 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

2253 CFGS Desenvolupament d'Aplicacions Web

Mòdul 2 – Bases de dades

UF1 – Introducció a les bases de dades

NF3 – Model relacional i normalització

APUNTS


Índex

1. Model Relacional.....pag 2
2. Traducció del Model ER al Model Relacional.....pag 17
3. Normalització.....pag 27

Breu introducció

En aquest apartat, s'aprofundeix en un model de dades concret: el model relacional, que actualment té una gran rellevància. Els seus conceptes fonamentals estan ben assentats i, a més, els sistemes de gestió de bases de dades relacionals són els més estesos en la seva utilització pràctica.

Els principis del model de dades relacional van ser establerts per E.F. Codd els anys 1969-1970. De totes maneres, fins a la dècada dels vuitanta no es van començar a comercialitzar els primers SGBD relacionals amb rendiments acceptables. Val a dir, però, que els SGBD relacionals que es comercialitzen actualment encara no suporten tot el que estableix la teoria relacional fins al darrer detall.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

1. Model Relacional

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

- 1) L'estructura, que ha de permetre representar la informació que ens interessa del món real.
- 2) La manipulació, a la qual dóna suport mitjançant les operacions d'actualització i consulta de les dades.
- 3) La integritat, que és facilitada mitjançant l'establiment de regles d'integritat, és a dir, condicions que les dades han de complir.

El principal objectiu del model de dades relacional és facilitar que la base de dades sigui percebuda o vista per l'usuari com una estructura lògica que consisteix en un **conjunt de relacions (taules)**, i no com una estructura física.

Un objectiu addicional del model és aconseguir que aquesta estructura lògica amb què es percep la base de dades sigui simple i uniforme. Per tal de proporcionar simplicitat i uniformitat, tota la informació es representa d'una única manera: mitjançant valors explícits que contenen les relacions (no es fan servir conceptes com ara apuntadors entre relacions). Amb el mateix propòsit, tots els valors de dades es consideren atòmics, és a dir, no és possible descompondre'ls.

Cal precisar que un SGBD relacional, a nivell físic, pot emprar qualsevol estructura de dades per a implementar l'estructura lògica formada per les relacions. En particular, a nivell físic, el sistema pot fer servir apuntadors, índexs, etc. Aquesta implementació física queda, però, amagada a l'usuari.

Estructura de les dades


El **model relacional** proporciona una estructura de les dades per a representar la informació que ens interessa del món real que consisteix en un **conjunt de relacions (taules o representacions tabulars)**.

Cal precisar que:

- 1) Tota relació ha de tenir un nom que la identifiqui unívocament dins de la base de dades.
- 2) Cada **fila** està constituïda per una **tupla** de dades relacionades entre elles, anomenat també **registre**, que guarda les dades que ens interessa reflectir d'un objecte concret del món real.
- 3) En canvi, cada **columna** conté, en cada cel·la, dades d'un mateix tipus, i se la pot anomenar **atribut** o **camp**.
- 4) Cada **cel·la**, o intersecció entre fila i columna, pot emmagatzemar un únic **valor**.

La següent taula reflecteix l'estructuració tabular de la relació EMPLEAT, que conté les dades personals corresponents als empleats d'una empresa. Cada fila conté unes quantes dades que pertanyen a un mateix empleat.

La relació té un nom (EMPLEAT), com cadascuna de les columnes (DNI, nom, cognom i sou). Si aquests noms són prou significatius, permeten copsar de seguida el sentit que tenen els valors de

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

les dades emmagatzemades en la relació.

EMPLEAT			
DNI	nom	cognom	sou
11111111A	Anna	Sala	2000.00
22222222B	Àlex	Comas	2200.00
33333333C	David	Valls	180.00

Un **domini** consisteix en un conjunt finit de valors **atòmics** (indivisibles).

Els atributs només poden prendre els valors que estiguin inclosos dins del domini respectiu. Altrament no són valors vàlids, i un SGBD relacional no en pot permetre l'emmagatzematge.

Els dominis poden ser de dos tipus:

- 1) **Dominis predefinits**, que corresponen als tipus de dades que normalment són proporcionats pels llenguatges de bases de dades, com ara els enters, les cadenes de caràcters, els reals, etc.
- 2) **Dominis definits per l'usuari**, que poden ser més específics. Tota definició d'un domini ha de constar, com a mínim, del nom del domini i de la descripció dels valors que en formen part.

Exemples:


- 1) El domini de l'atribut sou serà els nombres reals (domini predefinit).
- 2) Si afegíssim l'atribut edat, el seu domini podria ser els enters entre 18 i 65 (domini definit per l'usuari).

Una relació es compon de l'**esquema** (o **intensió** de la relació) i de l'**extensió**.
 L'esquema de la relació consisteix en un nom de relació R i un conjunt d'atributs {A 1 , A 2 , ..., A n }.
 L'extensió de la relació consisteix en els valors de les dades emmagatzemades en tots els tuples que aquesta conté.

Si prenem com a exemple la taula EMPLEAT, l'esquema és el nom de la relació, EMPLEAT, i el conjunt d'atributs {DNI, nom, cognom, sou}.

L'extensió serà: <11111111A, Anna, Sala, 2000.00>, <22222222B, Àlex, Comas, 2200.00>, <33333333C, David, Valls, 1800.00>.

Convencions:

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

- 1) S'ha d'escriure el nom de la relació en majúscules i singular.
- 2) S'ha d'escriure el nom dels atributs en minúscules, sempre que no es tracti de sigles, ja que aleshores és més convenient deixar totes les lletres amb majúscules (com ara DNI). Per tal de fer els noms compostos més llegidors, es pot encapçalar cada paraula exceptuant la primera de les que formen el nom del camp amb una lletra majúscula (per exemple: dataNaixement, telefonParticular, etc.).

De vegades, els atributs de les relacions poden no contenir cap valor o, dit d'una altra manera, poden contenir valors **nuls**.

És possible que quan donem d'alta un empleat no se sap encara el sou que tindrà. Es podria considerar emmagatzemar un empleat sense sou, per tant l'atribut sou no tindria valor. El seu valor serà valor nul que significa sense valor.

El **grau** d'una relació depèn del nombre d'atributs que inclou el seu esquema.

La relació amb esquema EMPLEAT(DNI, nom, cognom, sou) és de grau 4, perquè té quatre atributs.

La **cardinalitat** d'una relació ve donada pel nombre de tuples que en formen l'extensió.

La cardinalitat de la relació EMPLEAT és 3, perquè la seva extensió conté de moment 3 empleats.

Claus candidates, claus alternatives i clau primària


Per tal de resultar útil, l'emmagatzematge de la informació ha de permetre la identificació de les dades. En l'àmbit de les bases de dades relacionals, els tuples de les relacions s'identifiquen mitjançant les anomenades **superclaus**.

Una **superclau** és un subconjunt dels atributs que formen l'esquema d'una relació tal que no és possible que hi hagi més d'una tupla en l'extensió respectiva, amb la mateixa combinació de valors en els atributs que formen part del subconjunt esmentat.

Però una superclau pot contenir atributs innecessaris, que no contribueixen a la identificació inequívoca de les diferents tuples. El que habitualment interessa és treballar amb superclaus **mínimes**, tals que cap subconjunt d'atributs de la superclau sigui capaç d'identificar les tuples de la relació. Els atributs de la superclau **no poden admetre valors nuls**, ja que no podrien garantir la identificació de les tuples.

D'altra banda, cal dir que en una mateixa relació pot passar que hi hagi més d'una superclau que permeti distingir les tuples unívocament entre elles.

S'anomenen **claus candidates** totes les superclaus d'una relació formades pels atributs o

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

conjunts d'atributs que permeten identificar les tuples que conté la seva extensió.

Però, a l'hora d'implementar una BD, entre totes les claus candidates de cada relació només se n'ha de triar una.

Quan parlem de **clau primària** ens referim a la clau que, finalment, el dissenyador lògic de la base de dades tria per distingir unívocament cada tupla d'una relació de la resta.

Aleshores, les claus candidates no triades com a clau primària resten presents en la relació.

Quan una relació ja té establerta una clau primària, la resta de claus presents en aquella, i que també podrien servir per identificar les diferents tuples de l'extensió respectiva, es coneixen com a **claus alternatives**.

Una forma de diferenciar els atributs que formen la clau primària de les relacions, dels altres atributs de l'esquema respectiu, és posar-los subratllats. Per aquest motiu, normalment es col·loquen junts i abans que la resta d'atributs, dins de l'esquema. Però només es tracta d'una qüestió d'elegància, ja que el model relacional no es basa ni en l'ordre dels atributs de l'esquema, ni tampoc en l'ordre dels tuples de l'extensió de la relació.

Considerem la relació EMPLEAT a la qual afegim més atributs

EMPLEAT (DNI, NSS, nom, cognom, sou)

Veurem fàcilment com els atributs DNI i NSS (número de la Seguretat Social) en ser personals i irrepetibles, ens podrien servir per identificar unívocament els empleats. Per tant, serien claus candidates.


Aleshores, el dissenyador de BD s'haurà de decidir per una clau candidata com a clau primària. Si, per exemple, tria DNI com a clau primària, les antigues claus candidates restants es passaran a considerar claus alternatives.

En aquest cas, doncs, l'esquema resultant haurà de reflectir quina és la clau primària de la relació, tot subratllant l'atribut DNI:

EMPLEAT (DNI, NSS, nom, cognom, sou)

si no es disposa d'un atribut que sigui capaç d'identificar les tuples de la relació per si sol, cal buscar un subconjunt d'atributs, tals que la combinació dels valors que adoptin no es pugui repetir. Si aquesta possibilitat no existeix, cal afegir a la relació un atribut addicional que faci d'identificador.

Per definició, el model relacional **no admet tuples repetides**, és a dir, no permet l'existència de tuples en una mateixa relació que tinguin els mateixos valors en cadascun dels atributs que constitueixen la clau primària.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Claus foranes

Fins ara hem estudiat les relacions de manera individual, però hem de tenir en compte que una base de dades relacional normalment conté més d'una relació, per tal de poder representar diversos tipus de fets que succeeixen al món real. Per exemple, podríem tenir una petita base de dades que contingués dues relacions: una que tingués per nom EMPLEAT, que emmagatzemaria dades dels empleats d'una empresa, i una altra amb el nom DESPATX, que emmagatzemaria les dades dels despatxos que té l'empresa.

Hem de considerar també que entre els diversos fets que es donen al món real hi sol haver lligams o vincles entre ells. Per exemple, els empleats que treballen per a una empresa poden estar lligats amb els despatxos de l'empresa perquè a cada empleat se li assigna un despatx concret per a treballar.

En el model relacional, per a reflectir aquests tipus de vincles, hi ha la possibilitat d'expressar connexions entre les diferents tuples de les relacions. Per exemple, en la base de dades anterior, que té les relacions EMPLEAT i DESPATX, pot ser necessari connectar tuples d'EMPLEAT amb tuples de DESPATX per a indicar quin despatx té assignat cada empleat.

