetrà una adequada manipulació de les dades emmagatzemades en la BD.



#### 4.1. PLANTEJAMENT DEL PROBLEMA

Suposem que un analista ha de fer el disseny de la base de dades per a una biblioteca i, entre altres coses, li han dit:

“Necessitem un llistat de llibres on aparega el codi del llibre, el títol, les pàgines, el codi de l'autor, el nom de l'autor i la seua data de naixement”.

L'analista no ha volgut fer l'esquema ER i ha dissenyat directament esta taula:


LLIBRES = codi + títol + n\_pàgines + codi\_autor + nom\_autor + data\_naix\_autor

Què veiem estrany? Està bé? L'hauríeu dissenyat així? Quin problema té?

Totes les ocurrencies de LLIBRES que tinguen el mateix codi\_autor, també tindran el mateix nom\_autor i la mateixa data\_naix\_aur. Per tant, hi hauria molta informació repetida, amb els problemes que això suposa en una base de dades.

Això passa perquè donat 1 codi\_autor tenim només 1 nom\_autor i només 1 data\_naix\_autor. En eixe cas direm que “codi\_autor determina nom\_autor i data\_naix\_autor”. O bé, que “nom\_autor i data\_naix\_autor depenen de codi\_autor”. Es representa així:

LLIBRES = codi + títol + pàgines + codi\_autor + nom\_autor + data\_naix\_autor



The diagram shows a curved arrow pointing from 'codi\_autor' to 'nom\_autor' and another curved arrow pointing from 'codi\_autor' to 'data\_naix\_autor', indicating that 'codi\_autor' functionally determines both 'nom\_autor' and 'data\_naix\_autor'.

Com haureu imaginat, caldrà “fer bé” el disseny de la taula. És a dir: caldrà “normalitzar” la taula. Sense saber la teoria de normalització, podem imaginar que caldrà posar les dades de l'autor en una altra taula i, per tant, quedaria així:

LLIBRES = codi + títol + pàgines + **codi\_autor**

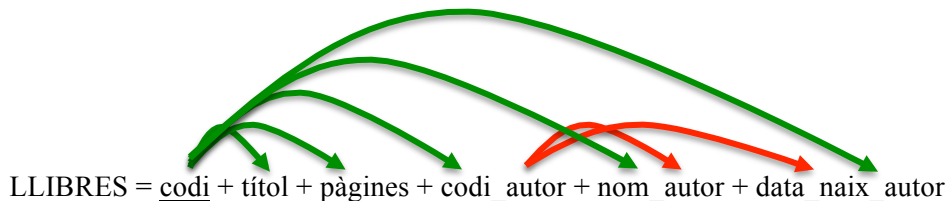
**C. Ali: codi\_autor → AUTORS(codi)**

AUTORS = codi + nom + data\_naix

Sense saber-ho, hem aplicat un dels passos de la normalització. Les “fletxetes” que eixien de codi\_autor s'anomenen “dependències funcionals”. Ara vorem què són exactament i després vorem els passos que cal seguir per a normalitzar una taula.

#### 4.1. DEPENDÈNCIES FUNCIONALS

Les Dependències Funcionals (DF) estableixen relacions de dependència entre els atributs d'una taula. Vegem-ho primer amb l'exemple anterior:



Eixes fletxes són les DF.

Definim ara què és una DF:

Siga la següent relació:  $T = a_0 + a_1 + \dots + a_i + \dots + a_j + \dots + a_n$

Es diu que l'atribut  $a_j$  depèn funcionalment de  $a_i$  o que  $a_i$  determina a  $a_j$  (i es representa  $a_i \rightarrow a_j$ ) si cada valor de  $a_i$  té associat un únic valor de  $a_j$  en la taula  $T$ . En altres paraules, si trobàrem dos tuples en  $T$  que coincidiren en el valor de  $a_i$ , també coincidirien en el valor de  $a_j$ .

En una taula poden haver dos tipus de DF:

- **Les que NO es lleven.** Són les que "ixen de la clau". En totes les taules eixiran DF de la clau cap a la resta d'atributs.
- **Les que SÍ que cal llevar.** Són les que "no ixen de la clau". En l'exemple anterior caldria llevar el nom\_autor i data\_naix\_autor de la relació LLIBRES, ja que depenen de codi\_autor (que no és clau de la taula).

**Normalitzar** consisteix bàsicament en llevar eixes DF que no ixen de la clau. Ja veurem detalladament com fer-ho.

Altres exemples:

Suposem que l'analista ha pensat en esta taula per a guardar la informació sobre els préstecs de la biblioteca:

PRÉSTECES = codi\_llibre + num\_soci + data + titol + nom\_soci

PRÉSTECES				
<u>codi_llibre</u>	<u>num_soci</u>	data	titol	nom_soci
HOMMAN	111	11-3-2017	L'home manuscrit	Pep Garcia
NOSVAL	111	11-3-2017	Nosaltres els valencians	Pep Garcia
HOMMAN	222	9-4-2017	L'home manuscrit	Maria Burguera
NOSVAL	111	7-4-2017	Nosaltres els valencians	Pep Garcia
NOSVAL	222	8-4-2017	Nosaltres els valencians	Maria Burguera

Què passaria si introduïrem les dades en una taula amb eixe disseny?

