ASIX - M02. Gestió de Bases de Dades
DAW - M02. Bases de Dades

UF1. Introducció a les Bases de Dades

Autor: *Robert Ventura Vall-llovera*

Data creació: Setembre2015
Actualització: Octubre 2018

Aquest document està creat en base el document original de Marc Nicolau Reixach i Úrsula Heredia Garcia
No està permesa la reproducció total o parcial d'aquest document, ni el seu tractament informàtic, ni la transmissió de cap forma o per qualsevol mitjà, ja sigui electrònic, mecànic, per fotocòpia, per registre o altres mètodes, sense el permís previ i per escrit del titular

NF3.
Model relacional

Contingut

Contingut ii

A 1. Model relacional 3
1.1. Conceptes bàsics 3
1.1.1. Introducció 3
1.1.2. Estructura de les dades 4
Visió informal d'una relació 4
Visió formal d'una relació 4
Domini 4
Esquema de relació 5
Relació 6
Esquema d'una base de dades 7
Exemplar d'una base de dades 7
1.1.3. Clau candidata, clau primària i clau alternativa 9
Clau candidata 9
Clau primària 10
Clau alternativa 10
1.2. Restriccions del model relacional 11
1.2.1. Restriccions de domini 11
1.2.2. Restriccions de clau 11
1.2.3. Restricció d'integritat d'entitat 11
1.2.4. Restricció d'integritat referencial 11
1.3. Operacions del model relacional 13
1.3.1. Actualització de dades 13
I. Inserir 13
II. Eliminar 14
III. Modificar 15
1.3.2. Consulta de dades 16

A 2. Transformació ER a relacional 17
2.1. Obtenció de l'esquema de base de dades relacional 17
2.2. Exemples d'esquemes de base de dades relacionals 28
2.2.1. BD Biblioteca 28
2.2.2. BD Llibreria 31

A 1. Model relacional

1.1. Conceptes bàsics

1.1.1. Introducció

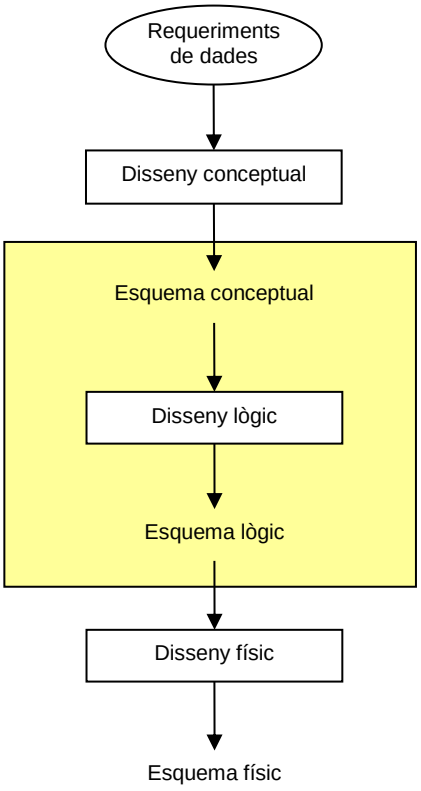
El **model relacional** és un model de dades i, com a tal, té en compte els tres següents aspectes de les dades:

- **L'estructura**, que ha de permetre representar la informació que interessa del món real.
- La **manipulació**, mitjançant les operacions d'actualització i consulta de les dades.
- La **integritat**, que s'aconsegueix establint regles d'integritat: és a dir, condicions que han de complir les dades.

Un **S**istema de **G**estió de **B**ases de **D**ades **R**elacional (SGBDR) dóna suport a la definició de les dades mitjançant l'estructura de les dades del model relacional, així com a la manipulació d'aquestes dades amb les operacions del model; a més a més assegura que es compleixin les regles d'integritat que el model relacional estableix.

El model relacional va ser definit per Edgar Frank Codd, l'any 1970. El principal objectiu del model de dades relacional és permetre que la base de dades es vegin per **l'usuari com una estructura lògica que consisteix en un conjunt de relacions i no com una estructura física d'implementació**. Això ajuda a aconseguir un alt grau d'independència de les dades.

Es basa en el concepte matemàtic de relació (relacionat amb teoria de conjunts i lògica de predicats)



1.1.2. Estructura de les dades

El **model relacional** representa una base de dades com un conjunt de relacions que representa la informació que interessa del món real.

Visió informal d'una relació

Nosaltres de manera informal, les relacions les anomenarem **taules** de dades. En la següent taula es mostra una vista tabular d'una relació que conté dades d'alumnes.

ALUMNE

DNI	nom	cognom1	cognom2	telefon	c_postal	poblacio
40175678	Pere	Pi	Vidal	972266789	17800	Olot
40347890	Josep	Pujol	Puig	972293445	17857	La Canya
77900707	Anna	Puig	Comas	972278796	17800	Olot
...	...	...	...	...	...	...

En la terminologia del model relacional parlarem de **registres** o **tuples** (corresponents a les files de les taules) i **camps** o **atributs** (fent referència a les columnes de les taules).

Cada fila de la taula conté un conjunt de valors de dades relacionades entre sí; en l'exemple, són les dades corresponents als alumnes. La taula té un nom (ALUMNES) i també té un nom per cadascuna de les seves columnes (dni, nom, cognom1, etc.).

El nom de la taula i de les columnes ajuden a entendre el significat dels valors que conté la taula. Cada columna conté valors d'un cert domini; per exemple, la columna dni conté valors del domini *númerosDNI* (tots els possibles valors que pot tenir un DNI).

Si definim les relacions de forma més precisa, ens adonarem que tenen unes característiques importants que, en la visió superficial que s'ha vist, queden amagades. Aquestes característiques són les que motiven que el concepte de relació sigui totalment diferent al de taula com sempre l'hem entès, tot i que a primera vista, relacions i taules puguin ser similars.

Visió formal d'una relació

En aquest apartat es defineix una relació formalment i altres conceptes que hi estan molt vinculats, com per exemple **domini**, **esquema de relació**, etc.

Domini

Un **domini *D*** és un conjunt de valors indivisibles (el model relacional no accepta valors compostos ni multivaluats) que representa els possibles valors que pot agafar un camp o columna.

Els dominis poden ser de dos tipus:

1) **Dominis predefinits**, que correspon als tipus de dades que normalment proporcionen els llenguatges de bases de dades, com per exemple els enters, les cadenes de caràcters, els reals, booleans, etc.

2) **Dominis definits per l'usuari**, que poden ser més sofisticats. Tota definició de domini ha de tenir, com a mínim, el nom del domini i la descripció dels valors que en formen part.

   Exemple**:** un usuari pot definir un domini per les edats dels treballadors que s'anomeni ***dom_edat*** i que contingui els valors enters entre 16 i 65.

Esquema de relació

Un **esquema de relació** (o esquema de la taula) $\mathbf{R(A_1,A_2,...,A_n)}$ es composa d'un nom de relació **R** i una llista d'atributs $\mathbf{A_1,A_2,...,A_n}$.

Un esquema de relació serveix per descriure una relació.

Exemple: nom i conjunt d'atributs de la relació ALUMNE

Partint de l'exemple anterior, el nom de la relació és *ALUMNE* i la llista d'atributs és *{DNI, nom, cognom1, cognom2, telèfon, c_postal, població}*. Per tant, l'esquema de relació és:

ALUMNE(*DNI, nom, cognom1, cognom2, telefon, c_postal, poblacio).*

Es poden posar els atributs en qualsevol ordre.