De vegades, fins i tot pot ser necessari reflectir lligams entre tuples que pertanyen a una mateixa relació. Per exemple, en la mateixa base de dades anterior, pot ser necessari connectar determinades tuples d'EMPLEAT amb altres tuples d'EMPLEAT per a indicar, per a cada empleat, qui fa de cap.


El mecanisme que proporcionen les bases de dades relacionals per a connectar tuples són les **claus foranes** de les relacions. Les claus foranes permeten establir connexions entre les tuples de les relacions. Per a fer la connexió, una clau forana té un conjunt d'atributs d'una relació que referencien la clau primària d'una altra relació (o fins i tot de la mateixa relació).

Les diferents combinacions de valors dels atributs de tota clau forana han d'existir en la clau primària a què fan referència, o bé han de ser valors nuls. Altrament, les referències serien errònies i, per tant, les dades serien incorrectes.

Cal parar atenció en les característiques següents de les claus foranes:

- 1) Tota clau forana ha de tenir el mateix nombre d'atributs que la clau primària a la qual fa referència.
- 2) Entre els atributs de l'esquema d'una clau forana i els de la clau primària respectiva s'ha de poder establir una correspondència (concretament, una bijecció).
- 3) Els dominis dels atributs de tota clau forana han de coincidir amb els dominis dels atributs de la clau primària respectiva (o, com a mínim, cal que siguin compatibles dins d'un cert rang).

Una relació pot contenir més d'una clau forana, o bé no contenir-ne cap. I, en sentit invers, la clau primària d'una relació pot estar referenciada per una o més claus foranes, o bé pot no estar referenciada per cap.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Finalment, cal dir que es pot donar el cas que un mateix atribut formi part tant de la clau primària de la relació com d'alguna de les seves claus foranes.

A la descripció següent, la relació EMPLEAT (DNI, nom, cognom, sou, DNICap, edificidesp, numerodesp), té una clau forana formada pels atributs edificidesp i numerodesp que es refereix a la clau primària de la relació DESPATX (edifici, numero, superfície, telefon). Aquesta clau forana indica, per a cada empleat, el despatx on treballa. A més, l'atribut DNICap és una altra clau forana que referencia la clau primària de la mateixa relació EMPLEAT, i indica, per a cada empleat, quin és l'empleat que li fa de cap.

```
DESPATX (edifici, numero, superfície, telefon)

EMPLEAT(DNI, nom, cognom, sou, DNICap, edificidesp, numerodesp)
ON {DNICap} REFERENCIA EMPLEAT(DNI)
I {edificidesp, numerodesp} REFERENCIA DESPATX (edifici, numero)
```

Operacions

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 les dades inclou bàsicament dos aspectes: l'actualització i la consulta.

L'actualització de les dades consisteix a fer que els canvis que es produeixen a la realitat quedin reflectits a les relacions de la base de dades.

Exemple d'actualització : Si una base de dades conté, per exemple, informació dels empleats d'una empresa i l'empresa contracta un empleat, caldrà reflectir aquest canvi afegint les dades del nou empleat a la base de dades.


Hi ha tres operacions bàsiques d'actualització:

- **Inserció**, que serveix per a afegir una o més tuples a una relació.
- **Esborrat**, que serveix per a eliminar una o més tuples d'una relació.
- **Modificació**, que serveix per a alterar els valors que tenen una o més tuples d'una relació per a un o més dels seus atributs.

La **consulta** de les dades consisteix en l'obtenció de dades deduïbles a partir de les relacions que conté la base de dades.

Exemple de consulta : Si una base de dades conté, per exemple, informació dels empleats d'una empresa, pot interessar consultar el nom i cognom de tots els empleats que treballen en un despatx situat en un edifici que té per nom Marina.

L'obtenció de les dades que responen a una consulta pot requerir l'anàlisi i l'extracció de dades

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

d'una o més de les relacions que té la base de dades.

Regles d'integritat

Una base de dades conté unes dades que, en cada moment, han de reflectir la realitat o, més concretament, la situació d'una porció del món real. En el cas de les bases de dades relacionals, això significa que l'extensió de les relacions, és a dir, les tuples que les relacions contenen, han de tenir valors que reflecteixin la realitat correctament.

Sol ser força freqüent que determinades configuracions de valors per a les tuples de les relacions no tenen sentit, perquè no representen cap situació possible del món real.

Per exemple, a la relació d'esquema EMPLEAT(DNI, nom, cognom, sou), una tupla que té un valor de -1.000 per al sou, probablement no té sentit perquè els sous no poden ser negatius.

Anomenem **integritat** la propietat de les dades de correspondre a representacions plausibles del món real.

Evidentment, perquè les dades siguin íntegres, cal que compleixin diverses condicions.

El fet que els sous no puguin ser negatius, és una condició que s'hauria de complir a la relació EMPLEAT.

Les condicions que garanteixen la integritat de les dades poden ser de dos tipus:


- 1) Restriccions d'**integritat de l'usuari**. Són condicions específiques de cada BD. Els SGBD han de permetre als administradors establir certes restriccions aplicables a casos concrets, i han de garantir que es respectin durant l'exploració habitual del sistema.
- 2) Regles d'**integritat del model**. Són condicions de caire general que han de complir totes les BD que segueixin el model relacional. No cal definir-les en implementar cada BD, perquè es consideren preestablertes.

La condició de que el sou dels empleats no pugui ser negatiu és una restricció d'integritat d'usuari.

Com que la relació EMPLEAT té definit l'atribut DNI com a clau primària, el sistema validarà automàticament que no s'introdueixi més d'un empleat amb el mateix carnet d'identitat, ja que aleshores la clau primària no compliria el seu objectiu de garantir la identificació inequívoca de cada tupla, diferenciant-lo de la resta. Es tracta d'una regla d'integritat de model.

Els SGBD han de proporcionar la manera de definir les restriccions d'integritat d'usuari d'una base de dades i, un cop definides, han de vetllar perquè se satisfacin.

Les regles d'integritat del model, en canvi, no s'han de definir per a cada base de dades concreta, perquè es consideren preestablertes. Un SGBD ha de vetllar perquè es compleixin les regles d'integritat del model.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Les regles d'integritat del model relacional són:

- Unicitat de la clau primària
- Entitat de la clau primària
- Integritat referencial

La regla d'integritat d'**unicitat de la clau primària**: estableix que tota clau primària que s'esculli per a una relació no ha de tenir valors repetits. Si la clau primària està formada per més d'un atribut, la combinació de valors per als seus atributs no pot estar repetida. No poden haver més d'una tupla amb el mateix valor a la clau primària.

Un SGBD relacional haurà de garantir el compliment d'aquesta regla d'integritat per a totes les insercions i, també, per a totes les modificacions que afectin atributs que pertanyen a la clau primària de la relació.

Exemple: a la taula EMPLEAT no poden haver dos empleats amb el mateix DNI.

Exemple: a la taula DESPATX no pot haver dos despatxos amb la mateixa combinació de valors als atributs que constitueixen la clau primària edifici i numero.

La regla d'integritat d'**entitat de la clau primària** disposa que els atributs de la clau primària d'una relació no poden tenir valors nuls.

Aquesta regla és necessària perquè els valors de les claus primàries serveixen per a identificar les tuples individuals de les relacions. Si les claus primàries tinguessin valors nuls, podria passar que algunes tuples no es poguessin distingir.


Exemple: a la taula EMPLEAT no podem tenir un empleat sense valor (amb valor nul) a l'atribut DNI. La tupla <NUL, Nuria, Garcia, 1200.00> no és correcta.

Exemple: a la taula DESPATX no podem tenir un despatx que no tingui valor l'atribut numero o l'atribut edifici. La tupla <Marina, NUL, 20, 931234567> i la tupla <NUL, 120, 13, 931111111> no són correctes.

Un SGBD relacional haurà de garantir l'acompliment d'aquesta regla d'integritat en totes les insercions i, també, en totes les modificacions que afectin atributs que pertanyen a la clau primària de la relació.

La regla d'**integritat referencial** està relacionada amb el concepte de clau forana. Concretament, determina que tots els valors que pren una clau forana han de ser valors nuls o valors que existeixen a la clau primària que referencia.

La necessitat de la regla d'integritat referencial prové del fet que les claus foranes tenen per objectiu establir una connexió amb la clau primària que referencien.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Si un valor d'una clau forana no fos present a la clau primària corresponent, representaria una referència o una connexió incorrecta.

Exemple:

EMPLEAT						
<u>DNI</u>	<u>nom</u>	<u>cognom</u>	<u>sou</u>	<u>edificidesp</u>	<u>numerodesp</u>	<u>DNICap</u>
11111111A	Anna	Sala	2000.00	Marina	120	22222222B
22222222B	Àlex	Comas	2200.00	Marina	120	NUL
33333333C	David	Valls	1800.00	Diagonal	100	22222222B

DESPATX			
<u>edifici</u>	<u>numero</u>	<u>superficie</u>	<u>telefon</u>
Marina	120	20	931111111
Diagonal	100	30	932222222

Suposem que a l'exemple anterior hi hagués un empleat amb els valors <44444444D, Pere, López, València, 325>. Com que no hi ha cap despatx amb els valors València i 325 per a edifici i numero, la tupla d'aquest empleat fa una **referència incorrecta**, és a dir, indica un despatx per a l'empleat que, de fet, no existeix.


Un SGBD relacional haurà de fer complir aquesta regla d'integritat. Haurà d'efectuar comprovacions quan es produeixin les operacions següents:

- Insercions en una relació que tingui una clau forana.
- Modificacions que afectin atributs que pertanyen a la clau forana d'una relació.
- Esborrats en relacions referenciades per altres relacions.
- Modificacions que afectin atributs que pertanyen a la clau primària d'una relació referenciada per alguna altra relació.

Tornant a l'exemple, les operacions següents provocarien l'incompliment de la regla d'integritat referencial:

- Inserció de <12764411E, Jordi, Puig, **Diagonal**, **220**> a EMPLEAT.
- Modificació de <11111111A, Anna. Sala, 2000.00, Marina, 120, 22222222B> per <11111111A, Anna. Sala, 2000.00, **Marina**, **300**, 22222222B>
- Esborrat de <Marina, 120, 20, 931111111> de DESPATX.
- Modificació de <Diagonal, 120, 20, 931111111> de DESPATXOS per <**París**, **120**, 10, 931111111>.

Un SGBD relacional ha de vetllar perquè es compleixin les regles d'integritat del model. Una manera habitual de mantenir aquestes regles és rebutjar tota operació d'actualització que deixi la

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

base de dades en un estat en què alguna regla no es compleixi. En alguns casos, però, el SGBD té la possibilitat d'acceptar l'operació i efectuar accions addicionals compensatòries, de manera que l'estat que s'obtingui satisfaci les regles d'integritat, tot i haver executat l'operació.

