

notebook

GOOD MONDAYS PAPER

Sections

1 *Introducció*

2 *El model relacional*

3 *Components logics*

4 *Introducció al diseny de bases de dades
relacionals*

5 *Transaccions i concurrencia*

6 *Estructures físiques d'emmagatzematge i
mètodes d'accés*

7

8

9

10

11

12

T

1

2

3

4

5

6

7

8

9

10

11

12

•
•
•

T

1

2

3

4

5

6

7

8

9

10

11

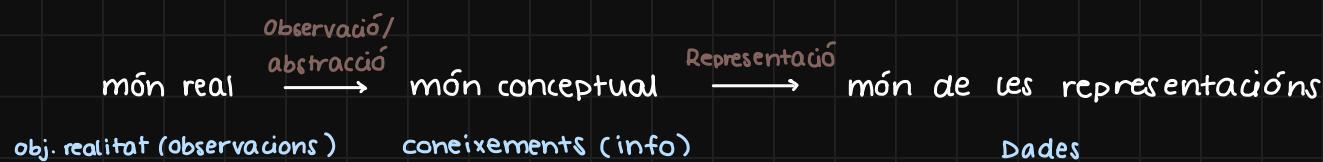
12

Section One

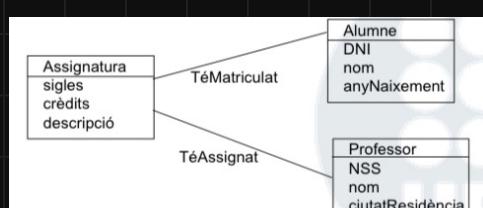
Introducció

[BACK TO INDEX](#)

ELS TRES mons



* mòn CONCEPTUAL



Esquemes amb la informació que el client necessita tenir a la seva BD.

- o Clàsses d'objectes: conjunt d'objectes que comparteixen propietats, semàntica, altres obj.
- o Atribut: Propietat característica
- o Associació: interrelació

* mòn REPRESENTACIÓ

- o Fitxers
- o BD relacionals* → SGDB (sistema gestor de BD) software per manegar BD's

Tipus de BD:

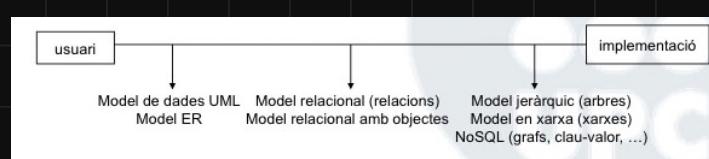
- Models jeràrquics → àrbres
- Models xarxes → grafs
- Models relacionals* → taules

CONCEPCION DE BD

Una BD és una col·lecció de dades que permet representar classes d'objectes els seus atributs i les seves associacions.

Model de BD: conjunt de components o eines del SGBD per estructurar i manipular dades

- Estructura de dades (construeixen BD)
- Operacions (manipular / consultar dades)
- Regles d'integritat



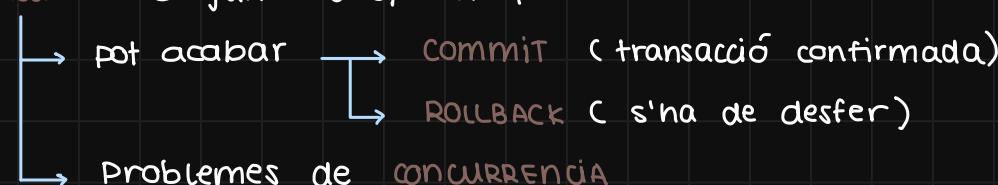
SGBD

- Persistència
- Eficiència en l'accés
- Emmagatzematge de quantitats massives de dades
- Accés multi-usuari
- Seguretat
- Fiabilitat
- Conveniència

* Accés multi-usuari

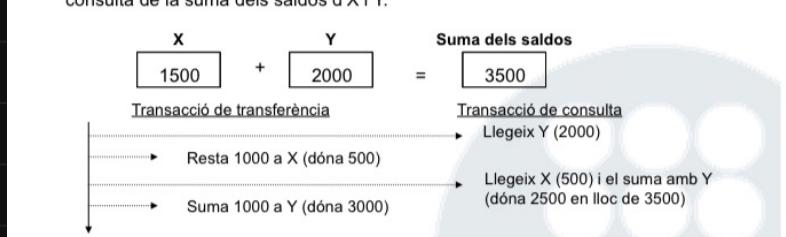
Diversos usuaris poden accedir concurrentment a la mateixa BD

Concepte Transacció: conjunt d'op simples



* Exemple de problema de concurrencia entre transaccions:

Transferència de 1000 euros del compte X al compte Y concurrentment amb una consulta de la suma dels saldo d'X i Y.



* Per que no passi s'utilitza el BLOQUEIG (limitacions als accessos)

* Seguretat

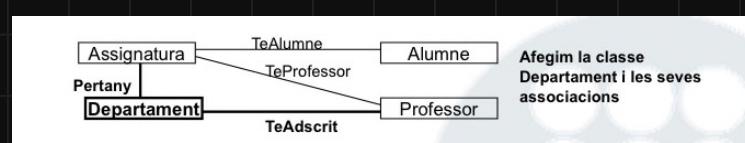
- Identificació i autenticació dels usuaris: ús de paraules clau, targetes, tècniques biomètriques.
- Definir autoritzacions i drets d'accés: per accedir a determinades dades, operacions concretes.

* Fiabilitat

- Regles d'integritat del model
- Restriccions d'integritat usuaris
- Redundàncies controlades
- Transaccions
- No pèrdua de dades
- Mecanismes de definició i manteniment
- Mecanismes de recuperació (copies de seguretat, diari)

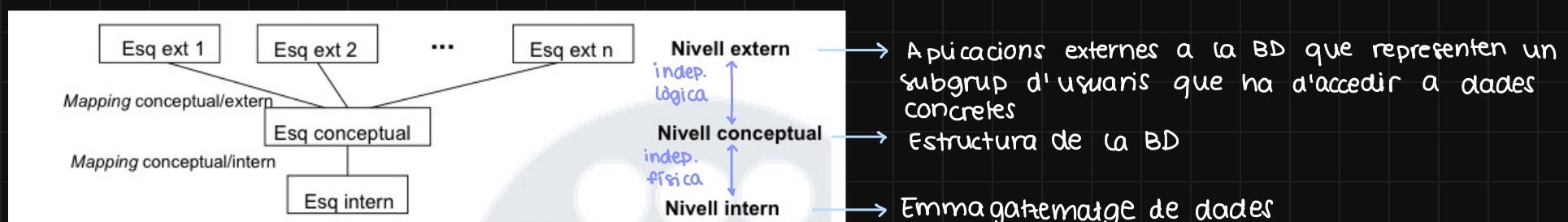
* Conveniència

Flexibilitat als canvis → canvis conceptuais (afegir/suprimir atributs, classes...)