Ens adonem que sempre que apareix en codi\_llibre HOMMAN, apareix el títol "L'home manuscrit", sempre que apareix en codi\_llibre NOSVAL, apareixerà el títol "Nosaltres els valencians". És a dir: el títol depèn **NOMÉS** del codi\_llibre (no del num\_soci). Igualment, el nom\_soci depèn **NOMÉS** de num\_soci (no del codi\_llibre). És a dir: sempre que en un préstec aparega un mateix codi\_llibre, tindrem el mateix títol. I sempre que aparega un mateix num\_soci tindrem el mateix nom del soci. Per tant: donat 1 codi\_llibre tenim només 1 títol; i donat 1 num\_soci tenim només un nom\_soci:

PRÉSTECES = codi\_llibre + num\_soci + data + titol + nom\_soci



Per tant, caldrà llevar eixes DF ja que ixen de PART de la clau, NO de TOTA la clau). Després de normalitzar obtindrem estes taules, on veiem que ja no es repeteix la informació:

PRÉSTECES		
<u>codi_llibre</u>	<u>num_soci</u>	data
HOMMAN	111	11-3-2017
NOSVAL	111	11-3-2017
HOMMAN	222	9-4-2017
NOSVAL	111	7-4-2017
NOSVAL	222	8-4-2017

LLIBRES	
<u>codi</u>	titol
HOMMAN	L'home manuscrit
NOSVAL	Nosaltres els valencians

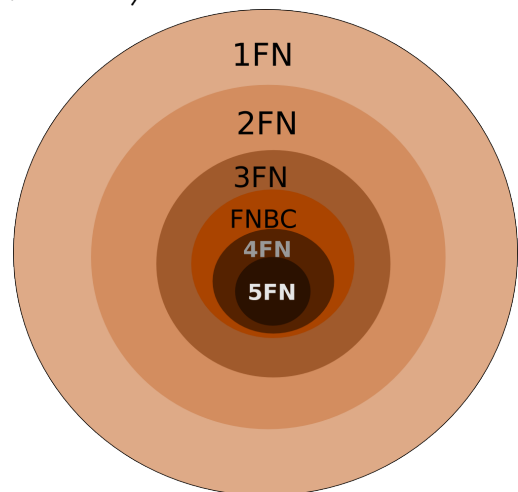
SOCIS	
<u>codi</u>	nom
111	Pep Garcia
222	Maria Burguera

## 4.2. FORMES NORMALS

Són les propietats que han de complir les relacions (taules).

El procés de normalitzar una relació consisteix en fer les següents fases i en eixe ordre:

- 1- Passar la relació a 1FN (i, si cal, optimitzar claus)
- 2- Passar la relació a 2FN
- 3- Passar la relació a 3FN
- 4- Passar la relació a FNBC
- 5- Passar la relació a 4FN
- 6- Passar la relació a 5FN



Així per exemple, si una relació està en 3FN compleix la norma 1FN, 2FN i 3FN. Només vorem fins la 3FN, ja que les altres formes normals no són tan importants i a vegades, presenten altres inconvenients.

Bàsicament, normalitzar una taula consisteix en:

- Per cada "origen de fletxa" (que no ix de tota la clau) crearem una taula, on:
  - L'origen de la fletxa serà la clau
  - Els destins de les fletxes seran els altres camps de la taula.
- En la taula original:
  - Llevarem els destins de les fletxes
  - Posarem claus alienes dels orígens de fletxes a les respectives taules.

PRÈSTECS = codi\_llibre + num\_soci + data + ~~títol~~ + ~~nom\_soci~~ + ~~tel\_soci~~



Normalització

PRÈSTECS = codi\_llibre + num\_soci + data  
 C. Ali: codi\_llibre → LLIBRES(codi)  
           num\_soci → SOCIS(num)

LLIBRES = codi + títol

SOCIS = codi + nom + tel

**FASE 1) Passar les taules a 1FN**

Def. teòrica: Una taula està en 1FN si no té atributs multivalents ni hi ha D.F. entre els components de la clau.

Def. informal: Una taula està en 1FN si:

- no té: {...}
- no té fletxes que van de dins a dins de la clau.

**Fase 1.1) Llevar atributs multivalents**

PERSONES = dni + nom + d\_naix + {telèfon}

Esta relació no està en 1FN perquè té l'atribut multivalent telèfon. És a dir, en esta relació, una persona pot tindre **molts** telèfons:

PERSONES						
<u>dni</u>	nom	d_naix	telèfon1	telèfon2	telèfon3	telèfon...
111	Pep	4-3-70	555111222	555111333	555111777	
222	Pepa	5-6-79	555222555	555222666		

El problema és que com no sabem quants telèfons pot tindre una persona, no podem reservar espai en la taula per a n telèfons.