Aquesta darrera política pot ser aplicada en les operacions d'actualització que violarien la regla d'integritat referencial següents:

- Restricció
- Actualització en cascada
- Anul·lació

La política de **restricció** consisteix a no acceptar l'operació d'actualització.
 Més concretament, la restricció en cas d'esborrat consisteix a no permetre esborrar una tupla si té una clau primària referenciada per alguna clau forana.
 Similarment, la restricció en cas de modificació consisteix a no permetre modificar cap atribut de la clau primària d'una tupla si té una clau primària referenciada per alguna clau forana.

CLIENT	
<u>numClient</u>
10
15
18


COMANDA		
<u>numComanda</u>	numClient*
1200	10
1320	10
1430	15

* referencia numClient de CLIENT

- Si apliquem la restricció en cas d'esborrat i, per exemple, volem esborrar el client número 10, no serà possible fer-ne l'esborrat perquè té comandes pendents que el referencien.
- Si apliquem la restricció en cas de modificació i volem modificar el número del client 15, no serà possible fer-ne la modificació perquè també té comandes pendents que el referencien.

La política d'**actualització en cascada** consisteix a permetre l'operació d'actualització de la tupla i a efectuar operacions compensatòries que propaguin en cascada l'actualització a les tuples que la referenciaven, per a mantenir la integritat referencial.

Més concretament, l'**actualització en cascada** en cas d'esborrat consisteix a permetre l'esborrat d'una tupla t que té una clau primària referenciada i esborrar també totes les tuples que

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

referencien t.

Similarment, l'actualització en cascada en cas de modificació consisteix a permetre la modificació d'atributs de la clau primària d'una tupla t que té una clau primària referenciada i modificar de la mateixa manera totes les tuples que referencien t.

Exemple d'aplicació de l'actualització en cascada

Suposem que tenim les relacions següents:

Relació EDIFICI:

EDIFICI	
<u>nomedifici</u>	...
Marina	...
Diagonal	...

Relació DESPATX:

DESPATX		
<u>edifici*</u>	<u>numero</u>	superficie
Marina	120	10
Marina	122	15
Marina	230	20
Diagonal	120	10

* edifici fa referència a nomedifici de EDIFICI


- Si apliquem l'actualització en cascada en cas d'esborrat i, per exemple, volem esborrar l'edifici Diagonal, s'esborrarà també el despatx Diagonal 120 que hi ha a l'edifici i ens quedarà:

Relació EDIFICI:

EDIFICI	
<u>nomedifici</u>	...
Marina	...

Relació DESPATX:

DESPATX		
<u>edifici*</u>	<u>numero</u>	superficie
Marina	120	10

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Marina	122	15
Marina	230	20

- Si apliquem l'actualització en cascada en cas de modificació i volem modificar el nom de l'edifici Marina per Mar, també es canviarà Marina per Mar als despatxos Marina 120, Marina 122 i Marina 230, i ens quedarà:

Relació EDIFICI:

EDIFICI	
<u>nomedifici</u>	...
Mar	...

Relació DESPATX:

DESPATX		
<u>edifici*</u>	<u>numero</u>	<u>superficie</u>
Mar	120	10
Mar	122	15
Mar	230	20


La política d'**anul·lació** consisteix a permetre l'operació d'actualització de la tupla i a efectuar operacions compensatòries que posen valors nuls als atributs de la clau forana de les tuples que la referencien, amb l'objectiu de mantenir la integritat referencial.

Com que generalment els SGBD relacionals permeten establir que un determinat atribut d'una relació no admet valors nuls, només s'hi pot aplicar la política d'anul·lació si els atributs de la clau forana admeten valors nuls

Més concretament, l'**anul·lació** en cas d'esborrat consisteix a permetre l'esborrat d'una tupla *t* que té una clau referenciada *i*, a més, modificar totes les tuples que referencien *t*, de manera que els atributs de la clau forana corresponent prenguin valors nuls.

Similarment, l'anul·lació en cas de modificació consisteix a permetre la modificació d'atributs de la clau primària d'una tupla *t* que té una clau referenciada *i*, a més, modificar totes les tuples que referencien *t*, clau referenciada *i*, a més, modificar totes les tuples que referencien *t*, de manera que els atributs de la clau forana corresponent prenguin valors nuls.

La millor manera d'entendre en què consisteix l'anul·lació és mitjançant un exemple. Tenim les relacions següents:

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Relació VENEDOR:

VENEDOR	
<u>numVenedor</u>
1
2
3

Relació CLIENT:

CLIENT		
<u>numClient</u>	venedorAssig*
23	1
35	1
38	2
42	2
50	...	3

* venedorAssig referencia a numVenedor de VENEDOR


- Si apliquem l'anul·lació en cas d'esborrat i, per exemple, volem esborrar el venedor número 1, es modificaran tots els clients que el tenien assignat i passaran a tenir un valor nul a venedorassig. Ens quedarà:

Relació VENEDOR:

VENEDOR	
<u>numVenedor</u>
2
3

Relació CLIENT:

CLIENT		
<u>numClient</u>	venedorAssig*
23	NULL
35	NULL
38	2
42	2

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

50	...	3
----	-----	---

* venedorAssig referencia a numVenedor de VENEDOR

- Si apliquem l'anul·lació en cas de modificació, i ara volem canviar el número del venedor 2 per 5, es modificaran tots els clients que el tenien assignat i passaran a tenir un valor nul a venedorassig. Ens quedarà:

Relació VENEDOR:

VENEDOR	
<u>numVenedor</u>
1
5
3

Relació CLIENT:

CLIENT		
<u>numClient</u>	venedorAssig*
23	NULL
35	NULL
38	NULL
42	NULL
50	...	3

* venedorAssig referencia a numVenedor de VENEDOR


El dissenyador pot escollir per a cada clau forana quina política s'aplicarà en cas d'esborrat de la clau primària referenciada i quina en cas de modificació de la clau primària referenciada. El dissenyador haurà de tenir en compte el significat de cada clau forana concreta per a poder escollir adequadament.

En cas d'esborrament, normalment s'opta per la restricció, i en cas de modificació, el més habitual és optar per l'actualització en cascada.

La política d'anul·lació és molt menys freqüent, i es posa en pràctica quan es volen conservar certes dades, encara que hagin perdut la connexió que tenien abans, de vegades amb l'esperança que la puguin recuperar més endavant.

Integritat del domini

La regla d'integritat del domini implica que tots els valors no nuls que contenen els atributs de les relacions de qualsevol BD han de pertànyer als respectius dominis declarats per als atributs en

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

qüestió.

Aquesta condició és aplicable tant pel que fa als dominis predefinits, com també pel que fa als dominis definits per l'usuari.

Recordem que els dominis poden ser de dos tipus: predefinits o definits per l'usuari. Observeu que els dominis definits per l'usuari resulten molt útils perquè ens permeten determinar de manera més específica quins seran els valors admesos pels atributs.

Suposem per exemple que a la relació EMPLEAT(DNI, nom, cognom, edatemp) hem declarat que domini(edatemp) és el domini definit per l'usuari edat. Suposem també que el domini edat s'ha definit com el conjunt dels enters que són entre 16 i 65. En aquest cas, per exemple, no serà possible inserir-hi un empleat amb un valor de 90 per a edatemp.

La regla d'integritat del domini també comporta que els operadors que és possible aplicar sobre els valors depenen dels dominis dels respectius atributs que els emmagatzemen.

2. Traducció del Model ER al Model Relacional

Els diagrames ER obtinguts a la fase de disseny conceptual cal ser traduïts al model relacional abans de ser implementats en un sistema gestor de bases de dades relacional.

A continuació us presentem les regles de traducció a través de diferents exemples. El procés es basa en les claus foranes.

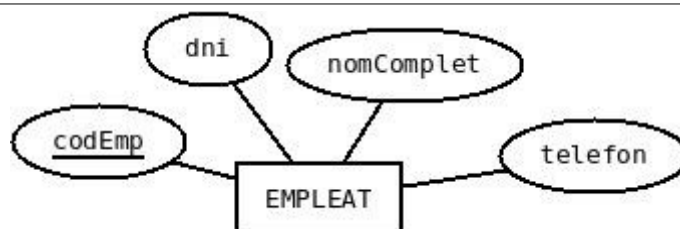
Els passos per a fer la traducció són els següents:

1. Traducció de les entitats amb els seus atributs i clau primària.
2. Traducció de les generalitzacions / especialitzacions i entitats febles.
3. Traducció de les interrelacions.

Tingueu en compte que:

- Les entitats referents a qüestions de temps no cal traduir-les: DATA, TRIMESTRE, SEMESTRE, TEMPORADA, etc.
- Quan indiquem que un atribut és clau forana, si aquest manté el mateix nom que la clau primària que fa referència, no cal indicar el nom de l'atribut.

Entitats i atributs

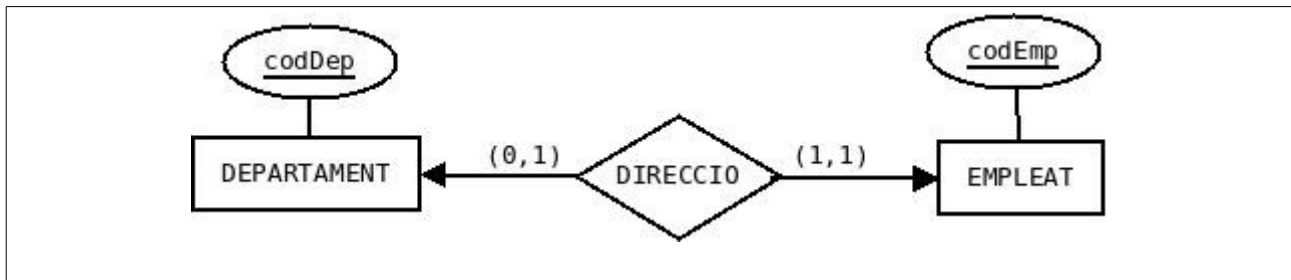


Cada entitat es tradueix a una taula.

Els atributs de l'entitat passen a ser atributs de la taula respectant la clau primària.

EMPLEAT (codEmp, dni, nomComplet, telefon)

Interrelació binària 1:1



Hi ha dues opcions:

- 1ª opció: a la taula EMPLEAT afegim la clau primària de DEPARTAMENT indicant que és clau forana.
- 2ª opció: a la taula DEPARTAMENT afegim la clau primària d'EMPLEAT indicant que és clau forana.

A continuació, amb l'exemple s'explica amb quina opció ens quedaríem.