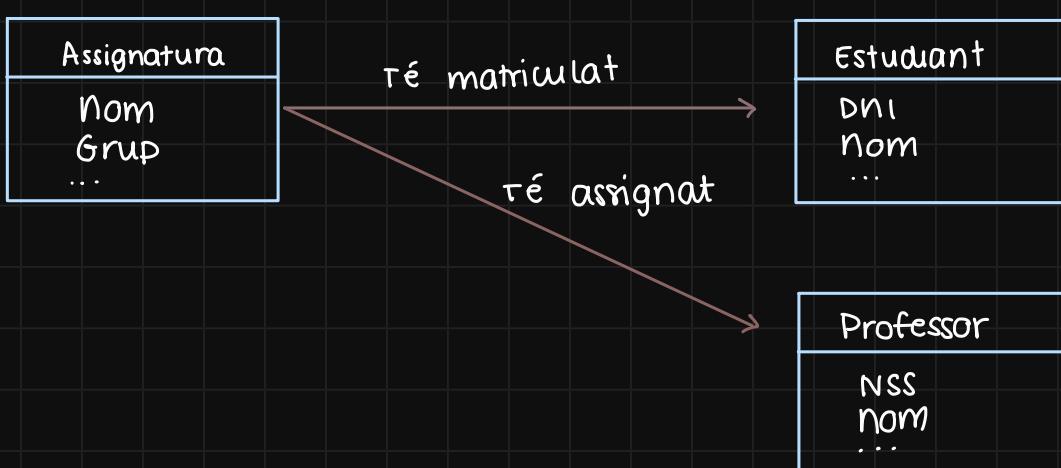


→ canvis a l'enregistrament físic

○ ARQUITECTURA ANSI/SPARC



Esq. conceptual

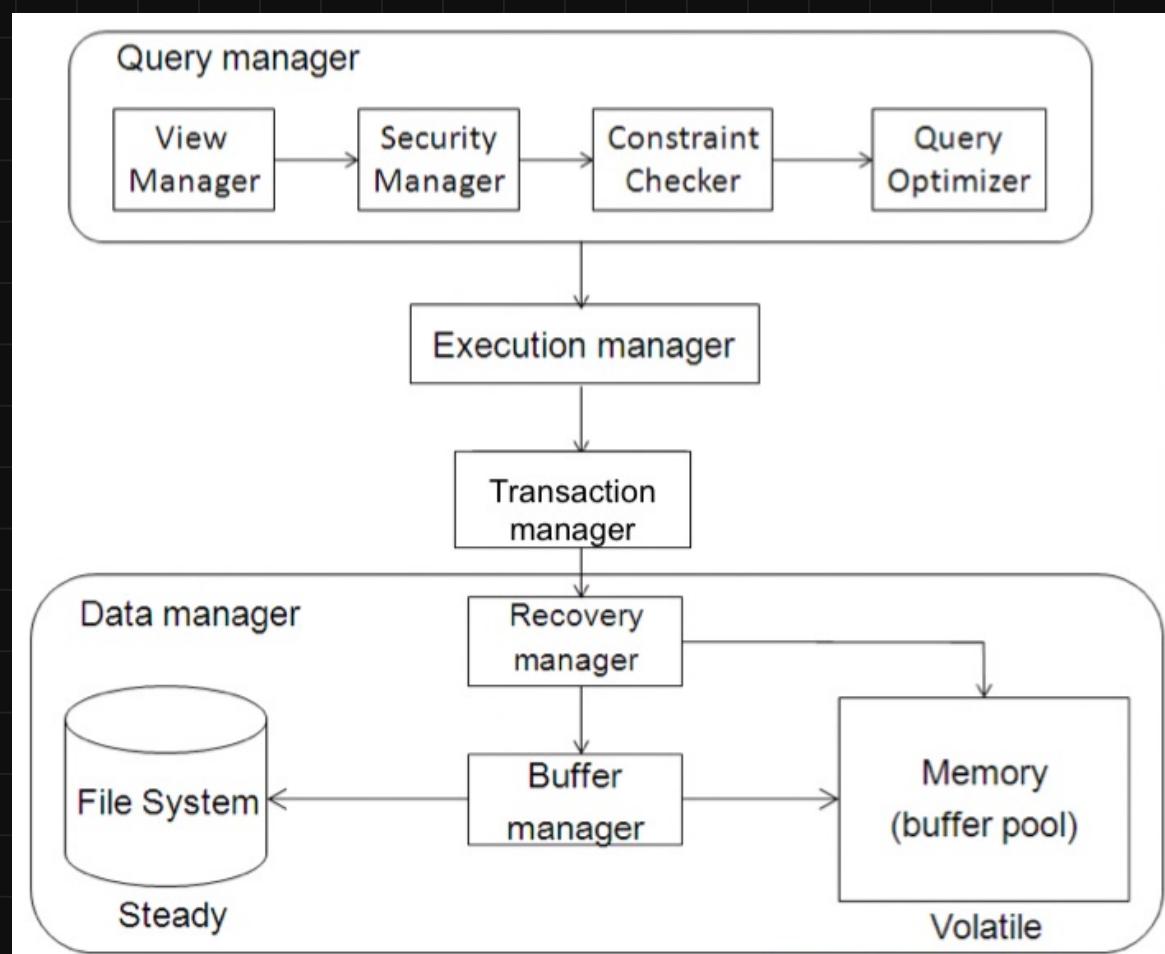


Aquesta arquitectura de 3 nivells

ens permet:

1. Independència física de dades
 - * Els canvis a l'esquema intern no afecten al conceptual ni a l'extern, però caldrà refer les correspondències entre l'esquema conceptual i l'intern.
2. Independència lògica de dades
 - * Els canvis a l'esquema conceptual no afecta als esquemes externs que no facin referència a les classes, atributs, ..., modificats
 - * Els canvis als esquemes externs no afecta ares

ARQUITECTURA d'un SGBD



USUARIS DE BD

* Usuaris informàtics

- Dissenyador BD
- Programador d'apps
- Administrador BD
- Implementador SGBD

* Usuaris no informàtics

usuaris paramètrics :

- Utilitzen les apps
- no coneixen la BD ni el seu uenguatge

usuaris finals :

- Realitzen consultes
- usen un uenguatge de consulta d'alt nivell
- Coneixen part de l'estrucció de BD i el uenguatge de consulta

•
•
•

T

1

2

3

4

5

6

7

8

9

10

11

12

Section Two

El model relational

[BACK TO INDEX](#)

OBJECTIUS

- BD = estructura lògica
- Estructura simple i uniforme → conjunt de relacions
Els valors que contenen relacions són atòmics