Els possibles valors que pot agafar cada atribut A_i formen el domini de A_i expressat per $\mathbf{dom(A_i)}$.

Exemple: domini de l'atribut DNI

Partint de l'exemple, el dom(*DNI*)=*númerosDNI*.

En una mateixa relació no pot haver-hi dos atributs amb el mateix nom. En canvi, si que hi pot haver el mateix nom d'atribut en relacions diferents. Per altra banda, els dominis en una mateixa relació no tenen perquè ser tots diferents: pot haver-hi atributs diferents amb el mateix domini.

El **grau d'una relació** (d'una taula) és el número d'atributs que té el seu esquema.

Exemple: grau de la relació ALUMNE

Partint de l'exemple, el grau de la relació és 7, ja que té 7 atributs.

Tot esquema de relació ha de tenir un atribut o conjunt d'atributs que formin la clau primària que s'acostuma a subratllar.

Relació

Una **relació** (o exemplar, instància de relació) **r** de l'esquema de relació **$R(A_1,A_2,...,A_n)$** és un conjunt de **m** tuples **$r=\{t_1,t_2,...,t_m\}$** on cada tupla **t** és una llista ordenada de ***n*** valors **$t=<v_1,v_2,...,v_n>$** i on cada valor **v_i** (amb $1<=i<=n$) és un valor del domini $dom(A_i)$ o bé un valor especial **NUL**. De forma més col·loquial ***és el conjunt de tots els valors d'una relació en un moment donat***.

Com a característiques d'aquestes relacions (taules) tenim:

- L'esquema d'una relació no acostuma a canviar gairebé mai, mentre que les relacions canvien tot sovint transformant-se en una altra relació.
- Les tuples en una relació no tenen cap ordre establert.
- Cada valor en una tupla és un valor simple (no divisible). El model relacional no permet atributs compostos ni multivaluats. Pel fet de no tenir atributs multivaluats ni compostos, direm que les relacions es troben en **primera forma normal**. Els atributs multivaluats s'han de representar amb relacions (taules) individuals i dels atributs compostos solament es representaran els atributs simples que els formen.
- Algunes relacions (taules) del model relacional representen els tipus d'entitats del model E-R. Exemple: la relació ALUMNE.
- Altres relacions (taules) representen les interrelacions del model E-R. Exemple: la interrelació *es matriculen* entre ALUMNE i CRÈDIT (N:M)

L'esquema d'una relació (taula) es pot interpretar com una declaració del tipus d'una entitat. Per exemple la taula ALUMNES té els atributs dni, nom, cognom1, etc. Cada tupla de la relació es pot interpretar com un fet o un exemplar d'aquesta declaració.

La **cardinalitat** d'una relació és el nombre de tuples que té.

Esquema d'una base de dades

Durant el disseny lògic d'un sistema de informació, s'obtenen els esquemes de cada una de les relacions que formen la base de dades, és el que es diu l'**esquema de la base de dades**.

Un **esquema d'una base de dades** relacional està format per un conjunt d'esquemes de taula $EBD=\{R_1,R_2,...,R_n\}$ més les possibles restriccions d'integritat que veurem en el següent apartat.

Exemplar d'una base de dades

Un **exemplar d'una base de dades** relacional, és un conjunt d'exemplars de relacions $BD=\{r_1,r_2,...,r_n\}$ tal que cada r_i és un exemplar de R_i i les relacions r_i satisfan les restriccions d'integritat especificades.

Resum

Definicions:

- Relació: És una taula, amb columnes i files.
- Atribut: nom d'una columna d'una relació.
- Tupla: fila de una relació.
- Grau d'una relació: nombre d'atributs que conté.
- Cardinalitat d'una relació: nombre de tuples que conté.
- Domini: conjunt de valors legals que pot prendre un atribut. Normalment s'assimila al tipus de dades.
- Base de Dades Relacional: Conjunt de relacions normalitzades.

Exemple:

OFICINA (numOf, carrer, poblacio, telefon, fax)

ATRIBUTS

TUPLES

numOf	carrer	poblacio	telefon	fax
111	C/ Major, 15	Blanes	972266789	972266559
222	Av. Sol, 5	Barcelona	972293445	932293675
333	Av. Del Mar, 16	Girona	972278796	972254796
...	...	...	...	...

- Grau: 5 (5 columnes)
- Domini: (valors vàlids)
  - numOf: 3 dígits (enters)
  - carrer/poblacio: 30 caràcters
  - telefon/fax: 9 dígits enters
- Cardinalitat: 4 (4 files)

Exemples de dominis:

- Números telefònics: Conjunt de 9 digits
- Noms: Conjunt de noms de persones (lletres)
- Promig de notes: Número (amb decimals) de 0 a 10.
- Edat d'empleat: Número (sense decimals) entre 15 i 80.
- Codis de departaments: INF, CAT, HIST (llista de valors)

Propietats de les relacions(taules):

- No es pot repetir:
  - el nom de les relacions.
  - el nom dels atributs dins una relació.
  - les tuples: cada tupla és diferent a les altres!.
- L'ordre de les tuples i dels atributs no és rellevant.
- Els valors dels atributs són atòmics: a cada tupla, cada atribut pren un únic valor.
- Cap atribut que formi part de la clau primària pot prendre valors nuls.

1.1.3. Clau candidata, clau primària i clau alternativa

Tota la informació que conté una base de dades s'ha de poder identificar d'alguna manera. En el cas de les de les bases de dades relacionals, per identificar les dades que la base de dades conté, es poden utilitzar les claus candidates de les relacions. A continuació es defineix el concepte de ***clau candidata***, ***clau primària*** i ***clau alternativa*** d'una relació.

Per poder definir aquests nous conceptes, primer cal definir el concepte de ***superclau***.

Una **superclau d'una relació que té per esquema** $R=(A_1,A_2,...,A_n)$ és un subconjunt dels atributs de l'esquema tal que no pot haver-hi dues tuples que tinguin la mateixa combinació de valors pels atributs d'aquest subconjunt.

Una superclau, per tant, permet identificar totes les tuples que conté la relació.

Exemple: superclaus de la relació ALUMNE

En l'esquema de relació *ALUMNE*(codi_alumne, *DNI, nom, cognom1, cognom2, telefon, c_postal, poblacio)* algunes de les superclaus de la relació serien els següents subconjunts d'atributs: *{DNI,nom,cognom1,cognom2,telefon}*, *{DNI,cognom1}*, *{DNI}*,*{codi_alumne,DNI,nom}*,*{codi_alumne}* etc.

Clau candidata

Una **clau candidata d'una relació** és una superclau *C* de la relació si l'eliminació de qualsevol dels seus atributs (els de la superclau *C*) dóna un conjunt d'atributs que no és superclau de la relació.

Una clau candidata permet identificar qualsevol tupla de la relació, de tal manera que no sobri cap atribut per poder identificar cada tupla.

Exemple: claus candidates d'*ALUMNE*

En l'esquema de relació *ALUMNE*(*DNI, nom, cognom1, cognom2, telefon, c_postal, poblacio)* només hi ha tres claus candidates: {*DNI*}, {codi_alumne}, {DNI, codi_alumne}.

Clau primària

Normalment, una de les claus candidates d'una relació s'escull com a clau primària de la relació. La **clau primària** és la clau candidata els valors de la qual s'utilitzaran per identificar les tuples de la relació.