Solució: el que fem en els atributs multivalents d'una entitat en l'ER: tindre les dades en 2 taules. Així, en compte de tindre els telèfons d'una persona en distintes columnes, els tindrem en distintes files (en la taula original no ho podíem fer perquè la clau era el dni).

Llevem el grup repetitiu {...} de la taula original.

PERSONES = dni + nom + d\_naix

TEL\_PERS = dni + telèfon

C.Ali: dni → PERSONES (dni)

PERSONES		
<u>dni</u>	nom	d_naix
111	Pep	4-3-70
222	Pepa	5-6-79

TEL_PERS	
<u>dni</u>	telèfon
111	555111222
111	555111333
111	555111777
222	555222555
222	555222666

Creem altra relació amb:

- Atributs: la clau de la taula original (dni) i **tots** els camps del grup repetitiu (en este cas només és el telèfon però en podrien ser més).
- Clau primària: formada per **tots** els camps (dni+tel)
- Clau aliena: l'atribut que hem "agafat" de la taula original (dni).

### Fase 1.2) Optimitzar claus

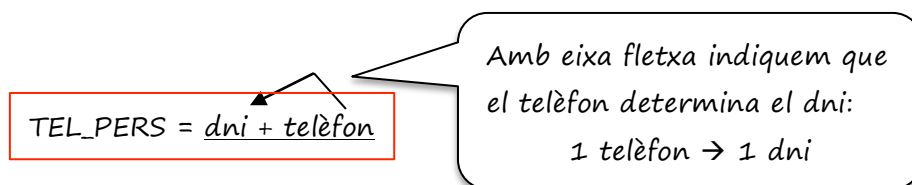
Si una relació té clau composta, consisteix en vore si podem fer que la clau no tinga tants camps. És a dir: cal llevar DF (fletxetes) entre els camps de la clau composta.

En l'exemple anterior, com una persona podia tindre molts telèfons, ens havia eixit la següent relació:

TEL\_PERS = dni + telèfon

Com la clau és composta, significa que 1 persona (dni) pot tindre MOLTS telèfons i que 1 telèfon pot correspondre a MOLTES persones (és com una relació M:M). Ja sabíem que 1 dni pot tindre molts telèfons, però... 1 telèfon es pot correspondre amb molts dnis?

Si els requeriments del sistema diuen que 1 telèfon només pot correspondre a moltes persones no haurem de fer res però si 1 telèfon pot correspondre a només 1 dni, tindrem una dependència funcional (el telèfon determina el dni), que expressarem així:

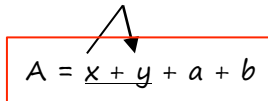


Vegem que tenim "una fletxa que va de dins a dins de la clau". En eixe cas, la relació TEL\_PERS no està en 1FN. Quan hem passat a 1FN hem "subratllat" més camps dels que tocaven. És a dir, haviem fet una relació M:M quan deuria haver sigut 1:M. Per tant, la clau no seria composta: caldria "optimitzar claus".

Com optimitzar claus:

Es resol diferent depenent de si la fletxa involucra tots els camps de la clau o no.

1.2.a) Si la fletxa involucra tots els camps de la clau

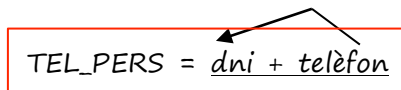


$$A = \underline{x + y} + a + b$$

Solució: llevar de la clau el destí de la fletxa (llevar-li el subratllat).

$$A = \underline{x} + y + a + b$$

Exemple:

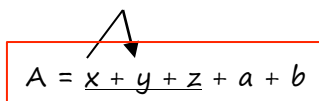


$$\text{TEL\_PERS} = \underline{\text{dni}} + \text{telèfon}$$

Solució: com la fletxa involucra tots els camps de la clau, la solució és llevar de la clau el destí de la fletxa.

$$\text{TEL\_PERS} = \underline{\text{telèfon}} + \text{dni}$$

1.2.b) Si la fletxa NO involucra tots els camps de la clau



$$A = \underline{x + y} + z + a + b$$

Solució:

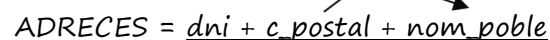
- Llevem de la relació el destí de la fletxa.
- Creem altra relació amb origen i destí de la fletxa.

$$A = \underline{x + z} + a + b$$

C. Ali:  $x \rightarrow A' (x)$

$$A' = \underline{x} + y$$

Exemple:



$$\text{ADRECES} = \underline{\text{dni}} + \text{c\_postal} + \text{nom\_poble}$$

Solució: com la fletxa NO involucra tots els camps de la clau (el dni no està involucrat), la solució és llevar de la relació el nom\_poble i crear altra relació amb el codi postal i el nom\_poble:

$$\text{ADRECES} = \underline{\text{dni}} + \text{c\_postal}$$

C. Ali:  $\text{c\_postal} \rightarrow \text{POBLES} (\text{cp})$

$$\text{POBLES} = \underline{\text{cp}} + \text{nom}$$

Esta forma de resoldre-ho serà la mateixa en la 2a i 3a forma normal.