ESTRUCTURA DE DADES

exemple de BD :

EMPLEATS	DNI	Nom	Sou
	40.444.255	Maria Domínguez	1500
	33.567.711	Pere Roca	2000
	55.898.425	Carles Bueno	2500
	77.232.144	Elena Pla	2000

- ★ Cardinalitat = nº tuples
- ★ Grau = nº atributs

1. VISIÓ INFORMAL D'UNA RELACIÓ

- Cada relació pot "visualitzar-se" com una taula
- Fila → Tupla (coleccions de dades intrarelacionades)
- Columna → Atribut (del mateix domini)

2. COMPONENTS I ESQUEMA D'UNA RELACIÓ

COMPONENTS:

- capçalera (esquema) → nom relació + conj. atributs
- cos (extensió relació)

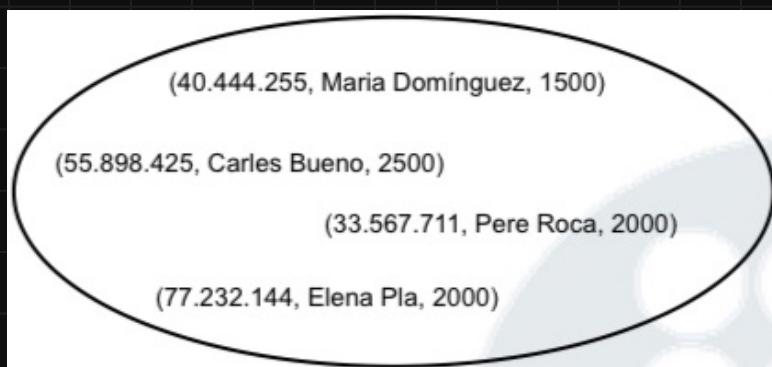
3. ATRIBUT I DOMINI

- Atribut: rol o paper que exerceix un domini en un esquema de relació
- Domini: conjunt valors atòmics
 - Predefinitos (integer)
 - Definitos per l'usuari (nom, num-dni, dpt...)

4. VALORS NULS

El valor d'un atribut per a una tupla concreta d'una relació pot ser desconegut o inaplicable en aquests casos s'usa un valor nul (NULL)

5. EXTENSIÓ DE LA RELACIÓ



- L'extensió d'una relació d'esquema $R(A_1, A_2, \dots, A_j)$ és un conjunt de tuples $t_i = \langle v_{i1}, v_{i2}, \dots, v_{ij} \rangle$ on v_{ij} es un valor del domini A_j o bé NULL.
- Tupla → element de l'extensió d'una relació

6. CARDINALITAT I GRAU

- Cardinalitat: nº tuples de la seva extensió
- Grau: nº atributs del seu esquema de relació

7. PARANYS DE LA VISIÓ INFORMAL D'UNA RELACIÓ

- Els valors dels atributs són atòmics
- No hi han tuples repetides
- No hi ha ordre entre les tuples
- No hi ha ordre entre els atributs

8. CLAUS D'UNA RELACIÓ

- SUPERCLAU D'UNA RELACIÓ: subconjunt dels atributs de l'esquema de la relació que identifica les tuples de l'extensió de la relació
- CLAU D'UNA RELACIÓ: superclau de la relació que no té cap subconjunt propi que sigui també superclau (\Rightarrow clau candidata)

ex:

EMPLEAT(DNI, NSS, Nom, Telèfon, Sou)

Superclaus: {DNI, NSS, Nom, Telèfon, Sou}
{DNI, Nom}, {DNI}, etc.

Claus candidates: {DNI}, {NSS}

DESPATX(Edifici, Número, Superficie)

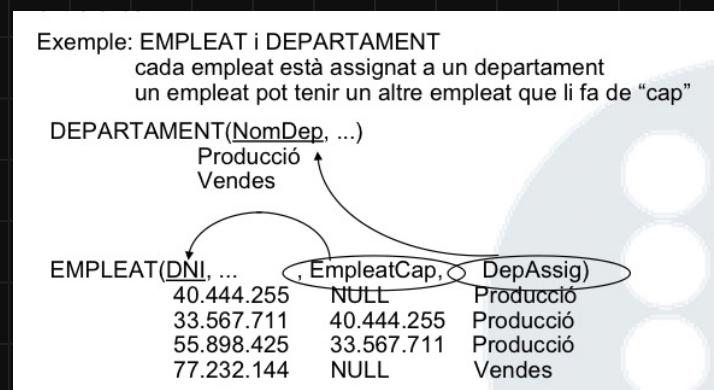
C6	119	15
C6	120	15
D4	119	10

Clau candidata: {Edifici, Número}

8.1. CLAU PRIMÀRIA

- Una de les claus candidates \Rightarrow CLAU PRIMÀRIA
- CLAU ALTERNATIVA \Rightarrow clau candidata no designada primària

8.2. CLAUS FORANES



- Les tuples poden necessitar connexions entre elles
- una clau forana d'una relació és \Rightarrow subconjunt d'atributs de l'esquema de la relació que referencia clau primària d'una altre relació.

- Característiques
 - \rightarrow té el mateix nombre d'atributs que la clau primària referenciada
 - \rightarrow els atributs que la formen han de tenir dominis compatibles amb la clau primària referenciada

9. CONNEXIONS AL MODEL RELACIONAL

- MODEL UNIFORME (només té un tipus d'element "les relacions")
- les connexions es fan \Rightarrow claus forana

OPERACIONS

- Permeten manipular
 - \rightarrow Actualització (inserir, modificar o borrar tuples)
 - \rightarrow consulta (obtenció de dades) \rightarrow àlgebra relacional
 - \rightarrow càlcul relacional

SQL incorpora càlcul relacional però també incorpora àlgebra relacional.

REGLES D'INTEGRITAT

REGLA D'INTEGRITAT D'ENTITAT

- Fà referència a les claus primàries
 - \rightarrow les claus han de tenir valors únics
 - \rightarrow No poden tenir valor NULL

REGLA D'INTEGRITAT REFERENCIAL

- Fà referència a les claus foranes

DEPARTAMENT(NomDep, ...)	
	Producció
	Vendes
EMPLEAT(DNI, ...)	
40.444.255	DepAssig)
33.567.711	Producció
55.898.425	Producció
77.232.144	Vendes
25.250.333	NULL
	Marketing

→ Estableix que només poden ser valors de la clau primària referenciada o NULLS

*RESTRICCIÓ: no permet esborrar / modificar una clau primària referenciada a una clau forana.

*CASCADA: en cas d'esborrar / modificar una clau primària referenciada a una clau forana s'esborren / modifiquen totes les referenciades.