El dissenyador de la base de dades és qui escull la clau primària entre les claus candidates.

Els atributs que formin part de la clau primària els marcarem en l'esquema subratllant-los. Així doncs, $R=(\underline{A_1},\underline{A_2},...,\underline{A_j},...A_n)$ indica que els atributs $A_1,A_2,...,A_j$ formen la clau primària de R.

Exemple: escollir la clau primària d'ALUMNE

En l'esquema de relació *ALUMNE*(*DNI, nom, cognom1, cognom2, telefon, c_postal, població, codi_alumne)* hi ha tres claus candidates: {*DNI*}, {codi_alumne}, {DNI, codi_alumne}. En aquest cas, el dissenyador ha escollit la clau candidata {DNI} com a clau primària subratllant l'atribut corresponent dins l'esquema de relació ***ALUMNE(<u>DNI</u>, nom, cognom1, cognom2, telefon, c_postal, població, codi_alumne).***

Pot passar que una clau candidata o una clau primària tingui més d'un atribut.

Exemple: clau primària de la relació AULA

En la relació de l'esquema AULA(*<u>edifici,numero</u>,superfície, num_finestres*), la clau primària està formada pels atributs *edifici* i *numero*. En aquest cas, pot passar que dos o més aules diferents estiguin al mateix edifici, o bé que tinguin el mateix número, però mai passarà que tinguin la mateixa combinació de valors per *edifici* i *numero*.

Clau alternativa

Les claus candidates no escollides com a primària s'anomenen **claus alternatives**.

1.2. Restriccions del model relacional

En els esquemes de bases de dades relacionals es poden especificar diferents tipus de restriccions (*"normes que cal complir"*). De vegades es parla de *regles d'integritat* en comptes de restriccions.

1.2.1. Restriccions de domini

Les restriccions de domini especifiquen que el valor de cada atribut A_i d'un esquema de relació R ha de ser un valor simple i del domini de A.

Exemple:

Si en la relació *ALUMNE*(*DNI, nom, cognom1, cognom2, telefon, c_postal, poblacio)* hem declarat que dom(*telefon*)=enter, llavors no podem inserir, per exemple, cap alumne que tingui per *telefon* els valors "+34" o "Pere", ja que no són enters!

1.2.2. Restriccions de clau

Totes les tuples d'una relació (taula) han de ser diferents, no podem tenir dues tuples amb els mateixos valors per tots els seus atributs. Ja sabeu que com a mínim el valor de la clau primària serà diferent, per simple definició clau primària.

1.2.3. Restricció d'integritat d'entitat

La restricció d'integritat d'entitat estableix que cap valor de la clau primària pot ser nul (NULL).

Això és perquè la clau serveix per identificar les tuples d'una relació. Les restriccions de clau i d'integritat d'entitats, s'estableixen per cada relació a nivell individual.

1.2.4. Restricció d'integritat referencial

La restricció d'integritat referencial s'estableix entre dues relacions i serveix per mantenir la consistència entre tuples de dues relacions.

De forma informal, la restricció d'integritat referencial estableix que una tupla d'una relació (taula) que fa referència a una tupla d'una altra relació, en aquesta haurà d'existir la tupla referenciada. Com exemple, suposem que tenim tres taules

ALUMNE

dni	nom	cog1	cog2	telèfon	codi_postal	població
40123564	Pere	Pi	Vidal	972276789	17800	Olot
35357853	Josep	Coromina	Faig		17857	Sant Joan
77900700	Maria	Puig	Vidal	972265641	17800	Olot
.....	.....	.....	.....	.....	.....	.....

dni	codi_as	nota	any
40123564	C1		
40123564	C2		
40123564	C3		
40123564	C4		
35357853	C1		
35357853	C5		

MATRICULA

codi_credit	nom	hores	curs
C1	SIMM	180	1
C2	GXAL	180	1
C4	FP	180	1
C6	SGBD	210	1
C7	RAT	70	1
C5	DFSI	210	2

CRÈDIT

En l'exemple es pot observar com la relació (taula) MATRICULA fa referència (mitjançant l'atribut *DNI*) a la relació ALUMNE i a la relació CRÈDIT (mitjançant l'atribut *codi_as*). No tindria sentit que el DNI 40123564 referenciat a la relació MATRICULA no tingués la corresponent tupla a la relació ALUMNE, amb tota la informació referent a l'alumne. De la mateixa manera, no tindria sentit que en la relació CRÈDIT no existís el crèdit amb codi C4.

Direm que l'atribut *DNI* de la relació MATRICULA és una **clau forana** que fa referència a la relació ALUMNE. De la mateixa manera, direm que *codi_as* de la relació MATRICULA és una clau forana que fa referència a la relació CRÈDIT.

De manera més formal, s'introdueix a continuació el concepte de clau forana (o clau externa).

Un atribut o conjunt d'atributs de l'esquema de relació R_1 és **una clau forana** de R_1, si es compleix que:

- L'atribut (o atributs) que formen la clau forana tenen el mateix domini que la clau primària CP d'un altre esquema de relació R_2. Es diu que la clau forana fa referència a la relació R_2.
- Un valor de la clau forana en una tupla ha d'existir com a valor de la clau principal en alguna tupla de R_2 o bé tenir el valor *NUL*. En el primer cas tenim que $t_i[CF]=t_j[CP]$ i direm que la tupla t_i de la relació R_1 fa referència a la tupla t_j de la relació R_2.

Aquestes restriccions que defineixen una clau forana especifiquen una **restricció d'integritat referencial** entre dos esquemes de relació R_1 i R_2. En una base de dades amb moltes relacions, normalment hi haurà moltes restriccions d'integritat referencial.

Les restriccions d'integritat referencial s'especifiquen per l'esquema d'una base de dades i l'han de complir tots els exemplars d'aquest esquema.

1.3. Operacions del model relacional

Les operacions del model relacional han de permetre manipular dades emmagatzemades en una base de dades relacional i, per tant, estructurades en forma de relacions. La manipulació de dades inclou bàsicament dos aspectes: l'actualització i la consulta.

1.3.1. Actualització de dades

L'**actualització de les dades** consisteix en fer que els canvis que es produeixen a la realitat quedin reflectits en les relacions de la base de dades **sense incomplir cap restricció** (de domini, de clau, integritat d'entitat, integritat referencial).

Exemple:

Una base de dades conté informació d'un institut i es matricula un nou alumne. En aquest cas cal reflectits aquest canvi afegint les dades del nou alumne a la base de dades.

Hi ha tres operacions bàsiques d'actualització:

a) **Inserir**, per afegir una o més tuples a una relació.

b) **Eliminar**, per eliminar una o més tuples d'una relació.

c) **Modificar**, per modificar els valors que tenen una o més tuples d'una relació per un o més dels seus atributs.

I. Inserir

Amb aquesta operació cal indicar els valors que tindrà cada atribut de la tupla que s'ha d'inserir a l'esquema de relació R. L'operació d'inserció pot *violar* qualsevol de les restriccions que hem indicat en l'apartat anterior:

- La restricció de domini es pot violar si donem a un atribut un valor que no sigui del seu domini.

  Exemple En un atribut dni s'intenti d'entrar "ABC"

- La restricció de clau pot violar-se al donar a l'atribut clau un valor ja existent en la relació.

  Exemple En un atribut dni s'intenti d'inserir el dni 111 i aquest ja existeix a la relació.