Veiem que en la nova taula (POBLES) no estem obligats a mantindre els noms dels atributs.



## Exercicis sobre la 1FN

Exercici 26: Normalitza la següent relació.

CLIENTS = codi + nom + {tel}

Exercici 27: Normalitza la següent relació.

CLIENTS = codi + nom + {tel}

Restr. d'integritat: un telèfon només pertany a un client

Exercici 28: Normalitza la següent relació.

CLIENTS = codi + nom + {tel + operadora}

Restr. d'integritat: un telèfon només treballa en una operadora

Exercici 29: Normalitza la següent relació.

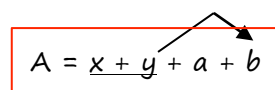
CLIENTS = codi + nom + {tel + operadora}

R. l:    - un telèfon només pertany a un client  
           - un telèfon només treballa en una operadora

### FASE 2) Passar les taules a 2FN

Def. teòrica: Una taula està en 2FN si està en 1FN i tot atribut no clau principal depèn de tota la clau i no de part d'ella. Per tant, només són sospitoses de no estar en 2FN aquelles taules que tinguen la clau principal composta.

Def. informal: Una taula està en 2FN si està en 1FN i no té "fletxes" (D.F.) de dins a fora de la clau.



Esta relació NO està en 2FN perquè hi ha una fletxa de dins de la clau (y) a fora (b).

Si una taula està en 1FN i no té la clau composta, segur que també està en 2FN.



Com passar a 2FN:

$$A = \underline{x + y} + a + b$$

Esta relació NO està en 2FN perquè hi ha una fletxa de dins de la clau (y) a fora (b).

Solució: igual que la forma 1.2.b:

- Llevem de la relació el destí de la fletxa.
- Creem altra relació amb origen i destí de la fletxa.

$$A = \underline{x + y} + a$$

C. Ali:  $y \rightarrow A' (y)$

$$A' = y + b$$

D'esta forma, desapareixen les redundàncies i possibles inconsistències, com veurem en l'exemple següent.

Exemple:

$$NOTES = \underline{codi\_assig + n\_exp} + nota + nom\_alumne$$

NOTES			
<u>codi_assig</u>	<u>n_exp</u>	nota	nom_alumne
BD	111	7	Pep Garcia
PRG	111	8	Pep Garcia
BD	222	9	Maria Burguera
FOL	111	7	Pep Garcia
PRG	222	8	Maria Burguera

- Hi ha redundàncies, ja que cada vegada que n\_exp val 111 el nom de l'alumne sempre és el mateix.

- Poden haver incoherències ja que potser que quan posem alguna nota a l'alumne 111 li posem Josep Garcia en compte de Pep Garcia.

Solució:

$$NOTES = \underline{codi\_assig + n\_exp} + nota$$

C.Ali:  $n\_exp \rightarrow ALUMNES$

$$ALUMNES = \underline{n\_exp} + nom\_alumne$$

NOTES		
<u>codi_assig</u>	<u>n_exp</u>	nota
BD	111	7
PRG	111	8
BD	222	9
FOL	111	7
PRG	222	8

ALUMNES	
<u>n_exp</u>	nom_alumne
111	Pep Garcia
222	Maria Burguera



## Exercicis sobre la 2FN

Exercici 30: Normalitza la següent relació

$COMPRES = \underline{art} + \underline{data} + preu + prov$

R.I: un article només es pot comprar a un proveïdor

Exercici 31: Normalitza la següent relació

$R = \underline{a} + \underline{b} + c + d$

D.F:  $b \rightarrow d$

Exercici 32: Normalitza la següent relació

$R = \underline{a} + \underline{b} + \underline{c} + d$

D.F:  $(b + c) \rightarrow d$

Exercici 33: Normalitza la següent relació

$ESCRIURE = \underline{cod\_autor} + \underline{cod\_llibre} + editorial + data + nom\_autor + nom\_llibre$

D.F:  $cod\_llibre \rightarrow (editorial + data + nom\_llibre)$

$cod\_autor \rightarrow nom\_autor$

Exercici 34: Normalitza les següents relacions

$ASSIG = \underline{codi} + nom + \{num\_treball + descr\_tre\}$

D.F:  $(codi + num\_treball) \rightarrow descr\_tre$

$NOTES = \underline{n\_exp} + \underline{assig} + nota\_ass + nom\_alumne + \{num\_treball + nota\_tre\}$

C.Ali:  $assig \rightarrow ASSIG(codi)$

D.F:  $n\_exp \rightarrow nom\_alumne$


$(assig + num\_treball + n\_exp) \rightarrow nota\_tre$

**FASE 3) Passar les taules a 3FN**

Def. teòrica: una taula està en 3FN si està en 2FN i cap dels atributs no clau (principal o alternativa) depenen entre ells.