1ª opció:

DEPARTAMENT (codDep,)

EMPLEAT (codEmp,, codDepDir)

ON {codDepDir} REFERENCIA DEPARTAMENT (codDep)

2ª opció:

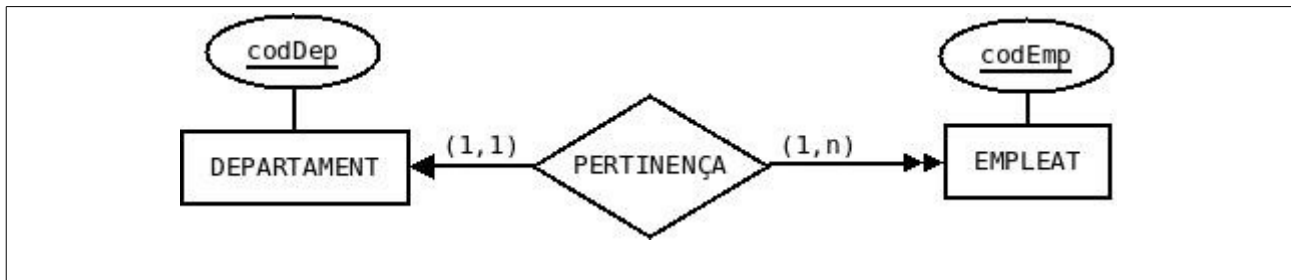
DEPARTAMENT (codDep,, codDir)

ON {codDir} REFERENCIA EMPLEAT (codEmp)

EMPLEAT (codEmp,)

És millor la 2ª opció perquè tot departament té un empleat que fa de director i per tant aquest atribut per a cada registre tindrà valor. És equivalent a dir que EMPLEAT és obligatori (mínim 1) a la interrelació DIRECCIÓ. En canvi DEPARTAMENT no és obligatori (mínim 0) a la interrelació DIRECCIÓ per què hi ha empleats que no dirigeixen cap departament i llavors codDep a DEPARTAMENT per a la majoria de registres no tindria valor (NULL).

Interrelació binària 1:N



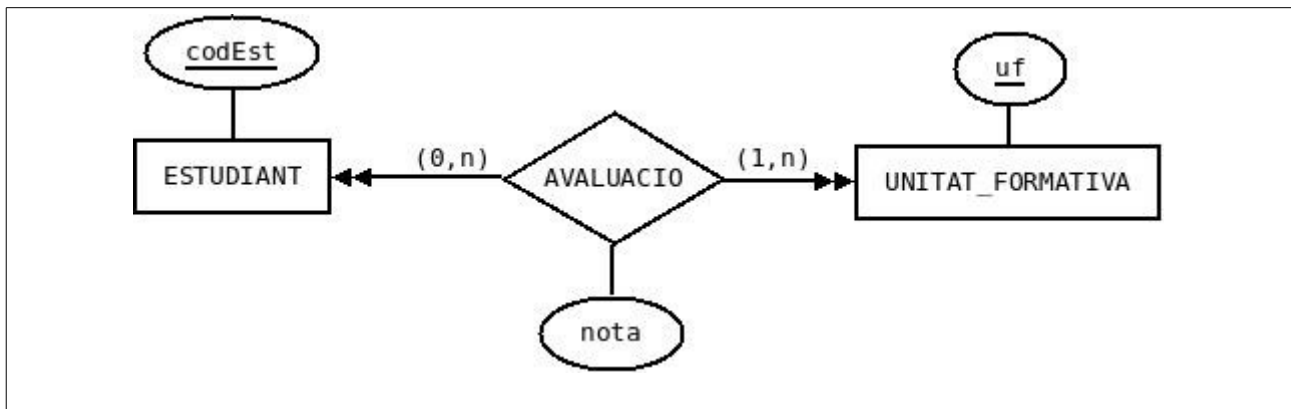
S'afegeix a la taula que ve de l'entitat que té la n (EMPLEAT) la clau primària de la taula que ve de l'entitat que té l'1 (DEPARTAMENT, codDep) i indicarem que és clau forana.

```

DEPARTAMENT (codDep, ..... )

EMPLEAT (codEmp, ....., codDep)
ON {codDep} REFERENCIA DEPARTAMENT
  
```

Interrelació binària N:M



Es crea nova taula, el nom de la qual és el nom de la interrelació (AVALUA).
 La clau primària d'aquesta taula estarà formada pels atributs que són clau primària a les taules que interrelaciona (codEst, uf) i indicarem que són clau forana.
 S'afegeixen els atributs de la interrelació (nota) com atributs d'aquesta nova taula.

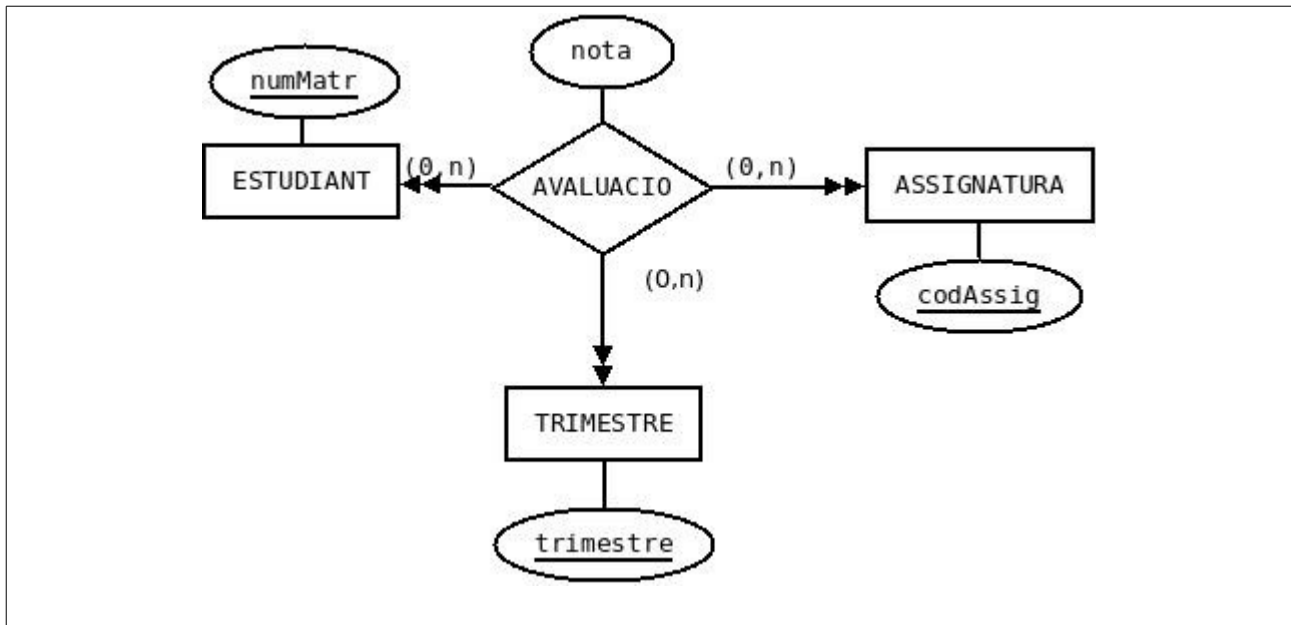
```

ESTUDIANT (codEst, ..... )

UNITAT_FORMATIVA (uf, ..... )

AVALUA (codEst, uf, nota)
ON {codEst} REFERENCIA ESTUDIANT
I {uf} REFERENCIA UNITAT_FORMATIVA
  
```

Interrelació ternària N:M:P



Es crea nova taula, el nom de la qual és el nom de la interrelació (AVALUACIO).
 La clau primària d'aquesta taula estarà formada pels atributs que són clau primària a les taules que interrelaciona (numMatr, codAssig, trimestre) i indicarem que són clau forana.
 S'afegeixen els atributs de la interrelació (nota) com atributs d'aquesta nova taula.

```
ESTUDIANT (numMatr, ....)
```

```
ASSIGNATURA (codAssig, ....)
```

```
TRIMESTRE (trimestre)
```

```
AVALUACIO (numMatr, codAssig, trimestre, nota)
```

```
ON {numMatr} REFERENCIA ESTUDIANT,
```

```
{codAssig} REFERENCIA ASSIGNATURA
```

```
I {trimestre} REFERENCIA TRIMESTRE
```

Com les entitats referents a qüestions temporals (TRIMESTRE) no cal traduir-les, la transformació quedaria així:

```
ESTUDIANT (numMatr, ....)
```

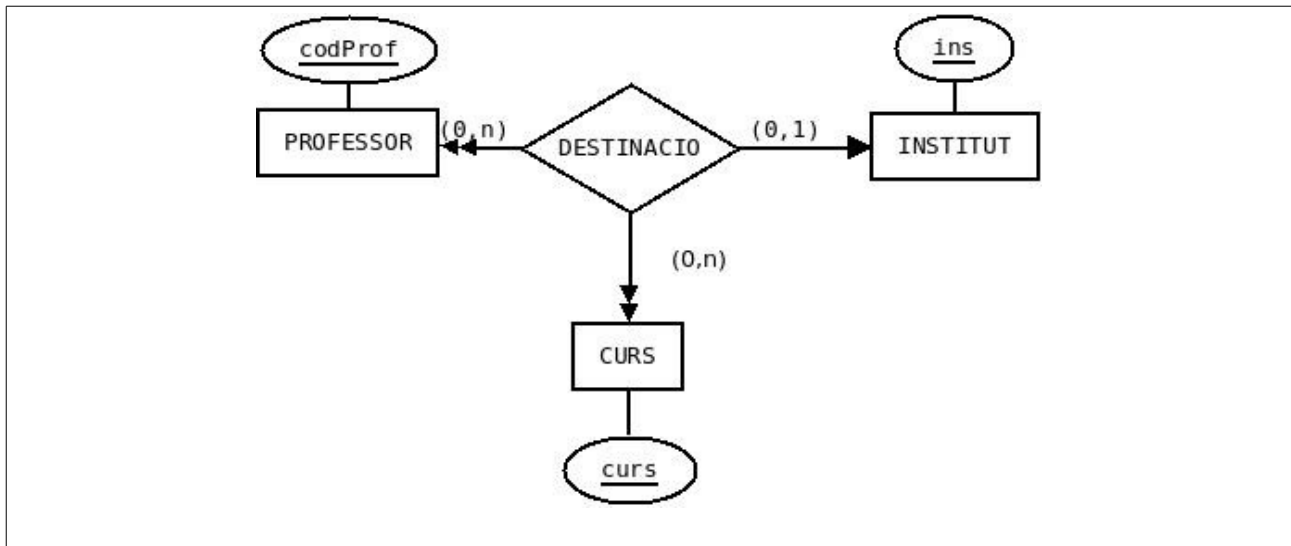
```
ASSIGNATURA (codAssig, ....)
```

```
AVALUACIO (numMatr, codAssig, trimestre, nota)
```

```
ON {numMatr} REFERENCIA ESTUDIANT
```

```
I {codAssig} REFERENCIA ASSIGNATURA
```

Interrelació binària 1:N:M



Es crea nova taula, el nom de la qual és el nom de la interrelació (DESTINACIÓ).

Els atributs d'aquesta nova taula seran els atributs que són clau primària a les taules que interrelaciona (codProf, ins, curs) indicant que són clau forana i els atributs de la interrelació (no hi ha cap).