*ANUL·LACIÓ: en cas d'esborrar / modificar una clau primària referenciada a una clau forana s'assignen valors nuls a totes les referenciades

REGLA D'INTEGRITAT DEL Domini

- un valor no nul ha de pertànyer al domini d'aquest ($\text{int} \rightarrow 6 \text{ int} \wedge 6.5$)
- les operacions que es poden aplicar sobre els valors depenen del domini ($\text{DNI} \neq 'Pere'$) i tenen que ser dominis predefinitos.

•
•
•
T
1
2
3
4
5
6
7
8
9
10
11
12

Section Three

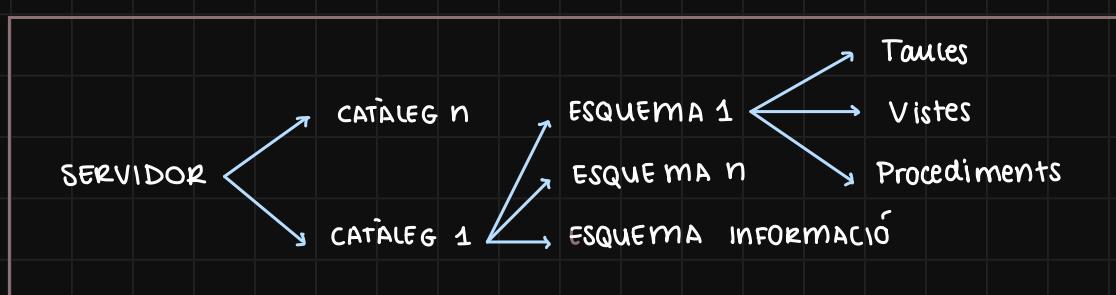
Components logics

[BACK TO INDEX](#)

ESQUEMAS

- Un esquema per un SGBD és un conjunt definit i relacionat de components de dades i de components de control
- ESQUEMA → contenidor de components lògics
 - Taules
 - Vistes
 - Procediments
- Sentència de creació: CREATE SCHEMA nom-schema [AUTHORIZATION usuari] [llista elements]
- Sentència d'eliminació: DROP SCHEMA nom-schema [RESTRICT / CASCADE]
 - RESTRICT: borra l'esquema si està completament buit
 - CASCADE: borra un esquema junt amb tots els seus elements.
- Sentència per utilitzar/modificar un schema: SET SEARCH PATH TO nom-schema
- Ens permet centralitzar tasques administratives que d'altre manera hauríem de repetir individualment (engegar, atorgar privilegis d'un conj. de components lògics a un usuari)

SERVIDOR, CÀTALOG I ESQUEMES



- CÀTALOG és un grup d'esquemes, un dels quals és l'esquema d'informació
- ESQUEMA
- SERVIDOR (cluster) conté 0 o més catàlegs

- No hi ha sentències a SQL standar per crear, destruir o modificar catàlegs (depen de cada implementador)

CONNEXIONS, SESSIONS I TRANSACCIONS

- Sentència per establir connexió: CONNECT TO nom-server [AS nom-connexió] [USER usuari]
SET SCHEMA nom-schema
- Sentència per destruir la connexió: DISCONNECT nom-connexió [DEFAULT / PUBLIC / ALL]

ASSERTIONS

- Són restriccions d'integritat que afecten a més d'una taula
- Sentència per crear-ne una: CREATE ASSERTION nom-asserçio CHECK (condició)
- La o les condicions d'una asserció són de tipus bool.

VISTES

- Es una consulta SQL que té com a resultat una "taula" virtual
- Sentència per crear una vista: CREATE VIEW nom-vista AS SELECT (nom-columna, ...)
FROM taula/es
WHERE (condició/ns)
no deixa fer actualitzacions que violin les condicions → WITH CHECK OPTION

- No totes les vistes són actualitzables, una vista és actualizable si només conté select sobre una única taula, sense agregats ni Distincts, el select conté tots els atributs amb restriccions NOT NULL i PK, sense GROUP BY.

PRIVILEGIS

- Otorguen o treuen autorització per fer algunes coses a una BD.
- Per otorgar drets : GRANT privilegis ON objecte TO user [WITH GRANT OPTION]
- Per treure drets : REVOKE [GRANT OPTION FOR] privilegis ON objecte FROM user { CASCADE | RESTRICT } y
- Tipus de privilegis : SELECT , INSERT , UPDATE , DELETE.

ROLS

- Es una agrupació de privilegis definida per un grup d'usuaris específics
- Sentència de creació : CREATE ROLE nom_role

GRANT privilegi ON Objetos TO nom_role

GRANT nom_role TO user

•
•
T
1
2
3
4
5
6
7
8
9
10
11
12

Section Four

Introducció al diseny de bases de dades relacionals

[BACK TO INDEX](#)

UML

ASSOCIACIONS

- Es la representació d'una relació entre dos o més objectes

BINÀRIES



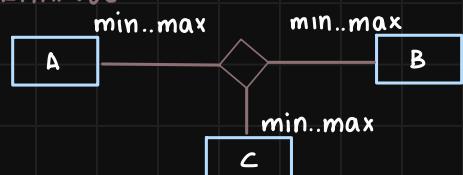
- Transformacions :

un → molts ($0..1 \rightarrow *$) = s'afegeix una foreign key / atribut a la classe (*).

un → un ($0..1 \rightarrow 0..1$) = Afegim a qualsevol de les dues classes una FK que referenciï a l'altra.

molts → molts ($* \rightarrow *$) = es crea una nova taula que conte les PK de les dues relacions.

TERNÀRIES



- Transformacions :

molts → molts → molts ($* \rightarrow * \rightarrow *$) = La primary key de la nova relació està formada per les primary key de les tres classes.

un → molts → molts ($0..1 \rightarrow * \rightarrow *$) = La pk de la nova relació està formada per les pk de les dues classes *.

un → un → molts ($0..1 \rightarrow 0..1 \rightarrow *$) = La pk de la nova relació està formada per la pk de la classe * i qualsevol de les altres dos.

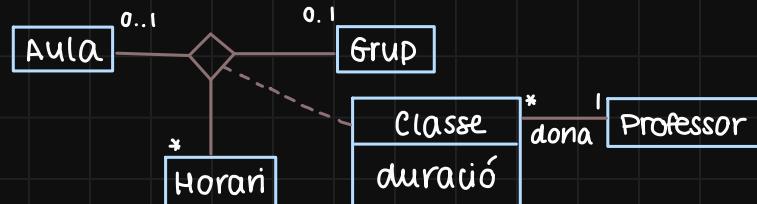
RECURSIVES



- Ex :

Persona (Dni, ... , DniParent) ↳ DniParent y referencia Persona

ASSOCIATIVES



- Ex :

Horari (dia, hora)

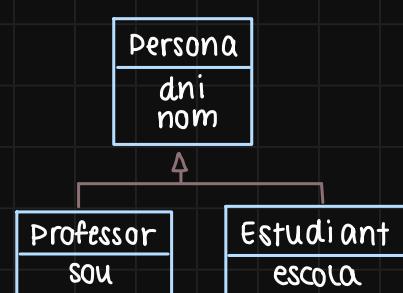
Aula (cod, ...)