- La restricció d'integritat d'entitats pot violar-se quan s'intenta donar d'alta una tupla amb un valor nul en la seva clau primària.

  Exemple En un atribut dni s'intenti d'entrar NULL

- La restricció d'integritat referencial pot violar-se quan s'intenta donar d'alta una tupla `t`, si el valor de qualsevol clau forana de `t` fa referència a una tupla que no existeix.

  Exemple S'intenta d'afegir una tupla a la relació de matrícules un DNI que no existeix a la taula d'alumnes

Si una inserció viola una o més restriccions es poden fer dues coses:

1. Rebutjar la inserció. El SGBD hauria d'informar del motiu.
2. Intentar corregir el motiu pel qual rebutja l'operació. Per exemple, si la restricció violada és la de clau o la d'integritat d'entitats podria demanar un valor correcte per la clau primària o demanar que s'entrés la tupla adient per la taula referenciada.

II. Eliminar

Aquesta operació només pot violar la integritat referencial en cas que altres tuples de la base de dades facin referència a la tupla que volem eliminar. Al violar-se la integritat referencial, hi ha tres opcions:

1. Rebutjar l'operació
2. Tractar de propagar l'eliminació, eliminant les tuples que fan referència a la tupla que es vol eliminar (**eliminació en cascada**).
3. Modificar els valors de la clau forana que violarien la restricció. Aquests valors es podrien posar a *nul* o demanar a l'usuari que canviés el seu valor.

Exemple: En l'exemple anterior volem eliminar l'alumne amb DNI 35357853 alumne de la taula d'ALUMNES i aquest també està a la taula MATRÍCULES.

ALUMNE

DNI	Nom	Cog1	Cog2	Telèfon	CP	Població
~~40123564~~	Pere	Pi	Vidal	972276789	17800	Olot
35357853	Josep	Coromina	Faig		17857	Sant Joan
77900700	Maria	Puig	Vidal	972265641	17800	Olot
…..	…..	…..	…..	…..	…..	……

Dni		Nota	Any
~~40123564~~	~~C1~~		
40123564	C2		
40123564	C3		
40123564	C4		
35357853	C1		
35357853	C5		

MATRICULA

codi_credit	Nom_cr	Hores	curs
C1	SIMM	180	1
C2	GXAL	180	1
C4	FP	180	1
C6	SGBD	210	1
C7	RAT	70	1
C5	DFSI	210	2

CRÈDIT

III. Modificar

Algunes modificacions podrien violar alguna/es de les restriccions.

Les modificacions es poden classificar com:

- Modificacions d'un atribut no clau (ni primària, ni forana). No sol portar problemes i només cal vigilar que el nou valor estigui en el domini d'aquest atribut.
- Modificacions del valor de la clau primària. És equivalent a eliminar la tupla vella i a continuació fer una inserció amb el nou valor de la clau primària. Tot el que s'ha dit per eliminacions i insercions és aplicable en aquest apartat. Si la tupla `t` modificada té tuples d'altres relacions que la referencien, llavors es podria propagar la modificació (**modificació en cascada**), modificant el valor de les claus foranes en aquestes altres relacions amb el nou valor de la clau primària de la tupla `t`.
- Modificacions del valor d'una clau forana. Cal que el SGBD s'asseguri de que el nou valor faci referència a una tupla existent.

1.3.2. Consulta de dades

La **consulta de les dades** consisteix en l'obtenció de dades deduïbles (que no es poden veure de forma directa) a partir de les relacions que conté la base de dades.

Exemple:

Una base de dades conté informació d'un institut. Si es vol saber el nom i cognom de tots els alumnes que estan matriculats del crèdit *SGBD*.

La consulta necessària per poder obtenir les dades que es demanen, potser necessita cercar les dades en una o més relacions (taules) de la base de dades.

Segons la forma com s'especifiquen les consultes, es poden classificar els llenguatges relacionals en dos tipus:

1) **Llenguatges basats en l'àlgebra relacional**. L'àlgebra relacional s'inspira en la teoria de conjunts. Si es vol especificar una consulta, cal seguir una o més passes que serveixen per anar construint, mitjançant operacions de l'àlgebra relacional, una nova relació que contingui les dades que responen a la consulta a partir de les relacions de la base de dades. Els llenguatges basats amb l'àlgebra relacional són **llenguatges procedimentals**, ja que les passes que formen la consulta descriuen un procediment.

2) **Llenguatges basats en el càlcul relacional**. El càlcul relacional té el seu fonament teòric en el càlcul de predicats de la lògica matemàtica. Proporciona una notació que permet formular la definició de la relació on estan les dades que es volen obtenir amb la consulta, indicant les relacions que les contenen. Aquesta definició no descriu un procediment; per tant, es diu que els llenguatges basats en el càlcul relacional són **llenguatges declaratius** (no procedimentals).

El **llenguatge SQL**, en les instruccions de consulta, combina construccions de l'àlgebra relacional i del càlcul relacional, tot i que la majoria són del càlcul relacional. És per això que es diu que és un llenguatge no procedimental.

L'estudi de l'àlgebra relacional (s'explica més endavant) té un interès especial, ja que ajuda a entendre quines possibilitats de consulta ha de proporcionar un llenguatge relacional, facilita la comprensió d'algunes de les construccions del llenguatge SQL i també serveix de base pel tractament de les consultes que fan els SGBD internament.

A 2. Transformació ER a relacional

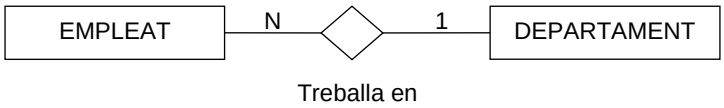
2.1. Obtenció de l'esquema de base de dades relacional

A l'hora de dissenyar una base de dades, normalment en primer lloc és fa el disseny conceptual utilitzant el model entitat-relació. Una vegada fet aquest disseny, ens interessarà obtenir l'esquema de la base de dades en el model relacional.

Per obtenir l'esquema lògic (esquema de base de dades) simplement caldrà seguir les passes que es detallen a continuació:

- **Pas 1**. **Entitats fortes**
  - Per cada Tipus d'Entitats forta (TE) de l'esquema entitat-relació es crea una relació (taula) R que contingui tots els atributs simples del TE. Escollim l'atribut o un conjunt d'atributs principals com a clau primària de R. Els atributs no NULLS s'han de marcar amb el caràcter estarístc (*)

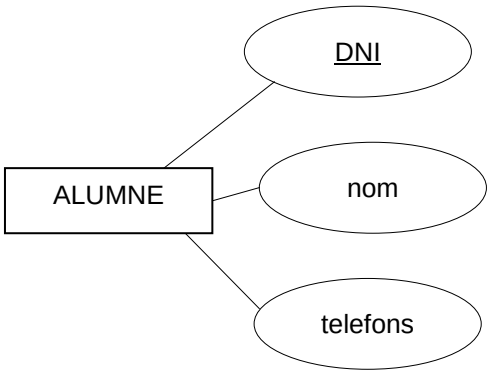
    Exemple:



EMPLEAT(<u>DNI</u>,nom,cog1,cog2,...)

DEPARTAMENT(<u>codiDep</u>,nomDep,telefon,...)