Def. informal: una taula està en 3FN si està en 2FN i no té "fletxes" (D.F.) de fora a fora de la clau.

Com passar a 3FN:

$$A = \underline{x} + a + b + c$$


Esta relació NO està en 3FN perquè hi ha una D.F. de fora (b) a fora (c).

Solució: igual que la forma 1.2.b:


- Llevem de la relació el destí de la fletxa.
- Creem altra relació amb origen i destí de la fletxa.

$$A = \underline{x} + a + b$$

$$C. \text{ Ali: } b \rightarrow A' (b)$$

$$A' = \underline{b} + c$$

Exemple:

$$PERSONES = \underline{dni} + nom + c\_postal + població$$


Esta taula no està en 3FN, ja que hi ha "fletxa" de fora a fora de la clau: població depèn codi\_postal (i cap dels 2 atributs són claus alternatives).

PERSONES			
<u>dni</u>	nom	c_postal	població
111	Pep Garcia	46410	Sueca
222	Maria Burguera	46430	Sollana
333	Manolo Vendrell	46410	Sueca
444	Carles Ventura	46410	Sueca
555	Andreu Albors	46430	Sollana

Té els mateixos problemes de redundàncies i possibles incoherències que teníem en la 2FN ja que sempre que apareix el 46410 es repeteix que és de Sueca.

Solució:

$$PERSONES = \underline{dni} + nom + c\_postal$$

$$C. \text{ Ali: } c\_postal \rightarrow POBLES$$

$$POBLES = \underline{c\_postal} + població$$

POBLES	
<u>c_postal</u>	població
46410	Sueca
46430	Sollana

PERSONES		
<u>dni</u>	nom	c_postal
111	Pep Garcia	46410
222	Maria Burguera	46430
333	Manolo Vendrell	46410
444	Carles Ventura	46410
555	Andreu Albors	46430



## Exercicis sobre la 3FN

Exercici 35: Normalitza la següent relació

$$R = \underline{a} + b + c + d$$

D.F:  $c \rightarrow d$

Exercici 36: Normalitza la següent relació

$$R = \underline{a + b} + c + d + e + f + g + h + \{i\}$$

D.F:  $c \rightarrow (d + e)$   
 $g \rightarrow h$   
 $b \rightarrow f$   
 $i \rightarrow b$

Exercici 37: Normalitza la següent relació

$$R = \underline{a + b + c} + d + e + \{f + g\}$$

D.F:  $a \rightarrow b$   
 $c \rightarrow d$

Exercici 38: Normalitza la següent relació

$$A = \underline{a_0} + \{a_1 + a_2\} + c + d$$

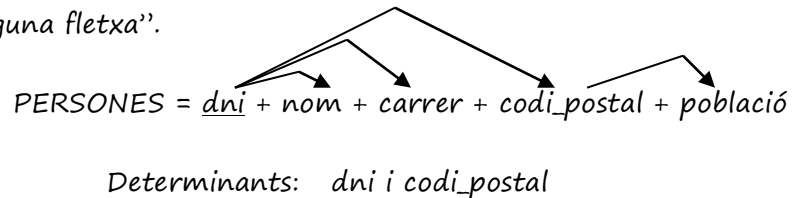
D.F:  $c \rightarrow d$   
 $a_1 \rightarrow a_2$

Nota: Falten per veure les formes normals FNBC, 4FN i 5FN. Com que són poc freqüents i, a més, passar a eixes formes normals pot presentar inconvenients, només vorem la de FNBC per a saber en què consisteix, però no es demanarà en l'examen.

**FASE 4) Passar les taules a FNBC (Forma Normal de Boyce Codd)**

Def. teòrica: una taula està en FNBC si està en 3FN i tot determinant és clau (principal o alternativa).

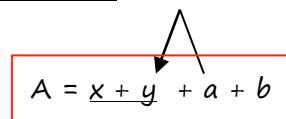
Determinant: atribut (simple o compost) del qual algun altre atribut depèn funcionalment d'ell. És a dir: un atribut del qual "ix alguna fletxa".



Def. informal: una taula està en FNBC si està en 3FN i no té fletxes de fora a dins de la clau.

Si una taula està en 3FN i no té la clau composta, segur que està en FNBC.

Com passar a FNBC:



Esta relació NO està en FNBC perquè hi ha una fletxa de fora de la clau (a) a dins (y).

Solució: **paregut** a la forma 1.2.b:

- Llevem de la relació el destí de la fletxa i l'origen de la fletxa també és clau.
- Creem altra relació amb origen i destí de la fletxa.
- Ens apareix una R.I.

$$A = \underline{x + a} + b$$

$$C. Ali: a \rightarrow A' (a)$$

$$A' = \underline{a} + y$$

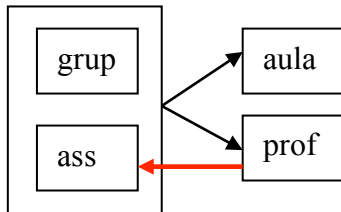
R.I: cal que es complisca la D.F:  $(x + y) \rightarrow a$

Per a resoldre els problemes d'una relació que no està en FNBC, l'hem de convertir en eixes dos relacions però tenim un problema: que ja no queda garantit que donat 1 ocurrència de (x,y) tinguem només 1 ocurrència de a. Per això ho hem hagut de posar en una R.I.