Grup (num, ...)

Prof (Dni, ...)

Classe (dia, hora, cod, num, prof, duracio)

GENERALITZACIÓ

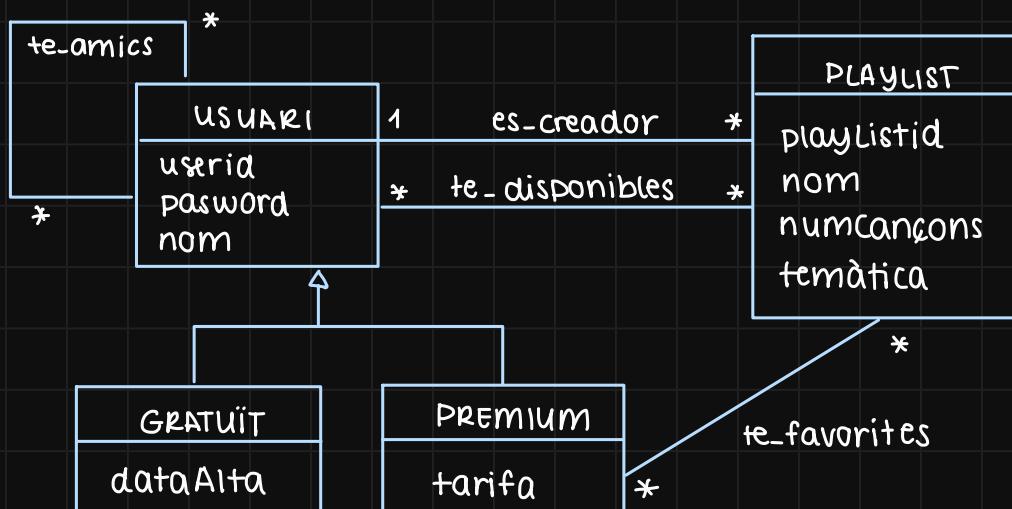
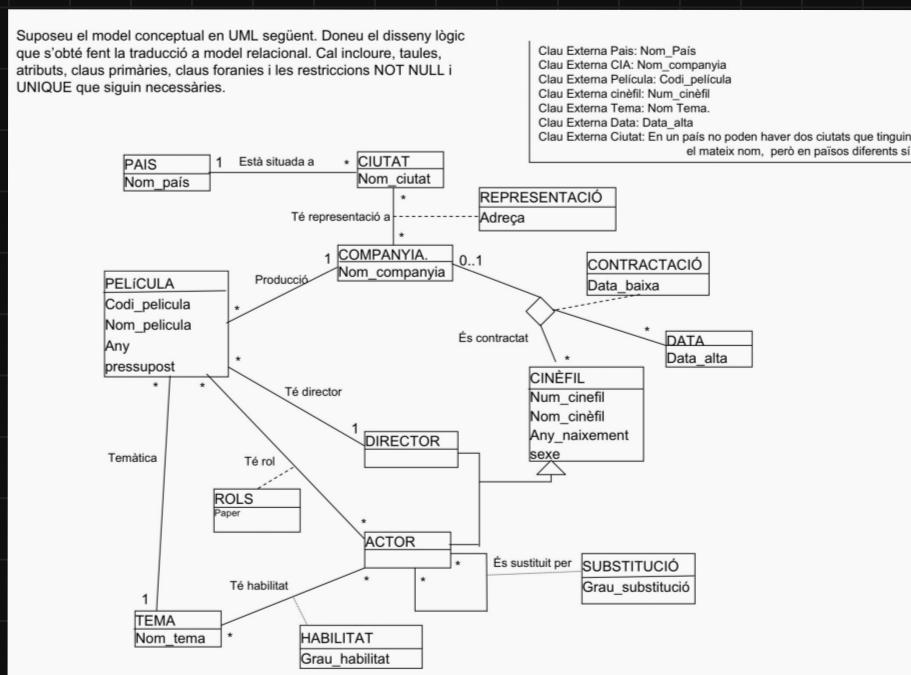


Persona (dni, nom)

Empleat (dni, sou)

Estudiant (dni, escola)

ex.

USUARI (userid, psw, nom)GRATUIT (userid, dataAlta) {userid} y references USUARIPREMIUM (userid, tarifa) {userid} y references USUARIPLAYLIST (playlistid, nom, numcancions, tematica, creador) {creador} y references USUARI, creador NOT NULLAMICS (userid, amicid) {userid} y references USUARI, {amicid} y references USUARIFAVORITS (userid, playlistid) {userid} y references PREMIUMDISPONIBLES (userid, playlistid) {userid} y references USUARIPAIS (nompais)CIUTAT (nomciutat, Pais) {Pais} ref. paisCOMPANYIA (nomcompanyia)REPRESENTACIÓ (nomcompanyia, nomciutat, pais, Adreça) {pais} y ref. pais {nomciutat} y ref.ciutatCINEFIL (numcinefil, nomcinefil, anyneixament, sexe)CONTRACTACIÓ (numcinefil, dataalta, nomcompanyia, databaixa) {numcinefil} y ref. cinefil {nomcompanyia} y ref. companyia NOT NULLDIRECTOR (numdirector) {numdirector} y ref. cinefilACTOR (numactor) {numactor} y ref. cinefil → ref. companyia, NOT NULLPELÍCULA (codedpel, nom, any, pressupost, nomcompanyia, director, tema) → ref. Tema NOT NULLTEMA (nomtema) → ref. Director, NOT NULLROLS (actor, pelcula, paper) {actor} y ref. actor, {pelcula} y ref. pelculaHABILITAT (actor, tema, grau) {actor} y ref. actor, {tema} y ref. temaSUBSTITUCIÓ (actor, sustitució, grausust) {actor} y ref. actor, {sustitució} y ref. actor

•
•
•
T
1
2
3
4
5
6
7
8
9
10
11
12

Section Five

Transaccions i concurrencia

[BACK TO INDEX](#)

Les SGBD han d'asegurar un accés simultani de múltiples usuaris a la BD sense comprometre la fiabilitat.

Per que això passi s'utilitzen un conjunt d'operations de lectura i actualització de la BD que acaben confirmant o cancelant els "canvis".