La clau primària d'aquesta taula estarà formada pels atributs que són clau primària a les taules que venen de les entitats que tenen la n (codProf, curs).

PROFESSOR (codProf,)

INSTITUT (ins,)

CURS (curs)

DESTINACIÓ (codProf, curs, ins)
 ON {codProf} REFERENCIA PROFESSOR,
 {curs} REFERENCIA CURS
 I {ins} REFERENCIA INSTITUT

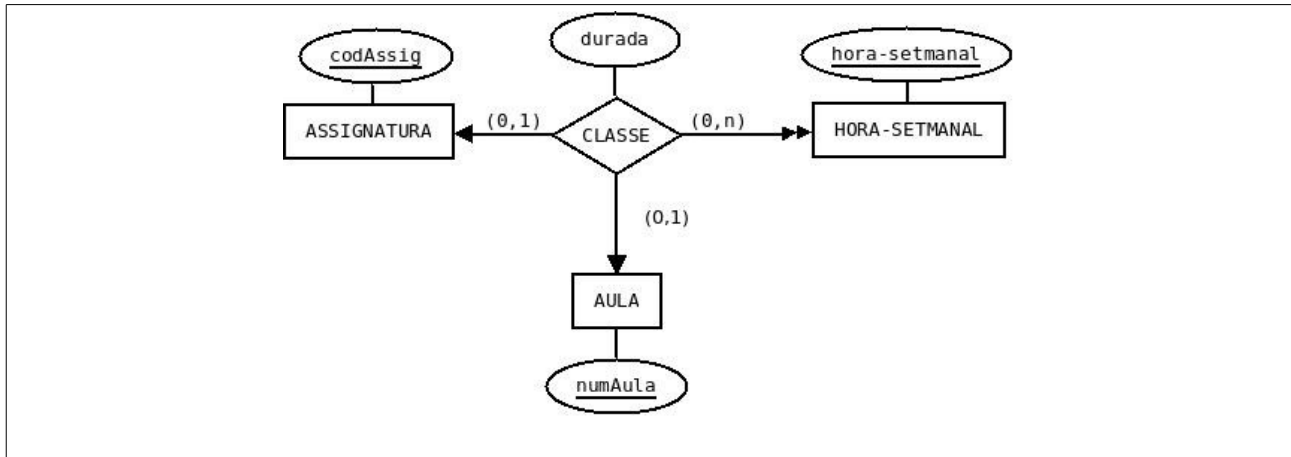
Com les entitats referents a qüestions temporals (CURS) no cal traduir-les, la transformació quedaria així:

PROFESSOR (codProf,)

INSTITUT (ins,)

DESTINACIÓ (codProf, curs, ins)
 ON {codProf} REFERENCIA PROFESSOR
 I {ins} REFERENCIA INSTITUT

Interrelació ternària 1:1:N



Es crea nova taula, el nom de la qual és el nom de la interrelació (CLASSE). Els atributs d'aquesta nova taula seran els atributs que són clau primària a les taules que interrelaciona (codAssig, hora-setmanal, numAula) indicant que són clau forana i els atributs de la interrelació (durada). La clau primària d'aquesta taula estarà formada pels atributs que són clau primària a la taula que ve de l'entitat que té la n (hora-setmanal) i el atributs que són clau primària en una de les taules que venen de les entitats que té l'1 (o codAssig o numAula, però no els dos). És a dir, hi ha dues opcions i hem d'escollir una d'elles. No hi ha cap preferència.

ASSIGNATURA (codAssig,)

HORA-SETMANAL (hora-setmanal)

AULA (numAula,....)

CLASSE (hora-setmanal, numAula,codAssig, durada)

ON {hora-setmanal} REFERENCIA HORA-SETMANAL,

{numAula} REFERENCIA AULA

I {codAssig} REFERENCIA ASSIGNATURA

Com les entitats referents a qüestions temporals (HORA-SETMANAL) no cal traduir-les, la transformació quedaria així:

ASSIGNATURA (codAssig,)

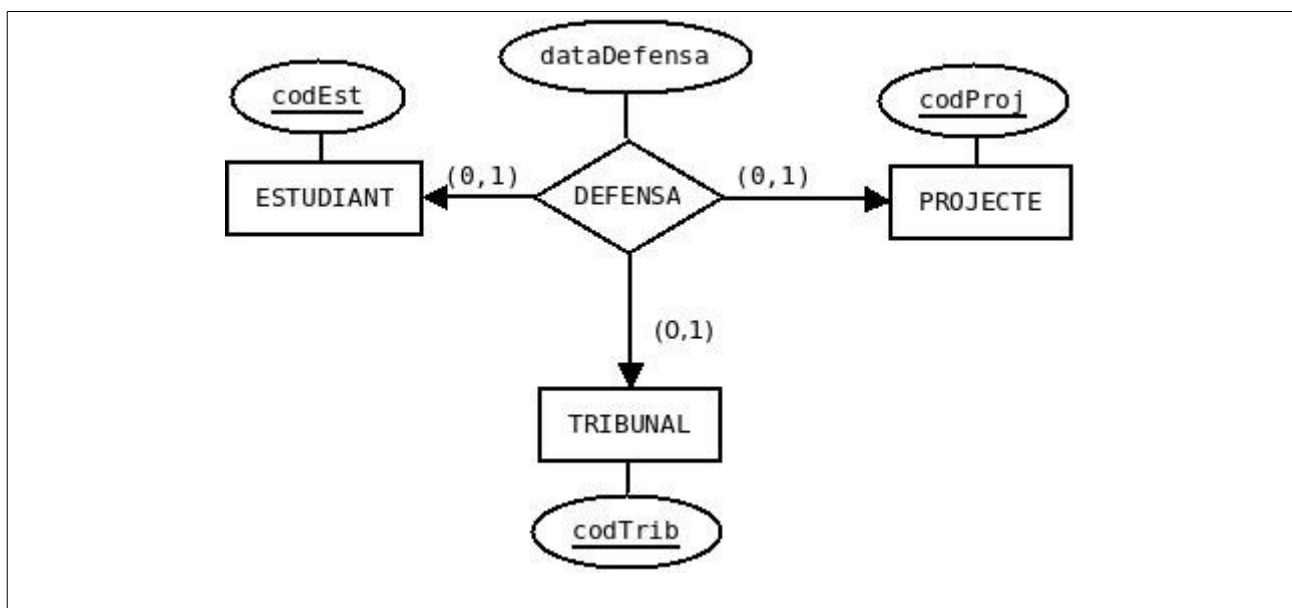
AULA (numAula,....)

CLASSE (hora-setmanal, numAula,codAssig, durada)

ON {numAula} REFERENCIA AULA

I {codAssig} REFERENCIA ASSIGNATURA

Interrelació ternària 1:1:1




Es crea nova taula, el nom de la qual és el nom de la interrelació (DEFENSA).

Els atributs d'aquesta nova taula seran els atributs que són clau primària a les taules que interrelaciona (codEst, codProj, codTrib) indicant que són clau forana i els atributs de la interrelació (dataDefensa).

La clau primària d'aquesta taula estarà formada pels atributs que són clau primària a dues taules que venen de dues de les entitats. És a dir, la clau primària pot ser:

- codEst, codProj

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

- codEst, codTrib
- codProj, codTrib

És a dir, hi ha tres opcions i hem d'escollir una d'elles. No hi ha cap preferència.

ESTUDIANT (codEst,)

PROJECTE (codProj,)

TRIBUNAL (codTrib,.....)

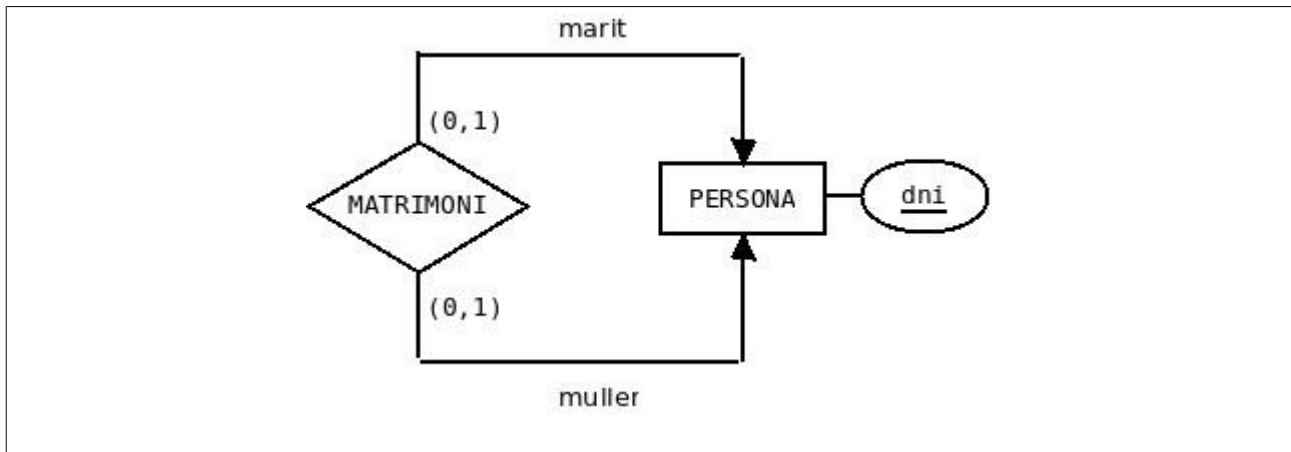
DEFENSA (codEst, codProj, codTrib, dataDefensa)

ON {codEst} REFERENCIA ESTUDIANT,

{codProj} REFERENCIA PROJECTE

I {codTrib} REFERENCIA TRIBUNAL

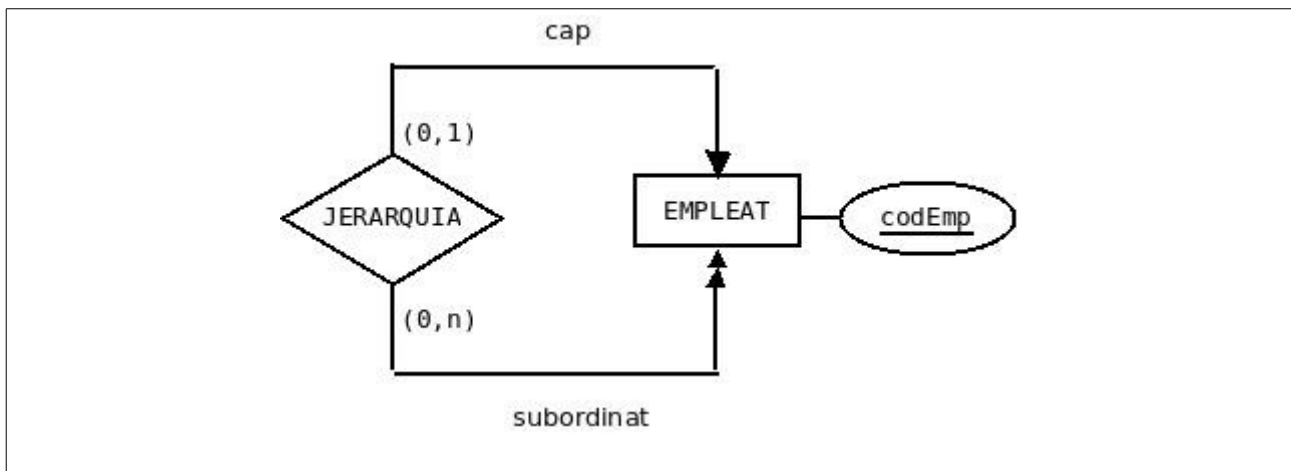
Interrelació recursiva binària 1:1



S'aplica la mateixa metodologia que a les interrelacions binàries amb aquesta connectivitat.

```
PERSONA (dni, ....., dniParella)
ON {dniParella} REFERENCIA PERSONA (dni)
```