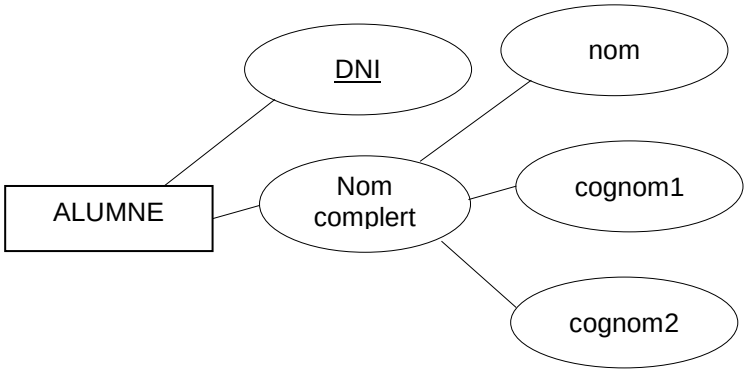
Atributs multivaluats: En principi en cap diagrama ER hi haurien d'haver atributs multivaluats., però si és el cas donen lloc a una nova relació (taula) on la clau primària és l'identificador principal de la seva entitat més el nom de l'atribut simple.



ALUMNE (<u>DNI</u>,nom)

TELEFON (<u>DNI, telefon</u>)

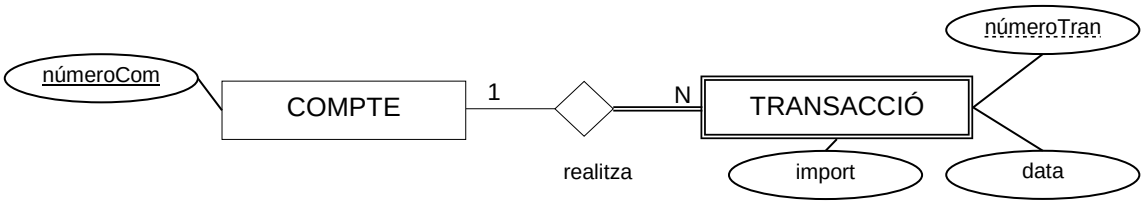
Atributs compostos: Els atributs compostos es transformen afegint cadascun dels atributs simples que el composen.



ALUMNE (dni, nom, cognom1, cognom2)

- **Pas 2**. **Entitats febles**
  - Per cada Tipus d'Entitats feble (TEF) de l'esquema entitat-relació es crea una relació (taula) R que contingui tots els atributs simples del TEF. S'afegeix com a atribut/s (com a clau forana), la clau primària del tipus d'entitat forta de qui depèn (TE). La clau primària d'aquesta relació serà la combinació de la clau primària de l'entitat forta i la clau parcial del TEF.

    Exemple:



TRANSACCIO(numeroCom,numeroTran,data,import...)

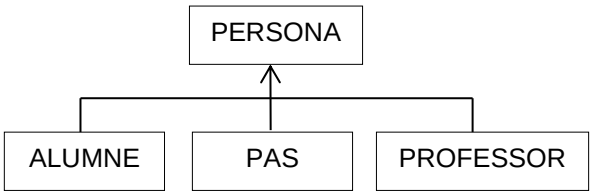
Indica clau forana

- **Pas 3**. **Generalitzacions i especialitzacions**

  - Per cada tipus d'entitat superclasse de l'esquema entitat-relació estès (EER) es crea una nova relació (taula) `R` que contingui tots els seus atributs de la superclasse (els comuns a totes les entitats subclasse). Per cada tipus d'entitat subclasse de l'esquema entitat-relació estès (EER) es crea una nova relació (taula) R_1, R_2, etc., que contingui tots els seus atributs (els específics de cada subclasse) i s'hi afegeix com a clau forana, la clau primària de la superclasse que també actuarà com a clau primària de les noves relacions (taules) R_1, R_2, etc.

  - És indiferent si les sublasses són dijuntes o solapades amb participació total o parcial.

    Exemple:



PERSONA(DNI,nom,cog1,cog2,…)

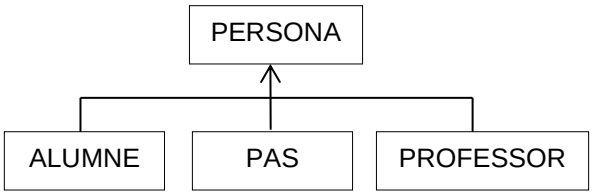
ALUMNE(DNI,codiEstudiant,…)

PROFESSOR(DNI,especialitat,…)

PAS(DNI, anyInici,.....)

A vegades es pot pensar de crear una sola entitat utilitzant la superclasse i afegint tots els atributs de les subclasses. Aquesta solució no és acceptada en cassos generals aportaria molts de valors NULLs en els camps específics de les subclasses. Només cal tenir-la en compte en casos molt concrets, com per exemple, en temes d'optimització.

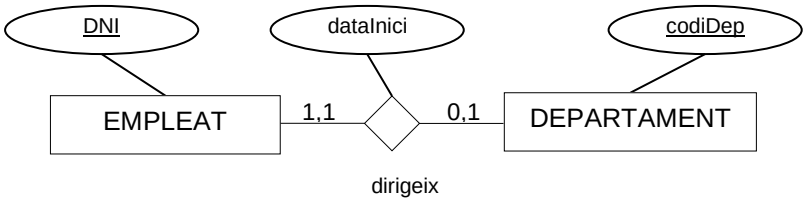
Exemple:



PERSONA(DNI,nom,cog1,cog2,codiEstudiant,especialitat,anyInici…)

- **Pas 4. Interrelacions 1:1**
  - Per cada interrelació 1:1 de l'esquema entitat-relació, identifiquem les relacions (taules) `R` i `T` que corresponen als tipus d'entitats que intervenen en la interrelació 1:1. Escollim una de les relacions, suposem `R`, i hi afegim com a clau forana a `R`, la clau primària de `T`.
  - Si un dels dos tipus d'entitats té participació total, escollirem aquesta en el paper de `R`. Afegirem també els possibles atributs de l'interrelació 1:1, com atributs de `R`. Convé propagar l'entitat amb cardinalitat (1,1).

Exemple:

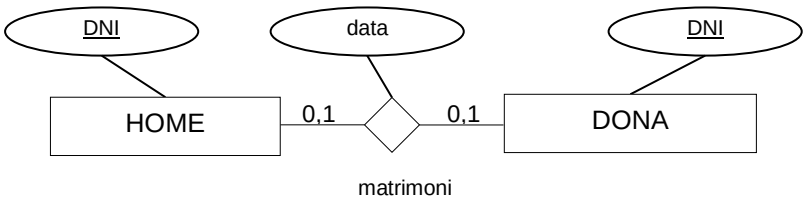


EMPLEAT(DNI,Nom,Cog1,Cog2,...)

DEPARTAMENT(codiDep,Nom,Telefon,...,DNIDirector,dataInici)

En aquelles interrelacions 1:1 si la cardinalitat de totes dues entitats és (0,1) es convenient transformar la relació en una nova taula.

Exemple:



HOME(DNI,Nom,Cog1,Cog2,...)

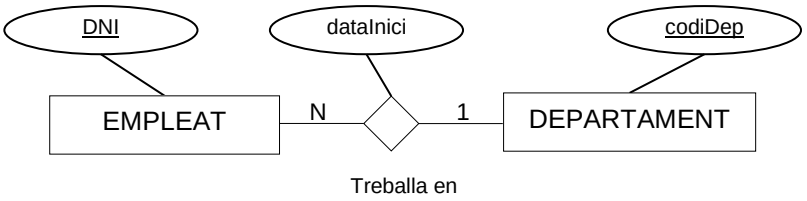
DONA(DNI,Nom,Cog1,Cog2,...)