Exemple:

IMPARTIR = grup + assignatura + professor + aula

És a dir:



- Donat 1 grup i 1 assignatura, només és impartit per 1 professor (restricció de la clau principal)
- Cada professor imparteix només 1 assignatura (restricció de la DF: professor → assignatura)

Problemes per no complir FNBC:

IMPARTIR			
<u>grup</u>	<u>assignatura</u>	professor	aula
1DAM	BD	Abdó	1
1DAM	FOL	J.Vicent	1
1ASIX	BD	Pastor	2
1ASIX	FOL	J.Vicent	1
1SMX	FOL	Maria	3

- Redundància: cada vegada que apareix Abdó tindrem repetit que imparteix BD.

- Incoherència: No podem garantir que un professor només imparteixi una assignatura.

- Pèrdua d'informació: Suposem que en el nou pla d'estudis ja no es veu FOL en 1SMX. Per tant, eliminaríem el registre corresponent. Però també estariem perdent la informació sobre que l'assignatura que imparteix Maria és FOL.

Solució:

IMPARTIR = grup + professor + aula

C.Ali: professor → PROFESSORS (professor)

PROFESSORS = professor + assignatura

R.I: cal que es complisca la D.F: (grup + assignatura) → professor

IMPARTIR		
<u>grup</u>	<u>professor</u>	aula
1DAM	Abdó	1
1DAM	J.Vicent	1
1ASIX	Pastor	2
1ASIX	J.Vicent	1
1SMX	Maria	3

PROFESSORS	
<u>professor</u>	assignatura
Abdó	BD
J.Vicent	FOL
Pastor	BD
Maria	FOL

Per què fa falta posar eixa R.I.?

Perquè en la distribució de les noves taules es perd la restricció que teníem inicialment: donat 1 grup i 1 assignatura, només és impartit per 1 professor.

És a dir, podríem afegir este últim registre en la taula IMPARTIR però seria incorrecte, ja que estaríem permetent que al grup 1DAM l'assignatura BD la donen 2 professors (Abdó i Pastor).

IMPARTIR		
grup	professor	aula
1DAM	Abdó	1
1DAM	J.Vicent	1
1ASIX	Pastor	2
1ASIX	J.Vicent	1
1SMX	Maria	3
1DAM	Pastor	1

PROFESSORS	
professor	assignatura
Abdó	BD
J.Vicent	FOL
Pastor	BD
Maria	FOL

Per això cal posar la R.I.: D.F. (grup + assignatura) → professor

Amb eixa R.I. estem indicant que caldria controlar per programa que:

- ✓ Quan s'insereixca un nou registre en la taula IMPARTIR (o es modifiqui un registre existent), caldrà comprovar que: si l'assignatura que dona eixe professor és donada per un altre professor, eixe altre professor no pot estar impartint classe a eixe grup.
- ✓ Quan es modifiqui un registre en la taula PROFESSORS, caldrà comprovar que: si l'assignatura que dona eixe professor és donada per un altre professor, eixos 2 professors no poden estar impartint classe a un mateix grup.



## Exercicis sobre la FNBC

Exercici 39: Normalitza la següent relació

$$R = \underline{a + b} + c + d + e$$

$$D.F: (c + d) \rightarrow b$$

Exercici 40: Normalitza la següent relació

$$R = \underline{a + b + c} + d + e$$

$$D.F: d \rightarrow (b + c)$$

Exercici 41: Normalitza el següent esquema relacional

$$A = \underline{a0} + a1 + a4 + \{a5\}$$

$$A1 = \underline{a0} + a1$$

$$A2 = \underline{a0} + a2$$

$$B = \underline{b0} + b1 + b2$$

$$R = \underline{a0 + b0} + r1 + r2 + r3$$

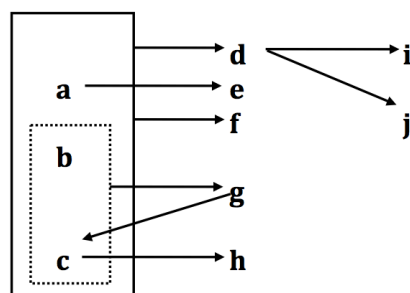
$$D.F: \quad a0 \rightarrow r1$$

$$r3 \rightarrow b0$$

Exercici 42: Normalitza la següent relació

$$R = \underline{r} + a + b + c + d + e + f + g + h + i + j$$

Graf de dependències funcionals:





FASE N) Llevar altres redundàncies

- Eliminar redundàncies intrataula: llevarem els camps que es puguin obtenir a partir d'altres camps de la mateixa taula.