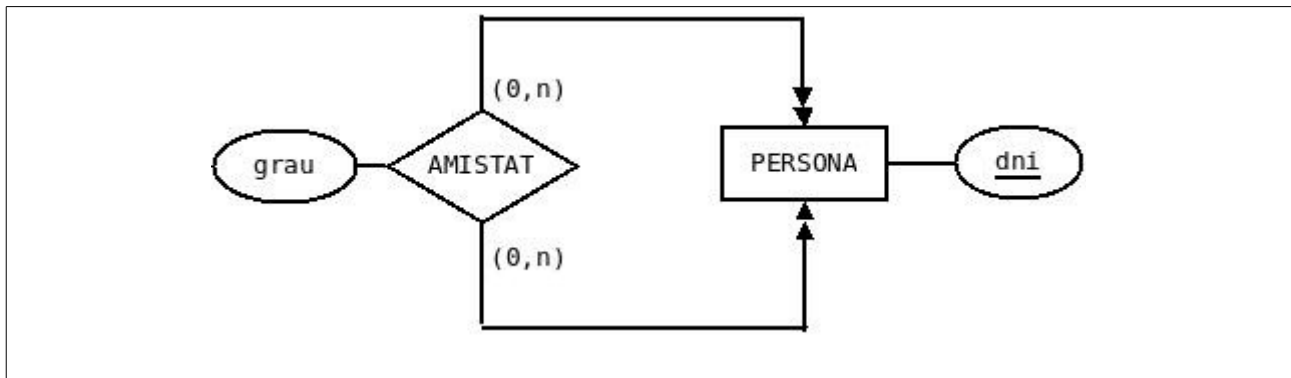
Interrelació recursiva binària 1:N



S'aplica la mateixa metodologia que a les interrelacions binàries amb aquesta connectivitat.

```
EMPLEAT (codEmp, ....., codEmpCap)
ON {codEmpCap} REFERENCIA EMPLEAT (codEmp)
```

Interrelació recursiva binària N:M

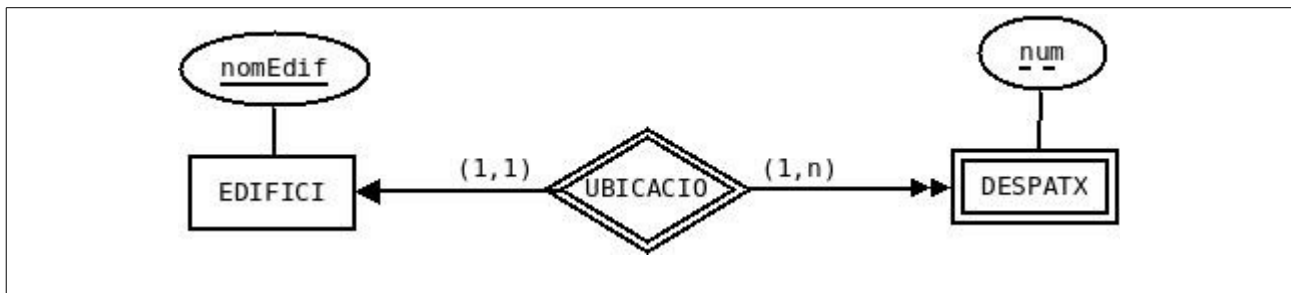


S'aplica la mateixa metodologia que a les interrelacions binàries amb aquesta connectivitat.

```
PERSONA (dni, ....)

AMISTAT (dni, dniAmic, grau)
ON {dni} REFERENCIA PERSONA
I {dniAmic} REFERENCIA PERSONA (dni)
```

Entitat feble

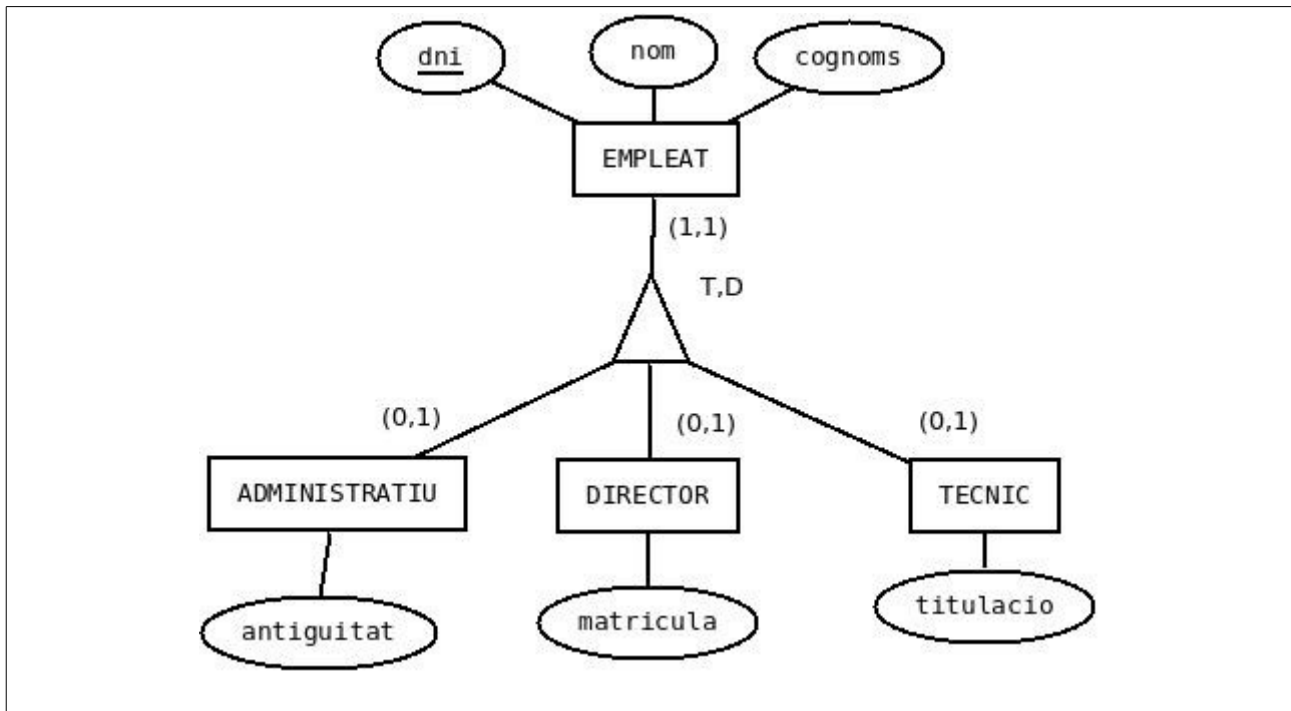


S'afegeix a la taula que ve de l'entitat que té la n (DESPATX) la clau primària de la taula que ve de l'entitat que té l'1 (EDIFICI, nomEdif) i indicarem que és clau forana. A més, aquest nou atribut forma part de la clau primària de la taula que té la n (DESPATX).

```
EDIFICI (nomEdif, ....)

DESPATX (nomEdif, num, ....)
ON {nomEdif} REFERENCIA EDIFICI
```

Generalització i especialització



Per a cada subclasse (ADMINISTRATIU, DIRECTOR, TECNIC) es genera una taula amb tots els seus atributs afegint els atributs de la clau primària de la superclasse (EMPLEAT, dni) i indicant que són clau forana. La clau primària d'aquestes taules serà la mateixa que la superclasse.


```

EMPLEAT (dni, nom, cognoms)

ADMINISTRATIU (dni, antiguitat)
ON {dni} REFERENCIA EMPLEAT

DIRECTOR (dni, matricula)
ON {dni} REFERENCIA EMPLEAT

TECNIC (dni, titulacio)
ON {dni} REFERENCIA EMPLEAT
  
```

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

3. Normalització

El disseny d'una base de dades pot ser una tasca extremadament complexa. Hi ha diferents metodologies que permeten abordar el problema de trobar l'esquema relacional que representi millor la realitat que es vol modelitzar.

Coneixem el model Entitat-Relació per establir models per a qualsevol realitat, del qual s'obté, com a resultat, el diagrama Entitat-Relació, altrament anomenat diagrama de Chen. També coneixem el procés de traducció d'un diagrama Entitat- Relació a un esquema relacional.

Per tant, si per arribar a l'esquema relacional que ha de modelitzar la realitat hem seguit el camí que consisteix a, primerament, efectuar el diagrama Entitat-Relació per després efectuar-ne la traducció al model relacional, i el diagrama Entitat- Relació era correcte, haurem obtingut un esquema relacional del tot correcte.

Aquest seria el camí aconsellable.

Però no sempre és així i ens trobem dissenys efectuats directament en l'esquema relacional. Hi ha diferents causes que ho provoquen:

- D'entrada, el model Entitat-Relació és posterior al model relacional i, per tant, hi ha bases de dades que van ser formulades directament en la terminologia relacional.
- Hi ha dissenyadors que “no volen perdre el temps” en un model Entitat- Relació i dissenyen directament en el model relacional.
- De vegades, s'ha de modificar la base de dades a causa de noves necessitats, i el disseny s'efectua directament sobre aquesta en lloc d'analitzar-se i realitzar-se sobre el model Entitat-Relació per després transferir els canvis a l'esquema relacional.


Fixeu-vos que donem un suport absolut al fet d'utilitzar el model Entitat-Relació per obtenir-ne posteriorment el model relacional. Un bon disseny en el model Entitat-Relació acostuma a proporcionar una base de dades relacional ben dissenyada, cosa que no passarà si el disseny Entitat-Relació incorpora errors.

Dependències funcionals

El procés de normalització es basa en els conceptes de dependència funcional i dependència funcional completa.

Una **dependència funcional** és una restricció sobre una relació amb esquema $R(A_1, A_2, \dots, A_n)$ denotada com $\{X\} \rightarrow \{Y\}$, on $\{X\}$ i $\{Y\}$ són subconjunts de $\{A_1, A_2, \dots, A_n\}$, que garanteix que donat un valor de $\{X\}$, aquest determina de manera única el valor de $\{Y\}$; és a dir, cada valor de $\{X\}$ té associat un, i només un, valor de $\{Y\}$. En aquest cas, diem que $\{Y\}$ depèn funcionalment de $\{X\}$ o, alternativament, $\{X\}$ determina funcionalment $\{Y\}$.