MATRIMONI(DNIHome, DNIDona, data): Afegir restricció única a DNIDona i NOT NULL.

- **Pas 5**. **Interrelacions 1:N**

  - Per cada interrelació 1:N no feble de l'esquema entitat-relació, identifiquem la relació (taula) `R` que representa el tipus d'entitat del costat N de l'interrelació. Afegim com a clau forana a `R` la clau primària de la relació `T` que representa el costat 1 de l'interrelació. Afegirem també els possibles atributs de l'interrelació 1:N, com atributs de `R`.

    Exemple:

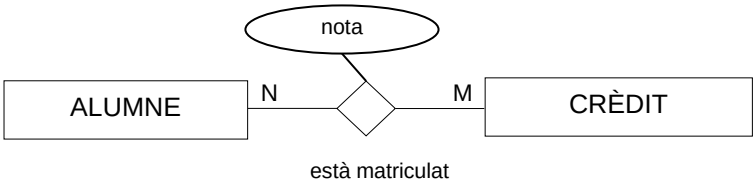


EMPLEAT(DNI,Nom,Cog1,Cog2,...,codiDep,dataInici)

- **Pas 6**. **Interrelacions N:M**

  - Per cada interrelació N:M de l'esquema entitat-relació, es crea una nova relació (taula) `R` per representar aquesta interrelació. Afegim com a atributs clau forana a `R`, les claus primàries de les dues relacions (taules) que representen els tipus d'entitats que participen en la interrelació. La clau primària de `R` serà la combinació d'aquestes dues claus. Afegirem també els possibles atributs de l'interrelació N:M, com atributs de `R`. En certs casos pot ser interessant tractar les interrelacions 1:N com si fossin N:M, concretament és útil quan hi ha pocs exemplars en la interrelació i amb la finalitat d'evitar molts valors nuls en les claus foranes. En aquest cas la clau primària està formada solament per la clau primària del tipus d'entitat del costat n.

    Exemple:



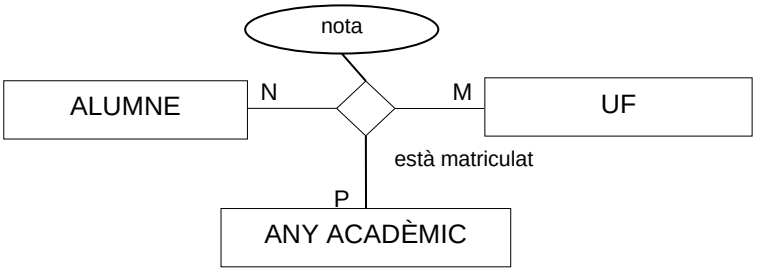
ALUMNE(DNI,nom,cog1,cog2,…)

CREDIT(codi,nom,curs,hores,…)

MATRICULA(DNIAlumne,codiCredit,nota)

- **Pas 7**. **Interrelacions N-àries (N>2)**
  - Per cada interrelació **N-ària** (amb N>2) es crea una nova relació (taula) `R` que representa aquesta interrelació. S'afegeixen coma claus foranes a `R`, les claus primàries de cada una de les relacions (taules) que representen els tipus d'entitat de la interrelació. La clau primària de `R` serà la combinació de totes elles. Afegirem també els possibles atributs de l'interrelació N-ària, com atributs de `R`.

    Exemple:



ALUMNE(DNI,nom,cog1,cog2,…)

UF(codi,nom,curs,hores,…)

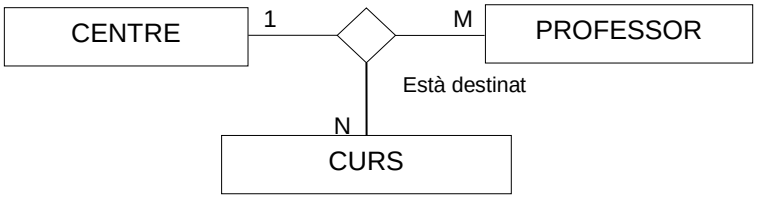
ANY_ACADÈMIC(any,descripcio,dataIni,dataFi,…)

MATRICULA(DNIAlumne,codiUF,anyAcademic,nota)

  - Per les interrelacions ternàries **N:M:1**: es crea una nova relació (taula) `R` que representa aquesta interrelació. S'afegeixen com a claus foranes a `R`, les claus primàries de cada una de les relacions (taules) que representen els tipus d'entitat de la interrelació. **La clau primària de R serà la combinació de les claus primàries de les entitats relacionades que**

tenen com a cardinalitat N. Afegirem també els possibles atributs de la interrelació, com atributs de R.

Exemple:



PROFESSOR(DNI, nom, cog1, cog2,…)

CURS(codi_curs, nom, hores,…)

CENTRE(codi_centre, nom, adreça,…)

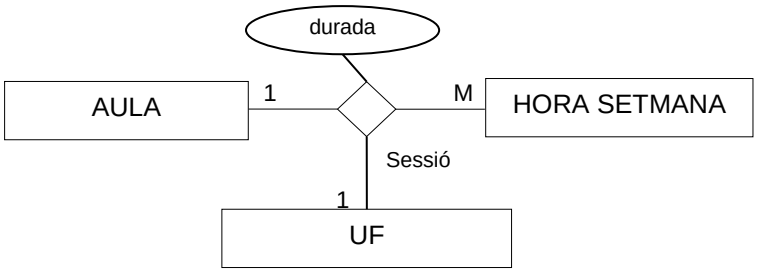
DESTÍ(DNIProfessor,codiCurs, codiCentre)

DNI Prof.	CodiCurs	CodiCentre	
111	ASIX	INSTITUT GIRONA	
111	DAM	INSTITUT GIRONA	
111	DAW	INSTITUT BCN	
111	DAM	INSTITUT BCN	**No possible**
...	...	...	

En aquest exemple un professor només pot està destinat a un centre per un mateix curs.

- Per les interrelacions ternàries **N:1:1**: es crea una nova relació (taula) R que representa aquesta interrelació. S'afegeixen com a claus foranes a R, les claus primàries de cada una de les relacions (taules) que representen els tipus d'entitat de la interrelació. **La clau primària de R serà la combinació de la clau primària de l'entitat que té com a cardinalitat N més qualsevol de les dues entitats que tenen com a cardinalitat 1**. Afegirem també els possibles atributs de la interrelació, com atributs de R.

Exemple:



AULA(codi, nom,…)

UF(codi, nom, ...)

HORA SETMANA(codi, número,…)

Opció A

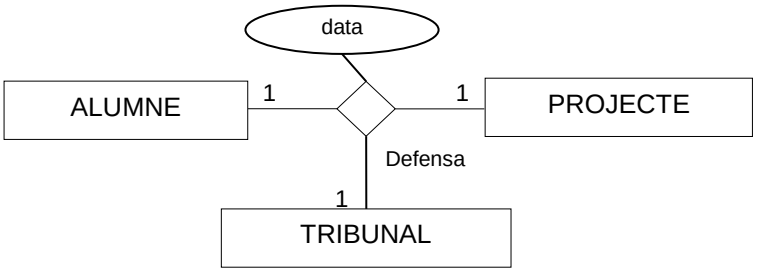
SESSIÓ(codiHora, codiAula, codiUF)

Opció B

SESSIÓ(codiHora, codiUF, codiAula)

- Per les interrelacions ternàries **1:1:1**: es crea una nova relació (taula) `R` que representa aquesta interrelació. S'afegeixen com a claus foranes a `R`, les claus primàries de cada una de les relacions (taules) que representen els tipus d'entitat de la interrelació. **La clau primària de R serà la combinació de dues claus primàries de dues entitats qualsevols**. Afegirem també els possibles atributs de la interrelació, com atributs de `R`.