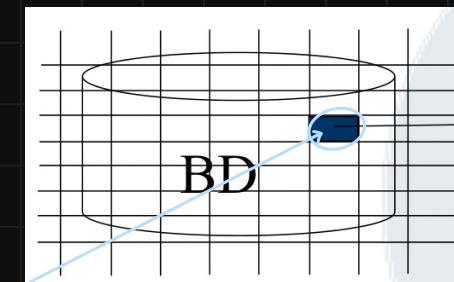
- INICI EXPLÍCIT : START TRANSACTION
- INICI IMPLÍCIT: qualsevol sentència SQL
- ACABAMENT : COMMIT / ROLLBACK

ACID → Atomicitat

Consistència

Aïllament

Definitivitat



SERIALITZABILITAT

Les transaccions són op. de dos tipus

Lectura $R(G)$, $RUC(G)$
Escriptura $W(G)$

sobre grànuls (unitat de dades controlades per la SGBD)

SQL SOBRE GRANULS

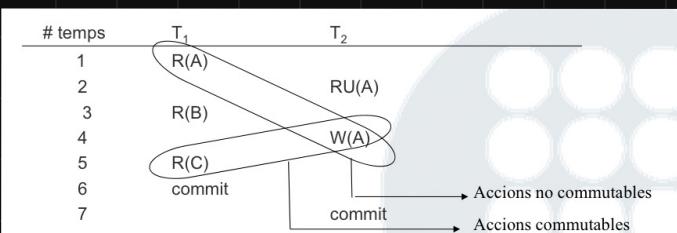
SQL	Op. Granuls
SELECT	$R(G)$: lectura granul G .
INSERT / UPDATE / DELETE	$RUC(G)$: lectura amb intenció de modificació posterior de G. $W(G)$: escriptura (modificació) de G.

ACCIONS CONFLICTIVES

- En un horari dues accions es consideren conflictives (o accions no commutables) si :

1. Pertanyen a transaccions diferents
2. L'ordre en que s'executen pot afectar el valor del granul que s'ha llegit o al seu valor final

ex :



★ Els conflictes només poden aparèixer quan les dues accions són sobre el mateix granul i almenys una és d'escriptura

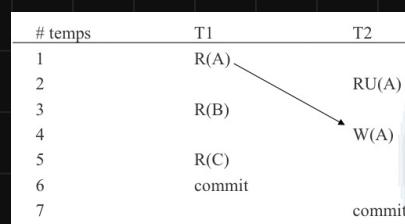
GRAF DE PRECEDÈNCIES



Aquest graf és fà si: $\exists op_i(G) \in T_i \wedge \exists op_j(G) \in T_j$ tal que

$op_i \wedge op_j$ no commutables
 op_i s'executa abans que op_j

ex :



HORARI SERIALITABLE

- un horari és serialitzable (correcte) si \Rightarrow el seu graf de precedències NO té cicles

RECUPERABILITAT

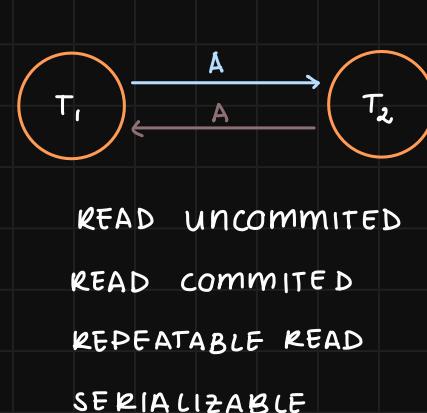
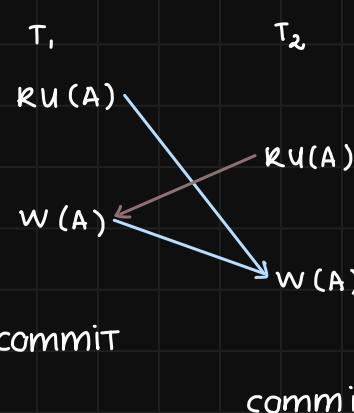
- En cas de cancel·lar una transacció hem de poder recuperar els valors anteriors
- Un horari compleix el criteri de recuperabilitat si \Rightarrow CAP transacció T_2 que uegeix/escriu a un granul escrit per un altre transacció T_1 confirma sense que T_1 confirmi abans.

INTERFERÈNCIES

- o Es produeixen quan les transaccions no s'ajullen adequadament
- ACTUALITZACIÓ PERDUDA

- o Quan es perd el canvi que ha efectuat una op d'escriptura

- o ex: T_1



READ UNCOMMITTED

READ COMMITTED

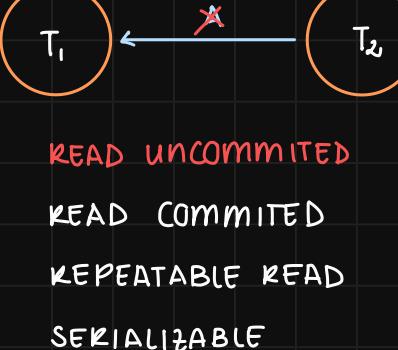
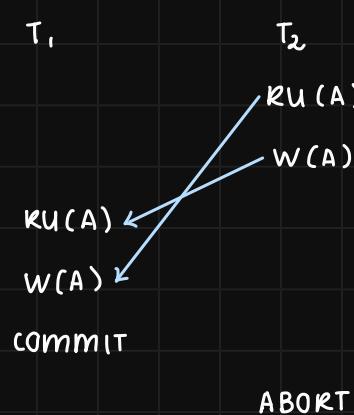
REPEATABLE READ

SERIALIZABLE

LECTURA NO CONFIRMADA

- o Succeeix quan T_1 vegeix una dada que ha estat modificada per T_2 i és una dada no definitiva (per exemple T_1 avorta o la torna a modificar)

- o ex: T_1



READ UNCOMMITTED

READ COMMITTED

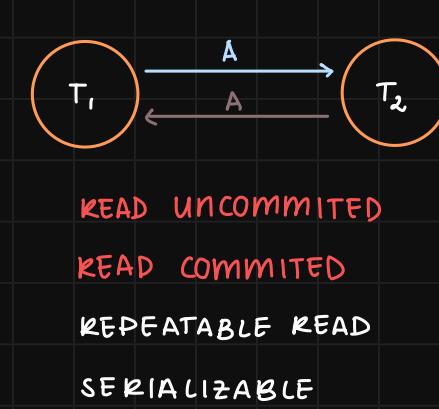
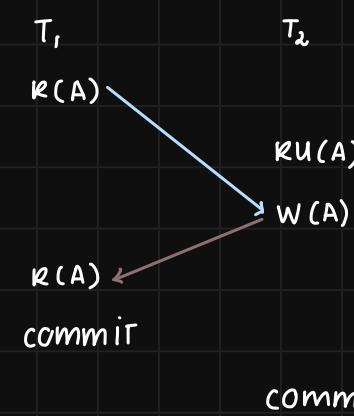
REPEATABLE READ

SERIALIZABLE

LECTURA NO REPETIBLE

- o Quan una transferència T_1 vegeix dos vegades la mateixa dada i obté dos valors diferents per un canvi efectuat per T_2 .