Intentarem d'explicar el significat de la definició de dependència funcional mitjançant l'exemple de relació que es mostra a continuació:

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>
1	a1	100	Reus
1	a2	150	Reus
2	a1	200	Vic
2	a2	250	Vic
3	a2	100	Vic

La clau primària de la relació d'esquema SUBMINISTRAMENTS (CodiProv, CodiArticle, Quantitat, CiutatProv) està formada pels atributs CodiProv i CodiArticle. Aquesta relació representa quins articles subministren els diferents proveïdors i en quina quantitat. A més, també enregistra la ciutat de cada proveïdor.

Alguns exemples de dependències funcionals que existeixen en aquesta relació serien:

$\{ \text{CodiProv}, \text{CodiArticle} \} \rightarrow \{ \text{Quantitat} \}$
 $\{ \text{CodiProv}, \text{CodiArticle} \} \rightarrow \{ \text{CiutatProv} \}$

Atès que $\{ \text{CodiProv}, \text{CodiArticle} \}$ és la clau primària de la relació SUBMINISTRAMENTS, estem segurs que cada valor de la parella $\{ \text{CodiProv}, \text{CodiArticle} \}$ determina de manera unívoca els valors dels atributs Quantitat i CiutatProv.

També seria dependència funcional:

$\{ \text{CodiProv} \} \rightarrow \{ \text{CiutatProv} \}$

En aquest cas també podem veure que cada cop que l'atribut CodiProv pren el mateix valor, el valor de l'atribut CiutatProv es repeteix.

El significat que té aquest fet en el món real és que un proveïdor està localitzat sempre a la mateixa ciutat.


En canvi, considerem el següent:

$\{ \text{CodiArticle} \} \rightarrow \{ \text{Quantitat} \}$

Això no seria dependència funcional, atès que no sempre que l'atribut CodiArticle pren el mateix valor, es repeteix el valor associat a l'atribut Quantitat.

Les conclusions principals que podem extreure de la definició i l'exemple de les dependències funcionals són aquestes:

a) Una dependència funcional $\{ X \} \rightarrow \{ Y \}$ sobre una relació R no és més que una funció que s'estableix entre un conjunt d'originals $\{ X \}$ i un conjunt d'imatges $\{ Y \}$. És més, una dependència funcional caracteritza una propietat de totes les possibles instàncies d'una relació. Així, doncs, una

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

dependència funcional representa una propietat d'intensió de la relació i, per tant, per a identificar-la cal analitzar detingudament el significat dels atributs que hi intervenen.

b) La clau primària d'una relació sempre determina funcionalment la resta d'atributs de la relació. Aquesta conclusió es pot estendre a totes les claus alternatives que la relació pugui tenir.

La primera conclusió és particularment important. A l'exemple, nosaltres hem deduït que el codi d'un proveïdor en determina funcionalment la ciutat. A més, aparentment, hem arribat a aquesta conclusió per examen de l'extensió de la relació Subministraments. Però això no és del tot exacte o, dit d'una altra manera, no és suficient. Examinar l'extensió d'una relació ens ajuda a descartar dependències funcionals (a l'exemple hem descartat que el codi dels articles determini el nombre d'unitats que un proveïdor és capaç de subministrar d'aquest article), però no ens permet d'assegurar rotundament que hi ha dependència funcional. Realment, hem d'anar més enllà, i preguntar-nos si el comportament que inferim a partir de l'examen de l'extensió es dona o no en la realitat.

En resum, per a deduir dependències funcionals, no necessitem en absolut examinar l'extensió de la relació, atès que és una propietat d'intensió. En tot cas, examinar l'extensió ens pot ajudar a corroborar les dependències funcionals.

Una dependència funcional $\{ X \} \rightarrow \{ Y \}$ és **dependència funcional completa** quan cap subconjunt propi de $\{ X \}$ determina funcionalment $\{ Y \}$.

Quan fem el disseny conceptual d'una base de dades i a continuació trobem el disseny lògic corresponent, totes les dependències funcionals que hi pugui haver a les diferents relacions que formen part del disseny lògic han de ser dependències funcionals completes. Altrament, com veurem tot seguit, es produeixen anomalies de disseny que denoten que el disseny conceptual de la base de dades no és prou correcte.


Anomalies de disseny

Com a conseqüència d'un disseny dolent, podem tenir relacions que presenten un alt grau de redundància, és a dir, presenten repeticions que són evitables.

Aquest fet en complica el manteniment, atès que es produeixen anomalies. A continuació analitzem aquestes anomalies sobre la nostra relació d'exemple SUBMINISTRAMENTS:

a) Anomalies de modificació: si un proveïdor canvia de ciutat, caldrà posar la nova ciutat del proveïdor a totes les tuples que facin referència al proveïdor en qüestió, si no volem que la base de dades sigui inconsistent. Per exemple, si el proveïdor amb CodiProv = 1 canvia de ciutat i passa a Salou, hauriem de modificar dues tuples; si aquest proveïdor subministrés mil articles diferents, hauríem de modificar mil tuples, etc. La situació ideal fóra que només haguéssim de fer una modificació.

SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

1	a1	100	Reus Salou
1	a2	150	Reus Salou
2	a1	200	Vic
2	a2	250	Vic
3	a2	100	Vic

Les anomalies de modificació obliguen a modificar totes les tuples que guarden un fet determinat, si aquest canvia.


b) Anomalies d'esborrament: si un proveïdor que només subministra un producte (per exemple, el proveïdor amb CodiProv = 3), deixés de subministrar-lo, caldrà esborrar-ne la tupla de la relació Subministraments. Com a conseqüència, haurem perdut les dades personals del proveïdor en qüestió, en aquest cas, el codi de proveïdor i la ciutat:

SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>
1	a1	100	Reus
1	a2	150	Reus
2	a1	200	Vic
2	a2	250	Vic
3	a2	100	Vic

A causa de les anomalies d'esborrament es poden perdre fets elementals sense voler.

c) Anomalies d'inserció: no podem emmagatzemar dades personals d'un proveïdor amb, per exemple, CodiProv = 4 i de la ciutat de Mollet, si no sabem els articles que subministra, atès que caldria afegir una tupla a la relació de Subministraments amb valor null per a l'atribut CodiArticle, i es violaria la regla d'integritat d'entitat de la clau primària:

SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>
1	a1	100	Reus
1	a2	150	Reus
2	a1	200	Vic
2	a2	250	Vic
3	a2	100	Vic

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

4	NULL	NULL	Mollet
---	------	------	--------

Les anomalies d'inserció consisteixen a no poder inserir fets elementals de manera independent.

L'origen de totes aquestes anomalies rau en que la relació SUBMINISTRAMENTS descriu dos fets elementals del món real diferents: d'una banda, els articles que subministra cada proveïdor; i de l'altra, el proveïdor en si mateix. A més, aquests fets són completament independents entre si, atès que els articles concrets que subministra cada proveïdor, no guarda cap relació directa amb el fet que el proveïdor sigui, per exemple, d'una ciutat o d'una altra, i a l'inrevés. En tot cas, entre aquests dos fets hi ha una relació indirecta, perquè afecten un mateix individu del món real, és a dir, el mateix proveïdor.

En conclusió, tota relació que no representa un concepte (o fet elemental) únic del món real està subjecta a presentar redundàncies, anomalies de manteniment i inconsistències potencials. Aquest és el cas de la nostra relació SUBMINISTRAMENTS.

Teoria de la normalització

La **teoria de la normalització** és un mètode que permet assegurar si un disseny relacional (tant si prové de la traducció d'un diagrama Entitat- Relació com si s'ha efectuat directament en el model relacional) és més o menys correcte. Es fonamenta en el principi bàsic següent: tota relació ha de descriure un concepte semàntic únic.


La teoria de la normalització ens permet de reconèixer els casos en els quals aquest principi no es compleix. El mecanisme que fa servir per a això són les formes normals (FN).

Una relació està en una forma normal determinada, si satisfà un conjunt de restriccions determinades que són pròpies d'aquesta forma normal. La violació d'aquestes restriccions origina que la relació tingui les anomalies i les redundàncies que hem esmentat abans.

Hi ha diverses formes normals, de les quals nosaltres estudiarem les tres primeres i la forma normal de Boyce-Codd. Cada forma normal indica unes restriccions específiques que ha de complir una relació. Com més gran és el grau d'una forma normal, més restrictiva és aquesta forma normal i elimina més redundàncies que les formes normals de grau menor, atès que inclou les restriccions d'aquelles i agrega una restricció addicional.

Quan una relació no compleix una forma normal, és perquè viola la restricció associada a aquella forma normal. Per a aconseguir que es verifiqui la forma normal, caldrà evitar la condició que fa que es violi la restricció en qüestió. El procediment que aplicarem per a aconseguir que no es violi la restricció associada a la forma normal rep el nom de normalització.

La primera forma normal (1FN) es basa en el concepte de valor atòmic, mentre que la resta de formes normals es basen en el concepte de dependència. La segona forma normal (2FN), la tercera forma normal (3FN) i la forma normal de Boyce-Codd (FNBC) es basen en el concepte de dependència funcional.

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

Primera forma normal (1FN)

Una relació està en **primera forma normal (1FN)** si, i només si, cap atribut de la relació és en si mateix una relació, és a dir, si tot atribut de la relació és atòmic, no descomponible, no grup repetitiu.

La relació Subministraments que mostrem tot seguit no està en 1FN, atès que els atributs CodiArticle i Quantitat no són atòmics:

SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>
1	a1	100	Reus
	a2	150	
2	a1	200	Vic
	a2	250	
3	a2	100	Vic

Per a aconseguir de normalitzar una relació a 1FN, caldrà aplicar una operació que es coneix amb el nom d'aplanar. Com a conseqüència d'aquesta operació, la relació Subministraments quedaria de la manera següent:


SUBMINISTRAMENTS			
<u>CodiProv</u>	<u>CodiArticle</u>	<u>Quantitat</u>	<u>CiutatProv</u>
1	a1	100	Reus
1	a2	150	Reus
2	a1	200	Vic
2	a2	250	Vic
3	a2	100	Vic

És important que ens adonem que quan s'aplanar la relació Subministraments original, la clau primària de la relació canvia i passa a ser composta. També cal destacar que el model relacional sempre garanteix que les relacions estan en 1FN, atès que només hi ha una estructura per a representar les dades (la relació) i, a més, cada dada es representa uniformement mitjançant valors atòmics.

Segona forma normal (2FN)

Una relació està en **segona forma normal (2FN)** si, i només si, està en 1FN i tot atribut no clau depèn funcionalment en forma completa de la clau primària.