Exemple:



ALUMNE(DNI, nom,…)

PROJECTE(codi, nom, ...)

TRIBUNAL(codi, nom_president ,…)

Opció A

DEFENSA(codiTribunal, DNIAlumne, codiProjecte, data)

Opció B

DEFENSA(codiTribunal, codiProjecte, DNIAlumne, data)

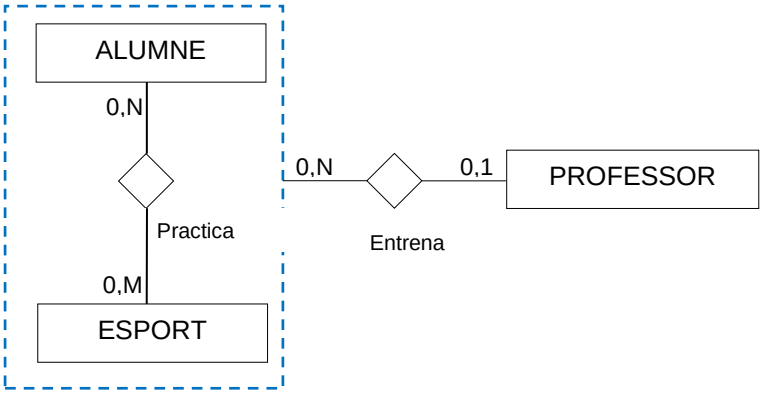
Opció C

DEFENSA(DNIAlumne, codiProjecte, codiTribunal, data)

Nota: ***Moltes vegades el que es fa és crear una nova relació (taula) R que representa la interrelació afegint com a claus foranes a R, les claus primàries de cada una de les relacions (taules) que representen el tipus d'entitat de la interrelació. La clau primària de R serà la combinació de totes elles. Afegirem també els possibles atributs de l'interrelació N-ària, com atributs de R.***

- **Pas 8. Entitats associatives / agregacions**
  - Les entitats associatives es basen en una interrelació entre entitats. La traducció d'aquesta interrelació a un model relacional equival a la traducció de l'entitat associativa.
  - Primer desenvolupem l'associativa i llavors les interrelacions entre aquesta.

Exemple:



ALUMNE(DNI, Nom, Cognoms)

ESPORT(Codi, Descripcio)

PROFESSOR(DNI, Nom, Cognoms

PRACTICA(DNIAlumne, CodiEsport, DNIProfessor)

Fixem-nos com la relació PRACTICA, derivada de l'entitat associativa, incorpora una clau forana que fa referència a la relació PROFESSOR, ja que l'entitat associativa originària és al costat N d'una interrelació

binària amb l'entitat PROFESSOR. Tal i com hem fet en el pas 5 de tractar les relacions 1:N

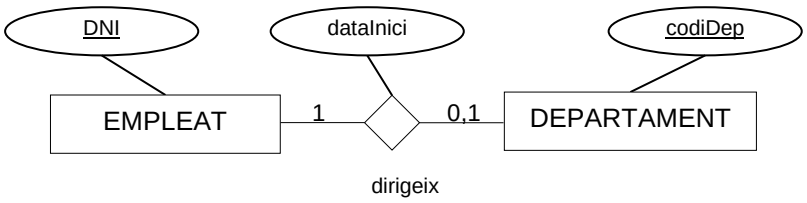
- **Pas 9**. **Incloure restriccions de domini i d'integritat.**
  - Cal definir per cadascuna de les relacions (taules) `R` del nostre model relacional les restriccions de domini de cadascun dels camps.
  - Les restriccions de domini i obligatorietat les especificarem un cop hàgim especificat la relació (taula) `R`.
  - Exemple:

    ALUMNE(DNI, Nom*, Cognoms*, Telèfon)

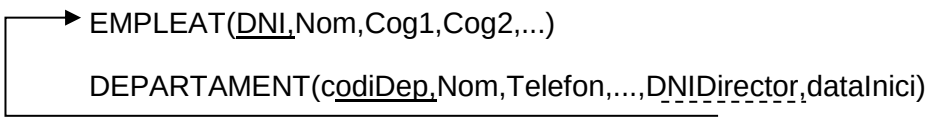
    Restriccions:

    - DNI: 8dígits + 1 lletra
    - Nom: Text 20 caràcters
    - Cognoms: Text 50 caràcters
    - Telèfon: Text 10 caràcters numèrics

  - És a dir clarificar les claus foranes de cada relació (taula) a quina clau primària de quina relació pertany hi ha dos mètodes utilitzats:



1.- Modelitzar-les amb un graf dirigit (No l'utilitzarem)



2.- Definir al final de la relació a quines relacions (taules) fan referència les claus foranes de la relació que estic tractant.

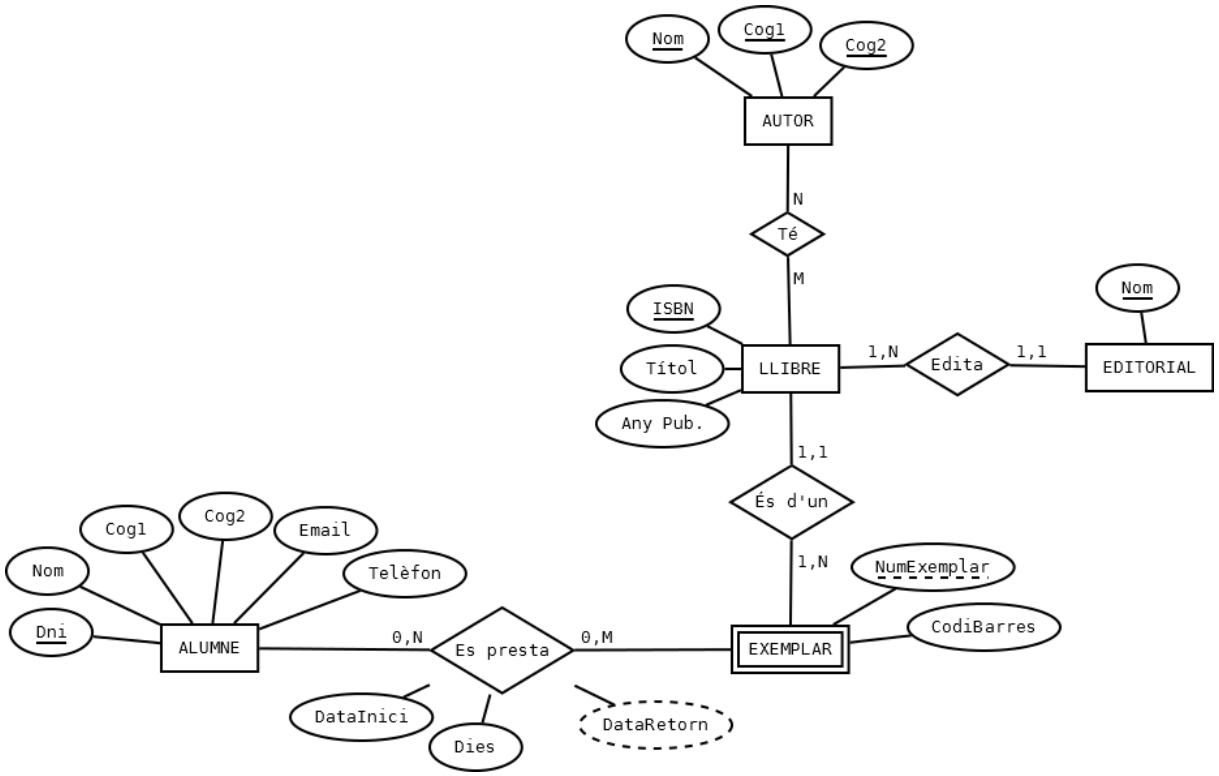
EMPLEAT(DNI,Nom,Cog1,Cog2,...)