- o ex: T_1



READ UNCOMMITTED

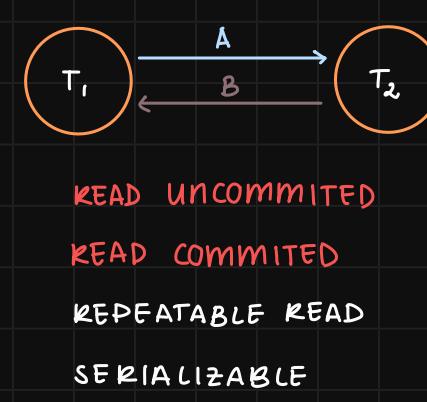
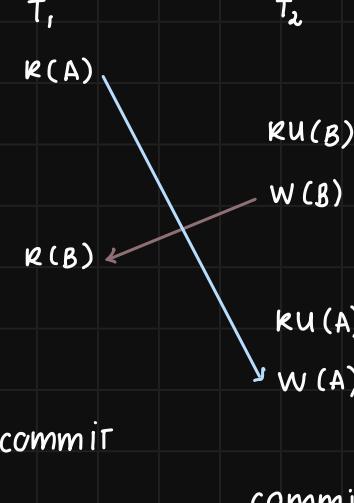
READ COMMITTED

REPEATABLE READ

SERIALIZABLE

ANALISI INCOMPATIBLE

- o Quan en mitj d'una transacció T_1 , una T_2 canvia l'estat d'una variable
- o ex: T_1



READ UNCOMMITTED

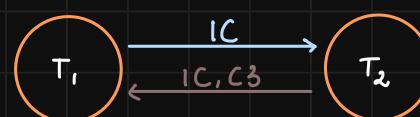
READ COMMITTED

REPEATABLE READ

SERIALIZABLE

LECTURA NO REPETIBLE

- Quan una transacció T_1 llegeix un conjunt de dades relacionades i una T_2 afageix noves dades que pertanyen a aquest conjunt.

o ex: T_1 T_2 $R(C_1)$ $R(C_2)$ $RU(C_3)$ $W(C_3)$ $RU(C_1)$ $W(C_1)$ $R(C_1)$ $R(C_2)$ $R(C_3)$ 

READ UNCOMMITTED

READ COMMITTED

REPEATABLE READ

SERIALIZABLE

CONCURRENCIA

- Permet solucionar les interferències tal que
 - cancaixa automàticament les transaccions problemàtiques desfent els canvis
 - suspén l'execució temporalment fins que no hi hagi perill

CONTROL CONCURRENCIES AMB RESERVES

Es reserva un grànul amb certa modalitat abans d'efectuar una acció

- $LOCK(G, m)$ = permet que una transacció T reservi el grànul G , on m es una modalitat
 - $LOCK(G, S)$ shared : permet fer lectures
 - $LOCK(G, X)$ exclusive : permet fer lectures i escriptures
- $UNLOCK(G)$ = allivera la reserva de G

GRAF D'ESPERA

Si G està reservat i a T no li concedeixen la reserva uavors es suspén l'execució de T .Quan s'alliberi G uavors l'SGBD mira si pot reprendre l'execució d'alguna transacció que s'hagi suspès a causa d'aquest grànul.★ Si T_i està esperant a adquirir una reserva sobre G concedida a T_j

# temps	T_1	T_2
1	LOCK(A, S) ↗	
2	R(A)	
3		LOCK(A, X)
4	LOCK(B, S)	⋮
5	R(B)	⋮
6	LOCK(C, S)	
7	R(C)	
8	UNLOCK(A)	
9		RU(A)
10		W(A)
11	UNLOCK(B)	
12	UNLOCK(C)	
13		UNLOCK(A)
14	commit	
15		commit

# temps	T ₁	T ₂
1	LOCK(A,S)	
2	R(A)	
3	UNLOCK(A)	
4		LOCK(A,X)
5		RU(A)
6		W(A)
7		UNLOCK(A)
8	LOCK(A,S)	
9	R(A)	
10	UNLOCK(A)	
11	commit	
12		commit

★ Tot i que la base del control de concurrencia es basa amb el servei de reserves i anteriorment, aquests no garanteixen l'ausollament de les transaccions.

⇒ L'interferència de lectura no repetible es produeix igualment

PROTOCOL DE RESERVES DE DUES FASES (PR2F)

- Una transacció T segueix PR2F si reserva qualsevol gràmul en la modalitat adequada abans d'operar-hi i mai no adquireix una reserva de qualsevol gràmul després d'haver-ne auiberat qualsevol altra abans
- Si PR2F + Commit ⇒ SERIALIZABLE

RECUPERABILITAT

# temps	T ₁	T ₂
1	LOCK(A,X)	
2	RU(A)	
3	W(A)	
4	UNLOCK(A)	
5		LOCK(A,S)
6		R(A)
7		UNLOCK(A)
8		commit
9	abort	

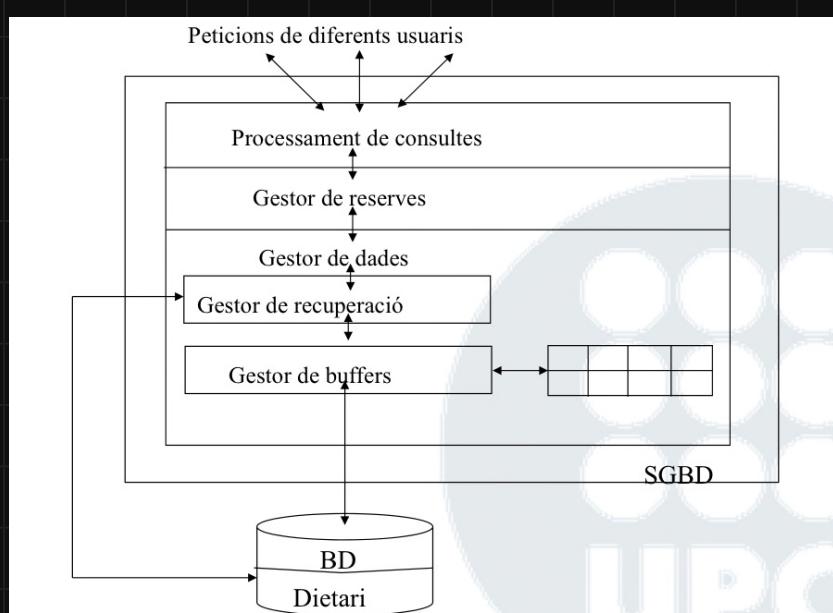
- Si T₂ confirma habans que T₁ ⇒ interferència, per això necessitem que es mantinguin les reserves fins el final de les transaccions

↳ PROTOCOL DE RESERVES EN 2 FASES (FORTAMENT) ESTRICTE

RESERVES i RELAXACIÓ

- la protecció total pot provocar
 - sobrecarga de l'SGBD
 - bajada del nivell de paralelisme
- En circumstàncies convé baixar el nivell d'ausollament per que es produeixin interferències
 - SET TRANSACTION mode-acces → (read only / read write)
 - ISOLATION LEVEL nivell-ausollament → (read uncommitted, read committed, repeatable read, serializable)
 - READ UNCOMMITTED protegeix les dades actualitzables, evitant que cap altra transacció les actualitzi fins que acaba la transacció
 - READ COMMITTED protegeix parcialment les lectures, impedint que un altre transacció legeixi dades que encara no s'han confirmat
 - REPEATABLE READ impedeix fins que acaba la transacció que una altre actualitzi una dada que s'ha llegit
 - SERIALIZABLE ofereix ausollament total i evita qualsevol tipus d'interferència incloent-hi els fantasmares.