Hi ha una excepció: un atribut pot dependre funcionalment de part de la clau primària, si aquest

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

atribut és part d'una clau alternativa.

És important de destacar que la 2FN es basa en el concepte de dependència funcional completa i, per tant, aquesta forma normal només es pot violar quan la clau primària d'una relació és composta. En resum, aquelles relacions amb clau primària simple, és a dir, amb clau primària formada per un únic atribut sempre estaran en 2FN.

Per exemple, la relació de SUBMINISTRAMENTS que mostrem tot seguit no està en 2FN:

```
{ CodiProv , CodiArticle } → { Quantitat }
{ CodiProv , CodiArticle } → { CiutatProv }
{ CodiProv } → { CiutatProv }
```

La idea que hi ha al darrere de la 2FN és molt simple. Si un atribut no clau (en el nostre exemple, CiutatProv) depèn funcionalment d'un subconjunt propi de la clau (CodiProv, a l'exemple), és perquè representa un fet d'aquest subconjunt propi i, per tant, tots dos (l'atribut no clau i el subconjunt propi de la clau) són un altre concepte semàntic (en el nostre cas, la ciutat de residència del proveïdor). La redundància que hi ha en una relació que no està en 2FN és immediata: els valors de l'atribut no clau es repetiran per a tots els valors diferents de la part de la clau de la qual no depèn.

Per a aconseguir de passar una relació a 2FN, cal que evitem que hi hagi dependències funcionals no completes respecte de la clau. Per tant, tots els atributs que participen en la dependència funcional no completa s'hauran de projectar en una nova relació que correspon al concepte semàntic que representen, tal com mostrem a continuació:

SUBMINISTRAMENTS (CodiProv,CodiArticle, Quantitat)
 CIUTATSPROVEÏDORS (CodiProv,Ciutat)

D'aquesta manera, els diferents conceptes semàntics es descriuen en diferents relacions seguint el principi bàsic de la normalització.

Tercera forma normal (3FN)


Una relació està en tercera forma normal (3FN) si, i només si, està en 2FN i cap atribut no clau depèn funcionalment de cap altre conjunt d'atributs no clau.

L'excepció aplicada a la 2FN es propaga també a la tercera forma normal.

Considerem la relació CLIENTS que mostrem a continuació:

CLIENTS (CodiCli,Carrer,Numero ,Ciutat,Provincia)

```
{ CodiCli } → { Carrer }
{ CodiCli } → { Numero }
{ CodiCli } → { Ciutat }
```

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

$\{ \text{CodiCli} \} \rightarrow \{ \text{Provincia} \}$
 $\{ \text{Ciutat} \} \rightarrow \{ \text{Provincia} \}$

Aquesta relació està en segona forma normal, però no en tercera forma normal, atès que hi ha una dependència funcional $\{ \text{Ciutat} \} \rightarrow \{ \text{Provincia} \}$ entre dos atributs no clau i, a més, cap dels dos no forma part ni és clau alternativa de la relació i, per tant, l'excepció no s'aplica.

De nou, el fet que l'atribut Provincia depengui funcionalment de l'atribut Ciutat és independent del client en si mateix. Les redundàncies i anomalies que presenta aquesta relació són immediates.

Per a normalitzar una relació que viola la 3FN, caldrà evitar la dependència funcional entre atributs no clau. Per tant, els atributs que participen en la dependència funcional, s'hauran de projectar en una nova relació que correspon al concepte semàntic que representen, tal com mostrem a continuació:

CLIENTS (<u>CodiCli</u> , Carrer, Numero, Ciutat) CIUTATS (<u>Ciutat</u> , Provincia)
--

$\{ \text{CodiCli} \} \rightarrow \{ \text{Carrer} \}$
 $\{ \text{CodiCli} \} \rightarrow \{ \text{Numero} \}$
 $\{ \text{CodiCli} \} \rightarrow \{ \text{Ciutat} \}$

$\{ \text{Ciutat} \} \rightarrow \{ \text{Provincia} \}$

Forma normal de Boyce-Codd (FNBC)

Analitzem ara la relació Notes següent:


NOTES (<u>DniAlumne</u> , <u>CodiAssignatura</u> , CodiMatricula, Nota)
--

La relació Notes verifica la 1FN (recordem que el model relacional garanteix sempre per defecte aquesta forma normal). Aquesta relació també està en 2FN, malgrat que hi ha una dependència funcional no completa de part de la clau cap a un atribut no clau ($\{ \text{DniAlumne} \} \rightarrow \{ \text{CodiMatricula} \}$). Atès que aquesta dependència involucra un atribut no clau (CodiMatricula) que forma part d'una clau alternativa de la relació (CodiAssignatura, CodiMatricula), s'aplica l'excepció. Per acabar, la relació també està en 3FN encara que hi ha una dependència funcional entre atributs no clau $\{ \text{CodiMatricula} \} \rightarrow \{ \text{DniAlumne} \}$.

De nou, atès que involucra atributs que formen part de claus alternatives, s'aplica l'excepció.

Malgrat això, tal com podem veure a continuació, la relació Notes presenta redundàncies i anomalies:

NOTES			
<u>DniAlumne</u>	<u>CodiAssignatura</u>	CodiMatricula	Nota

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

234567654	05.001	123	A
234567654	04.002	123	B
45678323	05.002	215 312	B
45678323	05.001	215 312	B
45678323	04.002	215 312	C

a) Redundàncies: per cada assignatura diferent en què estigui matriculat un alumne, es repetirà el seu codi de matrícula.

b) Anomalies de modificació: si volem modificar, per exemple, el codi de matrícula de l'estudiant amb DNI 45678323 perquè sigui 312 en comptes de 215, haurèm de modificar tres tuples. Com ja sabem, idealment només en voldríem modificar una.

L'origen d'aquestes anomalies és històric i es deu a un error d'omissió. Quan Codd va enunciar la 2FN i la 3FN l'any 1970, no va considerar la possibilitat que en una relació hi pogués haver diverses claus candidates que fossin compostes, ni tampoc no va considerar la possibilitat que entre aquestes hi pogués haver encavalcaments. Per això, l'any 1974, Boyce i Codd van proposar la forma normal de Boyce-Codd que soluciona les limitacions de la 2FN i la 3FN. De fet, sovint es normalitza una relació directament a la forma normal de Boyce-Codd sense passar prèviament per la 2FN i la 3FN.

Una relació està en **forma normal de Boyce-Codd (FNBC)** si, i només si, està en 1FN, i si tots els determinants són clau candidata de la relació.

A la definició apareix un concepte nou: el concepte determinant.


Donada una dependència funcional $\{ X \} \rightarrow \{ Y \}$, un determinant és el conjunt $\{ X \}$.

Per tant, si ho expressem d'una manera informal, els determinants en una dependència funcional són els orígens de fletxa.

Si tornem a examinar la relació NOTES, i busquem totes les dependències funcionals que hi ha en la relació, arribem a la situació següent:

NOTES (DniAlumne, CodiAssignatura, CodiMatricula, Nota)

$\{ \text{DniAlumne} \} \rightarrow \{ \text{CodiMatricula} \}$
 $\{ \text{DniAlumne}, \text{CodiAssignatura} \} \rightarrow \{ \text{Nota} \}$
 $\{ \text{DniAlumne}, \text{CodiAssignatura} \} \rightarrow \{ \text{CodiMatricula} \}$
 $\{ \text{CodiAssignatura}, \text{CodiMatricula} \} \rightarrow \{ \text{Nota} \}$
 $\{ \text{CodiAssignatura}, \text{CodiMatricula} \} \rightarrow \{ \text{DniAlumne} \}$
 $\{ \text{CodiMatricula} \} \rightarrow \{ \text{DniAlumne} \}$

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

La taula següent mostra tots els determinants (primera columna) i totes les claus candidates de la relació NOTES (segona columna):

NOTES	
Determinants	Claus candidates
DniAlumne	DniAlumne, CodiAssignatura
CodiMatricula	CodiAssignatura, CodiMatricula
DniAlumne, CodiAssignatura,	
CodiMatricula , CodiAssignatura	

Atès que no tots els determinants són clau candidata de la relació Notes, arribem a la conclusió que NOTES no està en FNBC. De nou, per a normalitzar la relació, caldrà evitar la condició que causa la violació de la restricció. Per tant, haurem d'evitar les dependències funcionals $\{ \text{DniAlumne} \} \rightarrow \{ \text{CodiMatricula} \}$ i $\{ \text{CodiMatricula} \} \rightarrow \{ \text{DniAlumne} \}$ de la relació NOTES.

Això s'aconsegueix si es projecten els atributs involucrats en una nova relació que correspon al concepte semàntic que representen. Ara bé, en aquest cas, i a diferència dels casos anteriors, tenim diverses alternatives:

a)

NOTES (DniAlumne, CodiAssignatura, Nota)
 ALUMNES (DniAlumne, CodiMatricula)

b)

NOTES (CodiMatricula, CodiAssignatura, Nota)
 ALUMNES(CodiMatricula, DniAlumne)


c)

NOTES (DniAlumne, CodiAssignatura, Nota)
 ALUMNES(CodiMatricula, DniAlumne)

d)

NOTES (CodiMatricula, CodiAssignatura, Nota)
 ALUMNES(DniAlumne, CodiMatricula)

Les diferents possibilitats de projecció d'atributs entre la relació NOTES i la nova relació ALUMNES s'obtenen com a resultat de combinar les claus candidates de la relació original NOTES. Com a dissenyadors, som lliures de triar qualsevol de les opcions presentades. Ara bé,

 INSTITUT AUSIÀS MARCH Consorci d'Educació de Barcelona Generalitat de Catalunya Ajuntament de Barcelona	APUNTS Desenvolupament d'Aplicacions Web Mòdul 2: Bases de Dades	FP_ICC0M02
--	---	-------------------

les més naturals són les opcions a o b, atès que les opcions c i d són mixtes. Entre les opcions a i b, des del punt de vista de la Secretaria del Centre, possiblement l'opció b seria més coherent*.

Observem que, independentment de l'opció triada, les relacions NOTES i ALUMNES estan en FNBC, atès que tots els determinants són claus candidates de la relació. Això és particularment important per a la nova relació ALUMNES.

Conclusions sobre dependències funcionals en les formes normals

- a) Sempre és possible normalitzar fins a la forma normal de Boyce-Codd.
- b) El procés de normalització no és únic.
- c) Donat un model lògic inicial d'una base de dades, si hi apliquem un procés de normalització, el model lògic final obtingut equival sempre al model lògic inicial. És a dir, el procés de normalització preserva la semàntica (o significat) del model lògic inicial.
- d) Com a conseqüència de la normalització, el model lògic final és millor que el model lògic inicial pels motius següents:
 - Elimina redundàncies i anomalies.
 - Separa fets semànticament diferents.

Altres formes normals

Existeixen la quarta forma normal (4FN) i la cinquena forma normal (5FN) però aquestes no s'estudien en aquest curs.