DEPARTAMENT(codiDep,Nom,Telefon,...,DNIDirector,dataInici)
ON {DNIDirector} REFERENCIA EMPLEAT {DNI}

2.2. Exemples d'esquemes de base de dades relacionals

2.2.1. BD Biblioteca

Donat aquest esquema d'Entitat Relació obtén l'esquema lògic per una BD Relacional.



ALUMNE (DNI, Nom*, Cog1*,Cog2, Email, Telèfon)

Dominis/tipus

- DNI: 10 caràcters alfanumèrics format per 8 dígits i 2 lletres (al principi o al final)
  - Exemples: 45784587L, M4587898S
- Nom: 15 caràcters
- Cog1: 25 caràcters
- Cog2: 25 caràcters
- Email: 50 caràcters
- Telèfon: 20 caràcters

Restriccions:

- O bé el Email o el telèfon no poden ser NULLS.

AUTOR (Nom, Cog1, Cog2)

Dominis/tipus

- Nom: 15 caràcters
- Cog1: 25 caràcters
- Cog2: 25 caràcters

Restriccions:

- Cap

EDITORIAL (Nom)

Dominis/tipus

- Nom: 25 caràcters

Restriccions:

- Cap

LLIBRE (ISBN, Títol*, Any Publicació)

Dominis/tipus

- ISBN: 13 dígits
- Títol: 50 caràcters
- Any Publicació: 4 dígits que expressen l'any (1999,2000,...)

Restriccions:

- Cap

EXEMPLAR (ISBN, Num_Exemplar, CodiBarres*)

ON {ISBN } REFERENCIA LLIBRE{ISBN}

Dominis/tipus

- ISBN: 13 dígits
- Num Exemplar: 3 dígits
- CodiBarres: cadena de 128 caràcters numèrics.

Restriccions:

- No hi poden haver Codi Barres repetits UNIQUE

AUTOR_LLIBRE(AutorNom, AutorCog1, AutorCog2, ISBN)

ON {AutorNom,AutorCog1,AutorCog2 } REFERENCIA AUTOR {Nom,Cog1,Cog2}

ON {ISBN } REFERENCIA LLIBRE{ISBN }

Dominis/tipus

- Nom: 15 caràcters
- Cog1: 25 caràcters
- Cog2: 25 caràcters
- ISBN: 13 dígits

PRÈSTEC(DniAlumne, ISBN, NumExemplar , DataInici*, Dies, DataRetorn)

ON {DniAlumne } REFERENCIA ALUMNE{DNI}

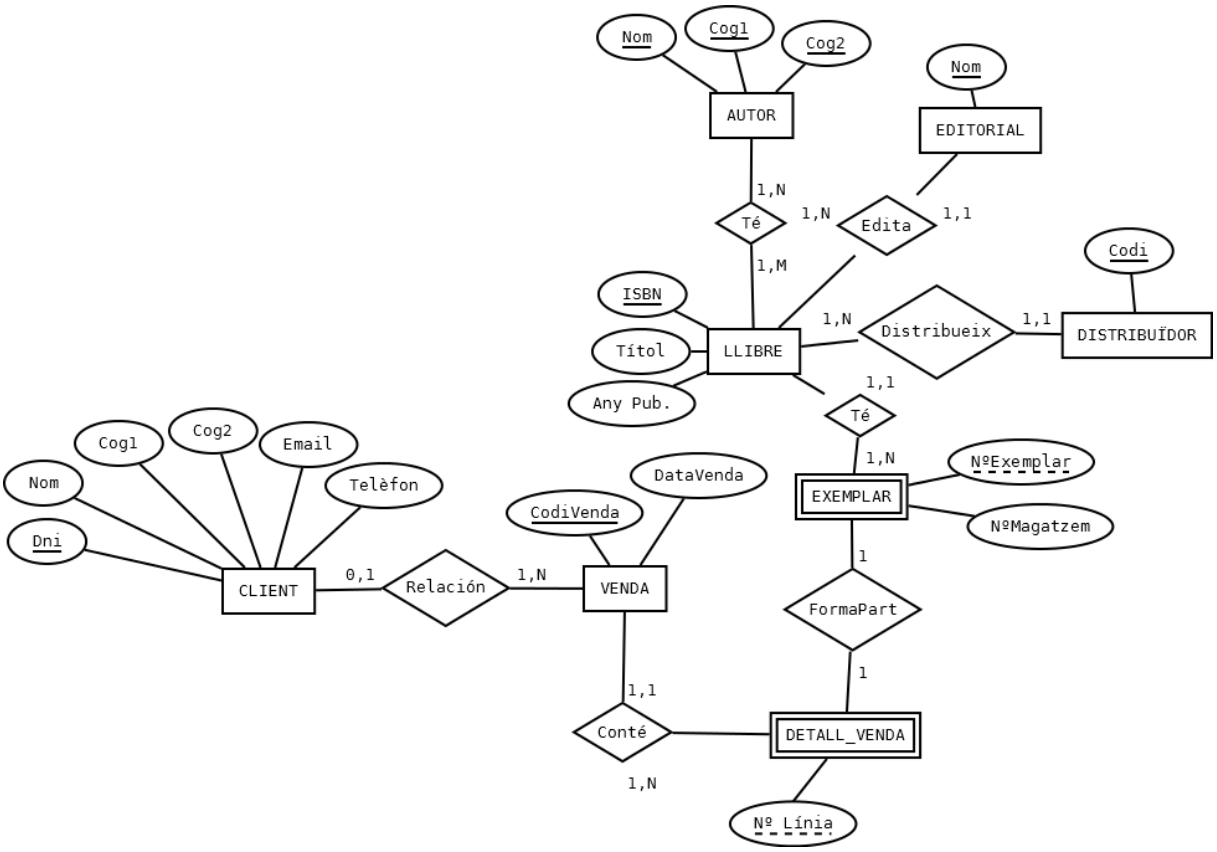
ON {ISBN, NumExemplar } REFERENCIA EXEMPLAR{ISBN, NumExemplar}

Dominis/tipus

- DniAlumne: 10 caràcters alfanumèrics
- ISBN: 13 dígits
- Num Exemplar: 3 dígits
- DataInici: Data, sense hores (dd/mm/YYYY)
- Dies: Enter de 3 dígits
- DataRetorn: Data, sense hores (dd/mm/YYYY).

Restriccions:

- L'atribut DataRetorn és calculat a partir de la DataInici i els Dies. S'obté sumant el número de dies a la DataInici.

2.2.2. BD Llibreria

Donat aquest esquema d'Entitat Relació obtén l'esquema lògic per una BD Relacional.