Exemple:

NOTES = dni + assig + nota1 + nota2 + nota3 + ~~nota\_mitja~~

Es pot eliminar *nota\_mitja* ja que és redundant: es pot obtenir a partir d'altres atributs de la mateixa taula:  $(\text{nota1} + \text{nota2} + \text{nota3}) / 3$

- Eliminar redundàncies intertaula: llevarem camps que es puguin obtenir a partir de camps d'altres taules.

Exemple:

NOTES = dni + assig + nota

ALUMNES = dni + nom + ~~nota\_mitja~~

Es pot eliminar *nota\_mitja* ja que és redundant: la nota mitja d'un alumne es pot obtenir sumant les seues notes de la taula NOTES i dividint-ho per la quantitat de notes que té (quantitat d'assignatures).

Ara bé: és discutible l'eliminació "perquè sí" d'estos camps, ja que l'eliminació de la redundància intertaula sol provocar ralentització de les consultes. Per tant, es tracta de buscar un compromís entre puresa i eficiència. Caldria estudiar cada cas en particular.

- Fussionar taules que compartixen la mateixa clau

Només fusionarem si no trenquem el resultat de l'opcionalitat i especialització obtinguts en la primera fase de la normalització.

Exemple:

ALUMNES\_DADES\_PERSONALS = codi + nom + d\_naix + tel

ALUMNES\_DADES\_ACADÈMIQUES = codi + curs + grup + repetidor\_sn

Solució:

ALUMNES = codi + nom + d\_naix + tel + curs + grup + repetidor\_sn

## 4.3. ESQUEMA-RESUM DE COM NORMALITZAR

NORMALITZACIÓ de BD relacionals		No està en eixa Forma Normal si tenim...	Com es passa a eixa Forma Normal?	
1FN	No ha d'haver " {... } "  Llevarem multivalents	$A = \underline{x} + \{\text{mul}\} + b$	- Llevem el multivalent.  - Creem altra relació amb la clau i l'atribut multivalent (sense { }) fent la clau composta.	$A = \underline{x} + b$  $A' = \underline{x + \text{mul}}$ C. Ali: $x \rightarrow A(x)$
	No ha d'haver "fletxa dins→dins"  Optimitzarem la clau	$A = \text{ori} + \text{des} + a$ (La fletxa ocupa tota la clau)	- Llevem el subratllat del destí de la fletxa.	$A = \underline{\text{ori}} + \text{des} + a$
		$A = \underline{x + \text{ori}} + \text{des} + a$ (La fletxa no ocupa tota la clau)		$A = \underline{x + \text{ori}} + a$ C. Ali: $\text{ori} \rightarrow A'(\text{ori})$  $A' = \underline{\text{ori}} + \text{des}$
2FN	No ha d'haver "fletxa dins→fora"	$A = \underline{x + \text{ori}} + \text{des} + a$	- Llevem el destí de la fletxa.  - Creem altra relació amb origen i destí de la fletxa.	$A = \underline{x + \text{ori}} + a$ C. Ali: $\text{ori} \rightarrow A'(\text{ori})$  $A' = \underline{\text{ori}} + \text{des}$
3FN	No ha d'haver "fletxa fora→fora"	$A = \underline{x + \text{ori}} + \text{des} + a$ (“ori” no es C. Alt)		$A = \underline{x + \text{ori}} + a$ C. Ali: $\text{ori} \rightarrow A'(\text{ori})$  $A' = \underline{\text{ori}} + \text{des}$
FNBC	No ha d'haver "fletxa fora→dins"	$A = \underline{x + \text{des}} + \text{ori} + a$ (“ori” no es C. Alt)	- Igual que la solució anterior però l'origen de la fletxa també se subratlla.  - R.I amb la D.F. “perduda”	$A = \underline{x + \text{ori}} + a$ C. Ali: $\text{ori} \rightarrow A'(\text{ori})$  $A' = \underline{\text{ori}} + \text{des}$  - R.I: D.F: $(x + \text{des}) \rightarrow \text{ori}$