RECUPERACIÓ i RESTAURACIÓ



RESTAURACIÓ

- Garantitza que la BD compleixi l'atomicitat de les transaccions.

RECUPERACIÓ (reconstrucció)

- Recupera l'estat de la BD quan es produeix una pèrdua total o parcial
 - Reconstrucció cap endavant: refà els canvis d'una transacció confirmada
 - Reconstrucció cap enrere: desfà els canvis d'una transacció abortada

•
•
•
T
1
2
3
4
5
6
7
8
9
10
11
12

Section Six

Estructures físiques d'emmagatzemat i mètodes d'accés

[BACK TO INDEX](#)

MEMÒRIA

- Externa: més lenta, més barata
- Interna: més ràpida, més cara, volàtil i de capacitat reduïda

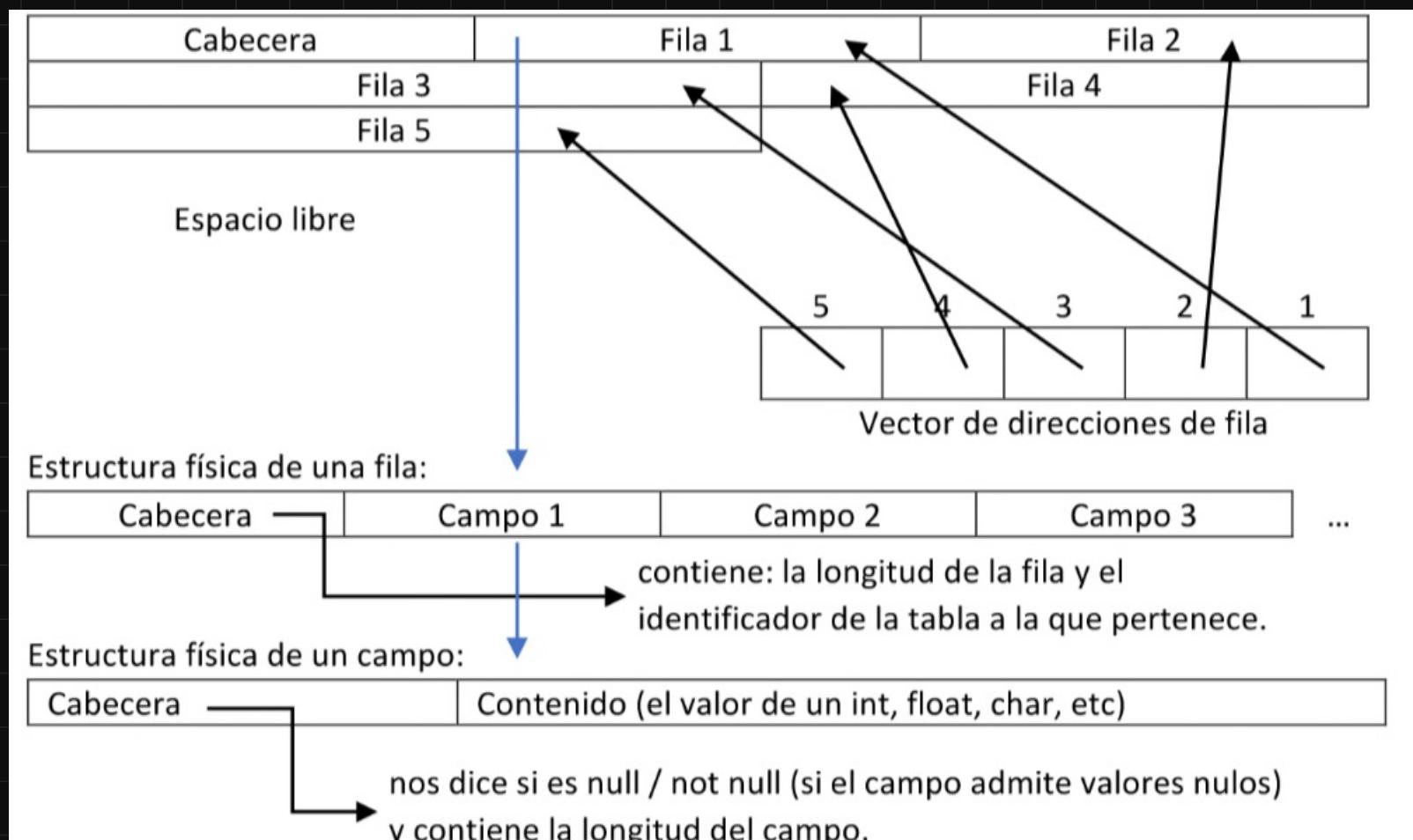
Donat que no existeix un estàndar per els fitxers de les bases de dades, aprendrem el patró que segueixen.

- Una BD té tres nivells:

1) NIVELL FÍSIC

Les dades s'emmagatzem en discs magnètics controlats per el SO que és el que realment efectua les lectures i escriptures, i l'SGBD decideix quan fer aquestes operacions i el que coneix com estan físicament estructurades les dades i com interpretar-los.

- Fitxers: unitat global a partir de la qual l'SO gestiona les dades (conj. d'extensions)
- Extensions: unitat d'adquisicions d'espai per cada fitxer (és un múltiple enter de la pàgina) ⇒ n° pàgines que l'SO necessita a petició de la SGBD quan detecta que necessita més espai.
- Pàgina: conté i emmagatzema dades a nivell lògic.



2) NIVELL VIRTUAL

Fà d'intermediari entre el nivell físic i lògic:

- Si les taules son molt grans ⇒ Hi ha fragments en els diferents dispositius ⇒ s'associa un fragment a un fitxer diferent.
- Si les taules son molt petites ⇒ Hi ha que agrupar-les en un fitxer
- Pels objectes grans, que s'almacenen separadament.
- Pels indexos, disparadors, restriccions ...

CORRESPONDÈNCIA PÀGINES REALS - VIRTUALS ↴