#### 4.4. RESOLUCIÓ D'EXERCICIS DE NORMALITZACIÓ

PASSOS	EXEMPLE	
Volem normalitzar les següents taules:	FACTURES = <u>num</u> + c_cli + dni + nom + tel_cli + data + {lin + art + des + preu + quant} ARTICLES = <u>codi</u> + des + familia + des_fam	
<b>1) Suposicions</b>	En una factura no poden haver línies de factura amb el mateix article. El preu sempre és el mateix per a cada article. ...	
<b>2) Detectar les DF</b> (que no isquen de CP ni C. Alt)	D.F en la taula FACTURES : c_cli → dni, nom, tel_cli art → des, preu num + lin → art, quant	D.F. en la taula ARTICLES: familia → des_fam
<b>3) Normalitzar cada taula</b>	Normalitzant taula FACTURES obtenim: FACTURES = <u>num</u> + c_cli + data C. Ali: c_cli → CLIENTS(c_cli)  LINIES_FAC = <u>num</u> + <u>lin</u> + art + quant C. Ali: num → FACTURES(num) art → ARTICLES(art)  ARTICLES2 = <u>art</u> + des + preu  CLIENTS = <u>c_cli</u> + nom + tel	Normalitzant taula ARTICLES obtenim: ARTICLES = <u>codi</u> + des + família C. Ali: família → FAMILIES(família)  FAMÍLIES = <u>família</u> + des
<b>4) Obtindre l'esquema integrat final</b> - Fusionar taules amb la mateixa clau (si no són especialització) - Posar noms coherents a taules i atributs	FACTURES = <u>num</u> + c_cli + data C. Ali: c_cli → CLIENTS(c_cli)  LINIES_FAC = <u>num</u> + <u>lin</u> + art + quant C. Ali: num → FACTURES(num) art → ARTICLES(art)  ARTICLES = <u>art</u> + des + preu + fam C. Ali: fam → FAMILIES(fam)  CLIENTS = <u>c_cli</u> + nom + tel  FAMÍLIES = <u>fam</u> + des	
<b>5) Dibuixar el graf de dependències entre les taules</b> (ordre en el qual es deurien crear les taules, segons les claus alienes)	<pre> graph LR     FAMÍLIES --&gt; ARTICLES     CLIENTS --&gt; FACTURES     ARTICLES --&gt; LINIES_FAC     FACTURES --&gt; LINIES_FAC           </pre>	



## Exercicis de normalització

### Exercici 43:

En un institut tenen el següent llistat amb informació d'alumnes, les assignatures on estan matriculats i a quines aules van.

DNI	COGNOMS	NOM	NOTA_M	ASSIGNATURA	NOTA	CURS	AULA	LLOC
34561000	Peris Sastre	Josep	6.0	Literatura I	7	1	209	2n pis pta 1
				Socials I	5	1	309	3r pis pta 2
23234320	Marqués Fuster	Maria	7.0	Literatura I	6	1	209	2n pis pta 1
				Socials I	7	1	309	3r pis pta 2
				Naturals II	8	2	409	3r pis pta 2
25134562	Garcia Marqués	Neus	6.0	Matemàtiques I	7	2	209	2n pis pta 1
				Naturals II	8	2	409	3t pis pta 2
				Matemàtiques II	3	2	209	2n pis pta 1

Ens han explicat eixa taula dient-nos que:

- ✓ L'atribut **nota\_m** és la nota mitjana que ha tret l'alumne entre les notes de les seues assignatures.
- ✓ L'atribut **assignatura** representa la identificació de les assignatures en què es troba matriculat cada alumne.
- ✓ L'atribut **nota** és la nota que ha tret cada alumne en cada assignatura.
- ✓ L'atribut **curs** representa el curs en què s'imparteix una assignatura. Una assignatura està assignada a un únic curs.
- ✓ L'atribut **aula** representa les aules en què s'imparteixen les assignatures. Una assignatura només es pot impartir en un aula.
- ✓ L'atribut **lloc** és l'espai físic que ocupa eixa aula dins del centre.

Es demana:

- a) Defineix en notació de De Marco la relació ALUMNES que tindrà tota eixa informació (on la clau serà el DNI).
- b) Detecta les possibles dependències funcionals a partir de les dades de la taula i de l'explicació que ens han donat.
- c) Normalitza eixa relació fins a FNBC no redundant.

**Exercici 44:**

L'hotel de Sueca vol informatitzar les factures i ens n'ha deixat una per a que tinguem clar quina informació vol guardar de cada factura:

<u>HOTEL CIUTAT DE SUECA</u>			<u>Factura:</u> 1934		
<u>Data:</u> 15 novembre 2004					
<u>Client:</u> Andreu Torlà Gomis			<u>Habitació:</u> 202		
<u>DNI:</u> 12.345.678			<u>Tipus:</u> Doble		
<u>Domicili:</u> C/ La Punta, 96			<u>Tarifa:</u> 2		
46410 Sueca					
<u>Telèfon:</u> 555 333 666					
<u>Nombre de persones:</u> 2					
<u>Lín</u>	<u>Codi</u>	<u>Concepte</u>	<u>Quantitat</u>	<u>Preu unitari</u>	<u>Total</u>
1	02	Mitja pensió	3	50,00	150,00
2	45	Dinar	2	12,00	24,00
3	23	Consumició bar	1	4,00	4,00
				Base imposable:	178,00
				IVA (16%):	28,48
					-----
				Total euros:	206,48

Es demana:

- Defineix en notació de De Marco la relació FACTURES que tindrà tota eixa informació (on la clau serà el número de la factura).
- Detecta les possibles dependències funcionals a partir de les dades de la factura i de l'experiència que tingues en les factures i els hotels. En cas de dubte, indica les possibles suposicions que cregues convenient.
- Normalitza eixa relació fins a FNBC no redundant.
- A partir de la factura fes l'esquema E-R. A continuació fes el pas a taules. Comprova si t'ha donat el mateix resultat que en la normalització.